



Developing ‘EasyTalk’ – a writing system utilizing natural language processing for interactive generation of ‘Leichte Sprache’ (Easy-to-Read German) to assist low-literate users with intellectual or developmental disabilities and/or complex communication needs in writing

by
Ina Steinmetz

Approved Dissertation thesis for the partial fulfilment of the requirements for a
Doctor of Natural Sciences (Dr. rer. nat.)
Fachbereich 4: Informatik
Universität Koblenz

Chair of PhD Board: Prof. Dr. Ralf Lämmel
Chair of PhD Commission: Jun.-Prof. Dr. Robert Panitz
Examiner and Supervisor: Prof. Dr. Karin Harbusch
Further Examiners: Prof. Dr. Martin Volk

Date of the doctoral viva: 20.06.2023

Erklärung

Hiermit erkläre ich gemäß §10 Abs. 3 Punkt 4 der Promotionsordnung des Fachbereichs 4: Informatik der Universität Koblenz Landau,

- dass ich die vorliegende Dissertation mit dem Titel „Developing ‘EasyTalk’ – a writing system utilizing natural language processing for interactive generation of ‘Leichte Sprache’ (Easy-to-Read German) to assist low-literate users with intellectual or developmental disabilities and/or complex communication needs in writing“ selbst angefertigt und alle benutzten Hilfsmittel in der Arbeit angegeben habe,
- dass ich die Dissertation oder Teile der Dissertation noch nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht habe, und
- dass ich weder diese noch eine andere Abhandlung bei einer anderen Hochschule als Dissertation eingereicht habe.

Im Fall von kooperativ erzielten Forschungsergebnissen erkläre ich:

- dass ich meine individuellen Beiträge an kooperativ erzielten Forschungsergebnissen in der Dissertation an den entsprechenden Stellen gekennzeichnet habe und dass meine Koautoren diese Einschätzung meines Beitrages teilen (siehe gesonderte Bestätigung der Koautoren nach §10 Abs. 4).

Karlsruhe, den 9. Dezember 2022



Ina Steinmetz

Acknowledgements

I would like to express my heartfelt gratitude to the following individuals for their invaluable contributions and support throughout my PhD journey:

First and foremost, I would like to express my sincere appreciation to Prof. Dr. Ralf Lämmel, Chair of the PhD Board, for his leadership in the oversight of this thesis.

I would also like to thank Jun.-Prof. Dr. Robert Panitz, Chair of the PhD Commission, for his crucial role in ensuring the quality and rigor of this research.

In addition, I would like to express my deep gratitude to Prof. Dr. Martin Volk for reviewing my thesis and providing me with insightful and thorough feedback, which has improved the quality of my thesis immensely.

Words cannot express my gratitude to Pro. Dr. Karin Harbusch, my professor and supervisor. Your commitment to my academic growth has been instrumental in shaping the outcome of this research. Your insightful feedback, rigorous examination, and continuous guidance have challenged me to strive for excellence and have not only enriched the quality of my thesis but also enhanced my skills beyond this work. I am truly grateful for your unwavering support and dedication to my academic pursuits. Thank you for being an integral part of this transformative experience.

In addition, I would like to thank *Tacton Systems GmbH* for providing essential infrastructure and resources for conducting remote interviews, which contributed greatly to the success of this research endeavour. I would also like to thank the *Schreibwerkstatt* of the *Habila Tannenhof Ulm* for their invaluable help in testing the system and providing crucial insights for this research. To all the interviewees who generously shared their time and insights, I extend my heartfelt thanks for your valuable contributions to this research.

Finally, I would like to express my deepest gratitude and appreciation to my family and friends for their encouragement, support and understanding throughout this journey. I am especially grateful to my husband who has always supported me. I am eternally grateful for your patience. Thank you for always believing in me.

Contents

Abstract	17
Kurzfassung	19
1 Introduction	21
1.1 Context	21
1.2 Research Problem	22
1.3 Rationale	24
1.4 Scope and Limitations	25
1.5 Significance	26
1.6 Structure	26
2 (Extended) Leichte Sprache	27
2.1 Introduction to Leichte Sprache	27
2.2 The target audience(s) of Leichte Sprache	30
2.3 Leichte Sprache rules for text generation	32
2.4 Extended Leichte Sprache	35
3 Writing systems for the target group	45
3.1 Symbol-based writing systems	45
3.2 Text-based writing support	53
3.3 Deep learning-based writing support	54
3.4 Systems for teaching text-production	57
4 The human-centered design of EasyTalk	59
4.1 Introduction to human-centered design	59
4.2 Specification of the context of use	60
4.3 Specification of the user requirements	64
4.4 Creation of the design solution	66
4.5 Evaluation of the design	69

5	EasyTalk	71
5.1	Natural language generation concepts	71
5.2	Theoretical system description	74
5.2.1	Base concepts	74
5.2.2	Use cases involved in sentence writing	78
5.2.3	Sentence writing activities	79
5.2.4	System components for sentence writing	82
5.2.5	Natural language generation core components	83
5.2.6	Computing word form suggestions	85
5.3	System introduction	90
5.4	Text-writing support	91
5.4.1	Letter- and word-level support	92
5.4.2	Sentence-level support	93
5.4.3	Text-level support	106
5.5	Teaching text production in a writing workshop	109
5.6	Additional features and system settings	111
5.6.1	Text functions	111
5.6.2	Options for system customization	112
6	System evaluation	115
6.1	System evaluation in interviews with expert groups	115
6.1.1	Test setup and participants	116
6.1.2	Test procedure	117
6.1.3	Test results	118
6.1.4	Changes to EasyTalk based on the test results	119
6.2	Testing with beginner L2 learners with low computer skills	120
6.2.1	Test setup and participants	120
6.2.2	Test procedure	120
6.2.3	Test results	121
6.2.4	Changes to EasyTalk based on the test results	123
6.3	Testing with IT-expert L2 learners	124
6.3.1	Test setup and participants	124
6.3.2	Test procedure	124
6.3.3	Test results	125
6.3.4	Changes to EasyTalk based on the test results	131
6.4	Testing with the target group	131
6.4.1	Test setup and participants	131
6.4.2	Test procedure	132
6.4.3	Results	133
6.4.4	Changes to EasyTalk based on the test results	141
6.5	Summary of the evaluation results	142

7	Results and discussion	147
7.1	The linguistic scope of the system	147
7.2	Writing support at the personal proficiency level	149
7.3	Intuitive dialogues for text-production theory-concepts	153
7.4	Usability of the system for the target group	155
7.5	Summary	158
8	Future Research	159
9	Conclusion	161
A	Original Test Data	165
A.1	Interviews with domain experts	165
A.1.1	LS reader	165
A.1.2	LS writer	167
A.1.3	AAC expert	168
A.1.4	Domain expert	169
A.2	L2 learners with low computer skills	170
A.2.1	Participant P1	170
A.2.2	Participant P2	172
A.2.3	Participant P3	174
A.3	L2 learners with expert computer skills	175
A.3.1	Participant P1	176
A.3.2	Participant P2	177
A.3.3	Participant P3	179
A.3.4	Participant P4	180
A.3.5	Participant P5	182
A.3.6	Participant P6	185
A.3.7	Participant P7	187
A.3.8	Participant P8	189
A.3.9	Participant P9	190
A.3.10	Participant P10	191
A.4	Participants with IDD and/or CCN	192
A.4.1	Participant P1	193
A.4.2	Participant P2	198
A.4.3	Participant P3	203
A.4.4	Participant P4	208
A.4.5	Participant P5	210
A.4.6	Participant P6	215
A.4.7	Participant P7	216
A.4.8	Participant P8	217

A.4.9	Additional eye-tracking metrics	218
B	Additions	221
B.1	Data Availability Statement	221
B.2	Ethics Statement	221
B.3	Curriculum Vitae	223

List of Figures

2.1	The European Easy-To-Read Logo	29
3.1	Two examples of non-technical aids for symbol-based Augmentative and Alternative Communication	46
3.2	Two examples of free popular symbol-based communication apps	47
3.3	<i>Gateway</i> running on an iPad	48
3.4	Layouts of different styles and complexity for <i>Mind Express</i>	50
3.5	Two layouts of <i>TD Snap Core First</i>	51
4.1	The human-centered design-process according to ISO 9241-210.	60
4.2	<i>EasyTalk</i> evaluation overview	69
5.1	Basic elements of a natural language generator	72
5.2	<i>Rich inflection</i> and <i>free word order</i> in the German language	75
5.3	Main use cases of <i>EasyTalk</i>	79
5.4	Activity diagram of writing a sentence in <i>EasyTalk</i>	80
5.5	<i>EasyTalk</i> system components	82
5.6	A class diagram of <i>EasyTalk</i> 's core component	83
5.7	The basic writing process of <i>EasyTalk</i> in five steps	90
5.8	Choosing the next possible component in <i>EasyTalk</i>	94
5.9	Handling verbs with separable verb prefix in <i>EasyTalk</i>	97
5.10	Writing complex verb constructions in <i>EasyTalk</i>	98
5.11	Writing a question with interrogative pronoun in <i>EasyTalk</i>	100
5.12	Writing a question without interrogative pronoun in <i>EasyTalk</i>	101
5.13	Writing in the preterite past in <i>EasyTalk</i>	102
5.14	Writing present perfect tense in <i>EasyTalk</i>	103
5.15	The <i>free writing mode</i> mode of <i>EasyTalk</i>	104
5.16	Connecting sentences via the Connector Panel of <i>EasyTalk</i>	107
5.17	An excerpt from the <i>EasyText</i> checklist	110
5.18	The text functions of <i>EasyTalk</i>	112
5.19	The dialog for adding proper names to the vocabulary in <i>EasyTalk</i>	114

6.1	The test texts of the low-literate L2 learners	121
6.2	The test setup presented to the IT-expert L2 learners	125
6.3	The UEQ distribution of answers to the single items	129
6.4	Two samples sessions of participants P1 and P5.	134
6.5	Eye-tracking data of participants looking at the Text Panel	135
6.6	Three heat maps of representative sentence-writing intervals . . .	137
6.7	Gaze paths of different users focusing the wh-cues of the Text Panel in different contexts of use.	138
6.8	Three gaze plots showing how P3 started to write a question . . .	139
6.9	P1's gaze plot while connecting sentences	140
6.10	<i>EasyTalk</i> - summary of the evaluation results	143
A.1	Sentences written by P1 in the test session.	193
A.2	P1's eye-tracking calibration-data	195
A.3	The gaze plots of P1's sentence-connecting intervals	196
A.4	The gaze plots and heat maps of P1's sentence-writing intervals . .	197
A.5	Sentences written by P2 in the test session.	198
A.6	P2's eye-tracking calibration-data	200
A.7	The gaze plots and heat maps of P2's sentence-connecting intervals.	201
A.8	The gaze plots and heat maps of P2's sentence-writing intervals. .	202
A.9	The sentences P3 wrote in the test session.	203
A.10	P3's eye-tracking calibration-data	204
A.11	The gaze plots and heat maps of P3's sentence-connecting intervals.	205
A.12	The gaze plots and heat maps of P3's sentence-writing intervals S1-S3.	206
A.13	The gaze plots and heat maps of P3's sentence-writing intervals S4-S6.	207
A.14	The sentences participant P4 wrote in the test session.	208
A.15	P4's eye-tracking calibration-data	209
A.16	The gaze plot and heat map of P4's sentence-connecting interval .	209
A.17	The gaze plots and heat maps of P4's sentence-writing intervals. .	210
A.18	The sentences P5 wrote in the test session	211
A.19	P5's eye-tracking calibration-data	212
A.20	The gaze plots and heat maps of P5's sentence-connecting intervals.	213
A.21	The gaze plots and heat maps of P5's sentence-writing intervals. .	214
A.22	The sentences P6 wrote in the test session.	215
A.23	The sentences P7 wrote in the test session.	217
A.24	The sentence P8 wrote before she opted out of the test session. . .	217
A.25	The areas of interest defined for eye tracking.	220

List of Tables

2.1	Milestones in LS history in the course of the inclusion movement .	28
2.2	Relevant LS rules for text generation	33
2.3	Summary of Extended Leichte Sprache constructions	43
4.1	Personas representing of prototypical users of <i>EasyTalk</i>	62
5.1	List of constituents in a main declarative sentence	78
5.2	Examples for the two component states in <i>EasyTalk</i>	95
5.3	The text of the <i>EasyTalk</i> example writing session	108
5.4	Options for system customization	113
6.1	Data on the L2 learner test participants with low computer skills .	120
6.2	Data on the IT-expert Second language; L2 learner: second lan- guage learner (L2) learner test participants	124
6.3	The IT experts' choices made in the Connector Panel.	127
6.4	The results of the UEQ user experience questionnaires.	128
6.5	Data on the test participants of the target group	132
A.1	Eye-tracking calibration values of participants P1-P5.	219
A.2	Panel visit durations in sentence-writing intervals	219
A.3	Panel visit durations in sentence-connecting intervals	219
A.4	Visit durations in the areas of interest of the Connector Panel . . .	220

Glossary

3rdPers third person.

AAC Augmentative and Alternative Communication.

Acc accusative.

AOI area of interest.

APPRART preceding prepositions with agglutinated definite determiner.

CCN Complex Communication Needs.

CEFR Common European Framework of Reference for Languages.

ELS Extended Leichte Sprache.

HCD human-centered design.

IDD intellectual or developmental disability.

Ind indicative.

KOUI preposition used to introduce infinitive clause.

L2 second language; L2 learner: second language learner.

LS Leichte Sprache.

LST Leichte-Sprache Treebank.

Neut neuter.

NLG natural language generation.

NLP natural language processing.

NN noun.

Nom nominative.

NP noun phrase.

POS the part-of-speech.

PP Prepositional phrase.

PPER irreflexive personal pronoun.

Pres present tense.

PRO PDS and PIS and PPOSS and PRELS and PRF and PPER and PWS.

RST Rhetorical Structure Theory.

Sing singular.

SVO subject verb object.

SVP separable verb prefix.

TOI time of interest.

TüBa-D/Z treebank of written German.

TüBa-D/S treebank of spoken German (systematically called VERBMOBIL).

UCD user-centered design.

UI user interface.

V2 verb second word order in German main declarative clauses.

VERBMOBIL synonym for TüBa-D/S.

VF verb final word order in German subordinate clauses.

VUI voice user interface.

VVFIN lexical verb as finite form.

Abstract

Leichte Sprache (LS, easy-to-read German) is a simplified variety of German. It is used to provide barrier-free texts for a broad spectrum of people, including low-literate individuals with learning difficulties, intellectual or developmental disabilities (IDD) and/or complex communication needs (CCN). In general, LS authors are proficient in standard German and do not belong to the aforementioned group of people. Our goal is to empower the latter to participate in written discourse themselves. This requires a special writing system whose linguistic support and ergonomic software design meet the target group's specific needs.

We present *EasyTalk* a system profoundly based on natural language processing (NLP) for assistive writing in an extended variant of LS (ELS). *EasyTalk* provides users with a personal vocabulary underpinned with customizable communication symbols and supports in writing at their individual level of proficiency through interactive user guidance. The system minimizes the grammatical knowledge needed to produce correct and coherent complex contents by intuitively formulating linguistic decisions. It provides easy dialogs for selecting options from a natural-language paraphrase generator, which provides context-sensitive suggestions for sentence components and correctly inflected word forms. In addition, *EasyTalk* reminds users to add text elements that enhance text comprehensibility in terms of audience design (e.g., time and place of an event) and improve text coherence (e.g., explicit connectors to express discourse-relations).

To tailor the system to the needs of the target group, the development of *EasyTalk* followed the principles of human-centered design (HCD). Accordingly, we matured the system in iterative development cycles, combined with purposeful evaluations of specific aspects conducted with expert groups from the fields of CCN, LS, and IT, as well as L2 learners of the German language. In a final case study, members of the target audience tested the system in free writing sessions. The study confirmed that adults with IDD and/or CCN who have low reading, writing, and computer skills can write their own personal texts in ELS using *EasyTalk*. The positive feedback from all tests inspires future long-term studies with *EasyTalk* and further development of this prototypical system, such as the implementation of a so-called *Schreibwerkstatt* (writing workshop).

Kurzfassung

Leichte Sprache (LS) ist eine vereinfachte Varietät des Deutschen in der barrierefreie Texte für ein breites Spektrum von Menschen, einschließlich gering literalisierten Personen mit Lernschwierigkeiten, geistigen oder entwicklungsbedingten Behinderungen (IDD) und/oder komplexen Kommunikationsbedürfnissen (CCN), bereitgestellt werden. LS-Autor*innen sind i.d.R. der deutschen Standardsprache mächtig und gehören nicht der genannten Personengruppe an. Unser Ziel ist es, diese zu befähigen, selbst am schriftlichen Diskurs teilzunehmen. Hierfür bedarf es eines speziellen Schreibsystems, dessen linguistische Unterstützung und softwareergonomische Gestaltung den spezifischen Bedürfnissen der Zielgruppe gerecht wird.

EasyTalk ist ein System basierend auf computerlinguistischer Verarbeitung natürlicher Sprache (NLP) für assistives Schreiben in einer erweiterten Variante von LS (ELS). Es stellt den Nutzenden ein personalisierbares Vokabular mit individualisierbaren Kommunikationssymbolen zur Verfügung und unterstützt sie entsprechend ihres persönlichen Fähigkeitslevels durch interaktive Benutzerführung beim Schreiben. Intuitive Formulierungen für linguistische Entscheidungen minimieren das erforderliche grammatikalische Wissen für die Erstellung korrekter und kohärenter komplexer Inhalte. Einfache Dialoge kommunizieren mit einem natürlingsprachlichen Paraphrasengenerator, der kontextsensitiv Vorschläge für Satzkomponenten und korrekt flektierte Wortformen bereitstellt. Außerdem regt *EasyTalk* die Nutzer*innen an, Textelemente hinzuzufügen, welche die Verständlichkeit des Textes für dessen Leserschaft fördern (z.B. Zeit- und Ortsangaben) und die Textkohärenz verbessern (z.B. explizite Diskurskonnectoren).

Um das System auf die Bedürfnisse der Zielgruppe zuzuschneiden, folgte die Entwicklung von *EasyTalk* den Grundsätzen der menschenzentrierten Gestaltung (UCD). Entsprechend wurde das System in iterativen Entwicklungszyklen ausgereift, kombiniert mit gezielten Evaluierungen bestimmter Aspekte durch Gruppen von Expert*innen aus den Bereichen CCN, LS und IT sowie L2-Lernende der deutschen Sprache. Eine Fallstudie, in welcher Mitglieder der Zielgruppe das freie Schreiben mit dem System testeten, bestätigte, dass Erwachsene mit geringen Lese-, Schreib- und Computerfähigkeiten mit IDD und/oder CCN mit *EasyTalk* eigene persönliche Texte in ELS verfassen können. Das positive Feedback aller Tests inspiriert Langzeitstudien mit *EasyTalk* und die Weiterentwicklung des prototypischen Systems, wie z.B. die Implementierung einer s.g. Schreibwerkstatt.

Chapter 1

Introduction

Simplified languages are instruments for inclusion aiming to overcome language barriers. *Leichte Sprache* (LS) is a variety of German with reduced complexity (cf. *Basic English*). So far, LS is mainly provided for but rarely written by its target group, which includes low-literate people with *intellectual or developmental disability* (IDD) and/or *complex communication needs* (CCN). One contributing reason may be the need for more technical support in the form of an assistive writing system.

We aim to enable the LS target group to participate in written discourse themselves. To this end, we develop *EasyTalk*, an assistive LS-writing system profoundly based on *natural language generation* (NLG) that minimizes the writing skills needed to produce correct and coherent LS texts and takes into account the target group's specific requirements for a writing system.

This Chapter will introduce this work by first discussing the context, followed by the research problem, its rationale, scope, and significance. Finally, the structure of this work is outlined.

Parts of this work have previously been published in Steinmetz and Harbusch (2020, 2021b,a, 2022) and Harbusch and Steinmetz (2022)).

1.1 Context

Written language is essential to integration and autonomous participation in social and economic life (Bingel (2018); Lieske and Siegel (2014)). However, recent studies (cf. Grotlüschen and Buddeberg (2020)) report that more than 10 percent of German-speaking adults have low literacy skills. People with intellectual and developmental disabilities and/or complex communication needs often belong to this group (Light et al., 2019; Buddeberg et al., 2020).

Leichte Sprache (Easy-to-Read German) is an established tool for reducing

barriers in written German. LS is a simplified variety of German defined by rules, such as “*Do not use commas*”, “*Avoid inversions*”, and “*Use clear sentence structure*” (Inclusion Europe (2009); BITV2.0 (2011); Netzwerk Leichte Sprache (2013)). The LS rules aim to reduce language complexity to promote text comprehensibility. Comparable to other easy languages (cf. Lindholm and Vanhatalo (2021)), like *Easy-to-Read English*, it was developed for the spectrum of people with intellectual disabilities or learning difficulties¹, as part of the plain language movement of the 2000s (Maaß and Bredel, 2016, p. 60). In Germany, LS is legally anchored as the means of choice for providing accessible information in text form (BITV2.0, 2011). Text in LS is generally produced by authors who are proficient in standard German (Maaß and Bredel, 2016, p.25) and then evaluated for ease of comprehension by people with intellectual disabilities or learning difficulties (BITV2.0, 2011; Netzwerk Leichte Sprache, 2013). The use and production of LS text are thus asymmetrical (Maaß and Bredel, 2016, 25) – generally, LS readers do not actively contribute to the written discourse themselves.

One reason may be the need for more technical support in the form of an assistive writing system. Systems that support writing in LS are usually made for authors with mature writing skills and focus on simplifying texts and their compliance with LS rules. To the best of our knowledge, there is no easy-to-use LS writing system designed for low-literate users with IDD and/or CCN that offers linguistic support beyond the phrasing of simple, partly personal sentences.

1.2 Research Problem

In recent years, the availability and variety of writing support has increased significantly. *Voice user interfaces* (VUI) and text-to-speech systems have long been widely available on computers, tablets, and smartphones. Basic grammar- and spell-checking are expected features of any text editor. In addition, an increasing variety of writing-support systems based on natural language generation attract attention (for their prospects, see, e.g., Dale and Viethen (2021); for approaches based on deep learning, see Otter et al. (2021)). Characteristic features of these systems are, for example, to provide the user with feedback on the impression of the text on the reader (i.e., does it sound friendly, professional, ...?) and to suggest alternative wordings for achieving a specific style. Of course, such systems can only work on existing text. Accordingly, these systems require users to have sufficient writing skills to create (and edit) a text base. Their benefit for users with low writing skills is, therefore, limited. Additionally, adaptive behavior, like automatically modifying the written text, incurs the risk that low-literate users do

¹*People with learning difficulties* is the self-chosen term to describe people with intellectual disabilities or learning disabilities.

not carefully check whether or not the changes express the originally intended meaning. Moreover, dynamic, complex interfaces can easily overtax users with IDD.

Low-literate users with CCN may use symbol-based systems from *Augmentative and Alternative Communication* (AAC) to express themselves. These systems – also called *talkers* – are primarily designed for active participation in direct communication. For users with (at least basic) literacy skills, systems with literacy support provide basic writing support functions. However, the writing support focuses on formulating individual sentences as part of a conversation – not on writing a coherent text. Additionally, the systems do not fully exploit the potential of NLP (Waller, 2019). Particularly for the German language, where complex morphology and free word order complicate the generation of meaningful, correct suggestions, profound linguistic support through NLP is missing. For example, users must select word endings themselves or manually correct incorrect system suggestions. The use of advanced talkers is typically trained as part of long-standing communication strategies. Accordingly, these systems are often costly, time-consuming to learn, and not accessible to users outside the AAC domain.

Accessible writing systems designed for users with IDD – like traditional text-writing systems – presuppose a certain level of writing and computer skills. Users start their text on a blank page and are then supported by adaptive word prediction, grammar, and spelling aid in an accessible way. In addition, various functions facilitate orientation in the text (for example, through color coding for word types, parts of sentences, or punctuation marks) and flexible read-aloud functions. Many of the accessibility features aim to compensate for handicaps such as visual impairment or motor difficulties. Systematic step-by-step support for sentence formulation and text construction for users with low writing skills is not at hand.

To the best of our knowledge, there is no easy-to-use LS writing system for low-literate users with IDD and/or CCN that offers linguistic support beyond the phrasing of simple, partly personal sentences, let alone a system capable of teaching the concepts of written text production. In addition to writing correct, complete sentences, a holistic writing aid should promote the creation of text coherence and comprehensibility through audience design and help users express their personal narratives. There is a need for adequate technical support in the form of a computer-assisted LS writing tool that remedies reading/writing deficits by supporting users during the process from *message conceptualization* to *sentence realization* and is designed to meet the specific requirements of low-literate people with IDD and/or CCN in terms of usability design. Providing technical assistance to barrier-free participation in all social spheres (Hirschberg and Lindmeier, 2013) – including technical assistance for active participation in written discourse – is an integral part of inclusion.

1.3 Rationale

We aim to develop an assistive LS writing system for the spectrum of low literate people with IDD and/or CCN (hereafter referred to as the *target group*, or simply *the users*), which uses linguistic processing and interactive user guidance to compensate for a deficiency of grammatical knowledge, ensures syntactic correctness and promotes text comprehensibility for the readership. The system's linguistic scope should be appropriate for the target group and offer constructions users might like to use to express themselves beyond needs-based communication. The system's design should meet our target group's specific needs, i.e., it should be accessible, suitable for users with low literacy and computer skills, and support users with low writing skills to compose their own LS texts.

The following research objectives would facilitate the achievement of this aim:

- Analyze LS in respect to text-generation by its target group.
- Develop and implement concepts to support users at their individual proficiency level in writing.
- Translate concepts from text-production theory into intuitive on-demand dialogues.
- Design an accessible interface and supportive interaction patterns suitable for the target users.
- Evaluate the writing system with users of the target group.

This leads to our research questions:

- **RQ1:** What linguistic constructions might LS readers like to use in a writing tool?
- **RQ2:** Which individual (NLP) support can help our target users to write correct, understandable, coherent text at their personal level of proficiency?
- **RQ3:** Can we transform concepts from text-production theory into intuitive on-demand dialogues?
- **RQ4:** Can we design an accessible interface and supportive interaction patterns that do not overtax the user?

1.4 Scope and Limitations

For this thesis, we developed an assistive writing system for our target group called *EasyTalk* to answer our research questions. To tailor the system to the needs of our specific user group, we applied the principles of *human-centered design* (HCD) to the development process. The HCD process for software systems consists of four iterative phases: Analyzing the context of use, defining the requirements, designing and developing the system, and evaluating it from the users' perspective.

We base the specification of the context of use and the initial specification of user requirements on an analysis of LS with a focus on text generation and the current state of research on writing systems for low-literate users with IDD and/or CCN. The research questions overarch the user requirements, which include criteria for the system's usability for the target group and the definition of an appropriate linguistic scope for the system. The ideal basis for exploring what constructions LS readers might like to use in a writing tool would be a text base created by the target audience. In the absence of such a text base, we define the scope using a corpus study of authentic LS texts that incorporates findings from a recent study, *the LeiSA project*, which tested the difficulty of various syntactic constructions within and beyond the core LS rules with LS readers. The result is an extended variant of LS, ELS (see Harbusch and Steinmetz (2022)), which defines the grammar of our NLG component.

In addition to the core functions of text writing in *EasyTalk*, we have developed a concept for a teaching mode in *EasyTalk*, called *EasyText*, which aims to teach users the concepts of text production for an audience in the form of a digital writer's workshop. *EasyText* extends the scope of research by addressing how to use *EasyTalk* to teach the target group text production for an audience. In this work, we present the concept of *EasyText* but note that the practical implementation and evaluation of the teaching mode is beyond the scope of this thesis.

According to the HCD process, we matured the system in iterative development cycles, combined with purposeful evaluations of specific aspects. We aimed to fulfill the criteria of *user-sensitive inclusive design* by Newell et al. (2011) and *disability-aware software engineering* by Nganji and Nggada (2011) such as the involvement of the target group in each development step. However, this was not feasible for us due to organizational overheads, limited access to the target group due to COVID-19 health precautions, and challenges in collecting feedback from the target group on more abstract elements of system development. Therefore, we employed substitute users in the form of expert groups from the fields of CCN, LS, and IT, as well as second language (L2) learners of the German language to evaluate specific aspects. In a final case study, members of the target audience tested the system in free writing sessions. In this study, we used eye-tracking

to obtain additional objective information about the target audience's interaction with the system.

We are aware that a successful application of a writing system for the target group, due to their highly individual needs and abilities, involves several additional factors that we cannot cover in the scope of this work. Depending on the user, there may be a need for alternative options to control the system, such as eye control or *scanning*, and interface customizations for users with visual impairments. However, the labor-intensive development of such features exceeds the scope of this work, which we delimit to developing a prototypical system to answer our research questions.

1.5 Significance

Being able to participate independently in written discourse and to express oneself in writing is a significant component of integration and autonomous participation in social and economic life – especially in the digital age of information. Providing a writing system that enables low-literate LS readers to become LS writers facilitates inclusion and participation. Additionally, exploring our research questions provides insights into how the potential of linguistic processing by computer in modern writing systems for the target group can be further exploited.

1.6 Structure

This work is structured as follows: In Chapter 2 we consider LS in terms of its readership and social functions, describe its target audience, and examine its linguistic scope in terms of text generation by its target group. In turn, Chapter 3 is dedicated to the state of the art of writing systems for our target audience. In Chapter 4, we describe the HCD process that frames the development of *EasyTalk* in response to our research questions and relate the HCD process phases to our research. In Chapter 5, we present our writing system, *EasyTalk*, and detail on the range of its writing support functions on different levels of the text creation process. Chapter 6 is dedicated to system evaluation conducted with different techniques and user groups. In Chapter 7, we discuss our results with respect to our research questions. Chapter 8 illustrates desirable future work. Finally, we conclude the thesis in Chapter 9.

Chapter 2

(Extended) Leichte Sprache

In this chapter, we first introduce Leichte Sprache in its historical and societal context in Section 2.1 and define its target audience and its functions in Section 2.2. In Section 2.3 we detail on the LS rules with respect to text generation before defining an extended version of LS in Section 2.4.

2.1 Introduction to Leichte Sprache

Controlled/Simplified natural languages, like Basic English (Ogden, 1930), have long been a topic of great interest (see Kuhn (2014) for a broad survey). Simple/easy languages designed to provide accessible information emerged during the plain language movement of the 2000s (cf. Easy-to-Read English), which aimed to produce easy-to-understand texts for people with intellectual disabilities or learning difficulties (Maaß and Bredel, 2016, p. 60), who often have low literacy skills (Light et al., 2019). In this context, Leichte Sprache, a simplified variety of German, was developed to meet this demand. The explicit goal of providing accessible information as an instrument for inclusion distinguishes LS from other simplified varieties of the German language, like *Einfache Sprache* ‘Easy Language’. (For a detailed distinction between *Einfache Sprache* and LS, see Maaß and Bredel (2016, Chapter 14.1). On the linguistic level, for example, *Einfache Sprache* allows more complexity than LS – such as the use of subordinate clauses.) Table 2.1 lists the major milestones in the history of LS and locates

Table 2.1: Milestones in LS history in the course of the inclusion movement since the 1990s based on Maaß and Bredel (2016, Chapter 2), Inclusion Europe (2022), Netzwerk Leichte Sprache (2022), and Mensch zuerst (2022).

1994	The Basic Law for the Federal Republic of Germany is amended so no one may be disadvantaged on the basis of a disability ¹ .
1998	The first Easy-to-Read guidelines are published by <i>Inclusion Europe</i> ² .
2001	The association <i>Mensch zuerst - Netzwerk People First Deutschland e.V.</i> ³ is founded and advocates for the right for easy-to-read information as a crucial instrument for people with learning difficulties to be able to represent their own interests.
2002	The Easy-to-Read Logo to label accessible materials is created by <i>Inclusion Europe</i> .
2002	The <i>German Act on Equal Opportunities for People with Disabilities Act</i> ⁴ demands equal participation in social life and a self-determined lifestyle for people with disabilities.
2006	The <i>Convention on the Rights of Persons with Disabilities</i> ⁵ . (CRPD) of the United Nations is drafted. Article 2 of the CRPD explicitly includes the demand for plain-language.
2006	The <i>Netzwerk Leichte Sprache</i> ⁶ is founded and focuses on the design barrier-free information for people with learning difficulties.
2009	The CRPD comes into force in Germany.
2009	The first guidelines for LS are published by <i>Inclusion Europe</i> ⁷ .
2011	In Germany, the CRPD is followed by the <i>National Action Plan 2011</i> ⁸ , <i>NAP 2011</i> , a strategy aiming for “equal participation in political, social, economic and cultural life, equal opportunities in education, vocational integration and [...] giving all people the opportunity to have a selfdetermined place in a accessible society.” (Federal Ministry of Labour and Social Affairs, 2012).
2011	The requirements of the NAP are concretized in the <i>Barrier-free Information Technology Ordinance, BITV 2.0</i> ⁹ that contains a set of rules for accessible texts and enshrines the right to LS information in law, making LS the means of choice for providing accessible information in text form in Germany.
2013	The <i>German Federal Ministry of Labour and Social Affairs (BMAS)</i> publishes a LS rulebook that largely adopts the rules developed by the <i>Netzwerk Leichte Sprache</i> ¹⁰ based on practical experiences.
2014	The requirement of the <i>BITV 2.0</i> for federal authorities to make LS information available in the first locations by March 2014 creates the first legal obligation to provide information in LS.



Figure 2.1: The European Easy-to-Read Logo: *Inclusion Europe*. More information at www.inclusion-europe.eu/easy-to-read. In Germany, the logo is used to brand LS texts.

its roots in the inclusion movement (for more detailed information on the history of LS, see, for example, Maaß and Bredel (2016, Chapter 2)). LS is an instrument to achieve the goal of an inclusive society in which people with learning difficulties can participate in all aspects in a self-determined way. As depicted in Table 2.1, the right for accessible LS information was enshrined in German law in 2011, and the first legal obligation for the federal agencies to provide (basic) LS information – such as information about the content and navigation of federal agency internet and intranet sites – defined in the BITV2.0 (2011) became active in March 2014.

¹Basic Law for the Federal Republic of Germany Article 3, Sentence 3:
www.gesetze-im-internet.de/gg/art_3.html

²*Inclusion Europe*: www.inclusion-europe.eu

³*Mensch zuerst*: <http://www.menschzuerst.de/>

⁴German Act on Equal Opportunities for People with Disabilities:
www.gesetze-im-internet.de/bgg/BJNR146800002.html

⁵United Nations - Convention on the Rights of Persons with Disabilities:
www.un.org/development/desa/disabilities/convention-on-the-rights-of-persons-with-disabilities.html

⁶*Netzwerk Leichte Sprache*: www.leichte-sprache.org

⁷“Information for all: European standards for making information easy to read and understand”: www.inclusion-europe.eu/easy-to-read-standards-guidelines/

⁸The National Action Plan of the Federal Government for the implementation of the UN CRPD on the website of the German Federal Ministry of Labour and Social Affairs:
<https://www.bmas.de/DE/Service/Publikationen/a740-aktionsplan-bundesregierung.html>

⁹BITV 2.0: www.gesetze-im-internet.de/bitv_2_0/BJNR184300011.html

¹⁰LS rules of *Netzwerk Leichte Sprache*: www.leichte-sprache.org/leichte-sprache/die-regeln/

Today, LS information has become available in more and more places. Examples include:

- election programs in LS¹¹,
- apps of the authorities (for instance, the emergency information and news app *NINA*¹², or the *Corona App*¹³),
- and news and entertainment in LS (see, e.g., *NDR Leicht*¹⁴, *TAZ Leicht*¹⁵ or *Einfach Heidelberg*¹⁶).

2.2 The target audience(s) of Leichte Sprache

The increasing availability of LS texts and their consistent branding with the LS logo shown in Figure 2.1 certainly increases awareness of LS among people beyond the original target group of people with learning difficulties and their communities. Needless to say, easy-to-understand information in the form of LS texts can benefit many people. In addition to people with learning difficulties, the LS rulebook of Netzwerk Leichte Sprache (2013) (adapted by BMAS (2013)¹⁷, cf. Table 2.1) names for example people who do not speak German very well (e.g., second language learners of German) or people who cannot read very well (i.e., people with low literacy skills) as LS target audiences.

¹¹See, for example, the election programs for the 2021 federal election of the four major parties of Germany (all accessed on 09.12.2022):

SPD: www.spd.de/fileadmin/Dokumente/Beschluesse/Programm/Wahlprogramm_SPD_2021_LS_barrierefrei.pdf,

CDU / CSU: www.csu.de/common/download/KM_Broschuere_Leichte_Sprache_BTW_2021_Ansicht.pdf,

Bündnis 90 / Die Grünen: cms.gruene.de/uploads/documents/Wahlprogramm_DIE_GRUENEN_Bundestagswahl2021_Leichte-Sprache.pdf,

FDP: www.fdp.de/sites/default/files/2021-08/FDP_BTW2021_KWP_leichteSprache.pdf,

¹²The federal catastrophe alert app *NINA*: www.bbk.bund.de/DE/Warnung-Vorsorge/Warn-App-NINA/warn-app-nina_node.html

¹³*Corona App*: www.coronawarn.app

¹⁴*NDR Leicht*: www.ndr.de/fernsehen/barrierefreie_angebote/leichte_sprache/Nachrichten-in-Leichter-Sprache,nachrichtenleichtesprache100.html

¹⁵*TAZ Leicht*: taz.de/leicht/!p5097//

¹⁶*Einfach Heidelberg*: www.einfach-heidelberg.de

¹⁷Due to publication by a federal ministry, the LS rule book of BMAS (2013) has become more widely known than its base, the rule book of Netzwerk Leichte Sprache (2013). In this paper, we only cite the original rule book of Netzwerk Leichte Sprache (2013). The LS rules are largely identical in both books.

The three main LS rulebooks, Inclusion Europe (2009), BITV2.0 (2011) and Netzwerk Leichte Sprache (2013), were developed with the purpose of creating accessible texts for people with intellectual disabilities and are accordingly based on their communication needs (Maaß and Bredel, 2016, p.19). One rule of the Netzwerk Leichte Sprache (2013) particularly highlights the original target group of LS: Every LS text should be checked for comprehensibility by people with learning difficulties. Considering the potential target audiences of LS, it is important to recognize the diverse needs and skills of potential LS readers. For example, low-literate native speakers of German have different conditions than L2 learners with low German skills who can read and write fluently in their native language. The skills and needs of people with disabilities may vary significantly for each person, depending on the type, age of acquisition, and degree of the disability. Recent research examines the reception of LS by different target groups. (For example, Gutermuth (2020) examine the reception of LS by different reader groups, like migrants, senior citizens, people with intellectual disabilities or people affected by prelingual deafness, Pottmann (2019) considers LS in the context of teaching German as a second language, and Burgherr (2021) specifically investigates how LS texts can benefit refugees.)

Maaß and Bredel (2016) define the primary target audience of LS as people who are dependent on LS for all or most types of written texts, who have no or only very limited access to texts in standard or technical German. Secondary addressees of LS may use LS offerings but are not dependent on them¹⁸. As a prerequisite for the comprehension of texts in LS, Maaß and Bredel (2016, p.40) define at least basal reading ability and at least basal knowledge of German; L2 learners of German need to have reached at least *CEFR*¹⁹ level A1 in order to be able to read LS texts successfully. Maaß and Bredel (2016) state, that for people who do not (yet) meet these requirements, reading LS texts is presumably an acquisition task.

As mentioned before, LS is an instrument for inclusion used to provide barrier-free access to easy to understand information. Three important societal functions of LS are: participation, learning, and building bridges to standard German (Maaß and Bredel, 2016, pp.56,57). LS texts are usually written by authors proficient in standard German and then evaluated for ease of comprehension by individuals with intellectual disabilities or learning difficulties. This implies that LS readers are commonly consumers of LS, whose participation in creating LS texts is limited. Accordingly, the functions of LS mentioned above also refer foremost to reading – but not to writing texts. An essential part of participation is express-

¹⁸see, for example, Maaß and Bredel (2016, Chapter 5) for a more detailed description of the potential target audiences of LS

¹⁹The Common European Framework of Reference for Languages, *CEFR*: www.coe.int/en/web/common-european-framework-reference-languages

ing oneself and being an active participant with a personal voice in the discourse. Therefore, we aim to help LS readers become LS writers themselves – exploring the possible extent of the participation, learning, and bridging functions of LS more deeply at the same time. One factor preventing the target users from producing texts themselves may be the lack of technical support during the process from *message conceptualization* to *sentence realization*. To this end, we are developing a system for computer-assisted writing in LS, *EasyTalk*. *EasyTalk* implements the grammar of LS in a Natural Language Generator. Following the repetitive structures and basic lexical patterns of LS sentences, *EasyTalk* aims to support users in writing freely while practicing general linguistic concepts at the same time. It is an open question to explore, whether a writing support system for writing in LS can complement the learning and bridging function of LS.

In the following section, we detail on the LS rules relevant for language generation by computer.

2.3 Leichte Sprache rules for text generation

LS is defined by rules which concern the medial and visual layout of texts as well as their linguistic and verbal representation. The rules for LS were originally derived from practical experience (Maaß and Bredel, 2016, p. 60). The three main rule books (Netzwerk Leichte Sprache, 2013; Inclusion Europe, 2009; BITV2.0, 2011) – in particular the rules shaping the linguistic form of LS texts – have been the subject of previous scientific investigation (cf. Maaß et al. (2014); Lieske and Siegel (2014); Löffler (2015); Zurstrassen (2015); Maaß and Bredel (2016); Bock (2019); Nüssli (2019); Pottmann (2019); Hansen-Schirra and Maaß (2020)).

Maaß and Bredel (2016) provide a detailed comparison of all rules of the three major rule books. Based on this comparison, we extracted the rules that are relevant for defining the linguistic scope of LS in the context of text generation by the user. Table 2.2 shows the results. Examples of rules that we omitted in Table 2.2 are rules concerning the text design, like “Add subheadings.” (Netzwerk Leichte Sprache, 2013; Inclusion Europe, 2009; BITV2.0, 2011), or the visual layout of LS texts, like “Start a new line for each sentence.” (Netzwerk Leichte Sprache, 2013; Inclusion Europe, 2009; BITV2.0, 2011).

Many rules concern the vocabulary (e.g., “Use short words.” (Rule 2.1), “Use easy words.” (Rule 3.1), “Do not use technical terms.” (Rule 3.3), or “Do not use abbreviations.” (Rule 2.6)). Other rules have a significant impact on the grammatical scope of LS – “Do not use commas.” (Rule 1.1), for instance, restricts LS texts to the use of main clauses.

Table 2.2: Relevant LS rules for text generation extracted from the rule comparison table of Maaß and Bredel (2016, pp. 109 - 116).

	Rules grouped by category	Source(s)		
		Netzwerk LS	BITV 2.0	Inclusion Europe
1	Interpunctuation and special characters			
1.1	Do not use commas.	X	X	-
1.2	Do not use special characters.	X	X	X
1.3	Do not use 'high numbers' or percentages.	X	X	-
2	Morphology			
2.1	Use short words.	X	X	X
2.2	Avoid composita or separate them with '-' .	X	X	X
2.3	Do not use simple past tense.	-	X	-
2.4	Do not use conjunctive mood.	X	-	X
2.5	Do not use genitive case.	X	-	X
2.6	Do not use abbreviations.	X	X	X
2.7	Do not use passive voice.	X	X	X
3	Lexic			
3.1	Use easy words.	X	X	X
3.2	Provide explanations for difficult words.	X	X	X
3.3	Do not use technical terms.	X	X	X
4	Syntax			
4.1	Use short sentences.	X	X	X
4.2	Do not use inversions.	X	-	X
4.3	Use clear sentence structure. Only use the connectors <i>oder</i> 'or', <i>wenn</i> 'if', <i>weil</i> 'because', <i>und</i> 'and', <i>aber</i> 'but' at the beginning of a sentence	X	-	-
4.4	Avoid questions in the text. You may use questions in headlines.	X	-	-
5	Semantics			
5.1	Avoid negations.	X	X	X
5.2	Only one proposition / thought per sentence.	X	X	-
6	Text			
6.1	Use consistent wording	X	X	X
6.2	Avoid pronominal resumption	-	-	X

In summary, the LS rules in Table 2.2 result in the avoidance of complex structures, for example, the use of:

- more than one statement per sentence;
- punctuation other than: “.”, “?”, “!”, “:”;
- complex clauses;
- inversions;
- the genitive case;
- the passive voice;
- the subjunctive mood; and
- the simple past tense.

In other words, only main clauses are included in LS. In main declarative clauses, the canonical word order is subject–verb–object (SVO). All sentences should be phrased in the active voice, indicative mood, and present or present perfect tense.

The primary LS audience of people with cognitive impairments or learning difficulties is very heterogenous, and the available authentic text data by which to identify the range of constructions LS readers naturally use in the process of formulating an idea are sparse. Usually, LS texts are written by authors proficient in standard German. Contrary to the recommendation in *Netzwerk Leichte Sprache* (2013) and *Inclusion Europe* (2009), ease of comprehension is not always tested by members of the target readership. The levels of difficulty of a range of syntactic constructions were systematically evaluated with LS readers as part of the recent *LeiSA project* (Bock, 2019). That study identified a number of constructions that were evaluated as being easy to comprehend but which fell beyond the definition of LS. Inspired by the finding of the *LeiSA study* that the majority of easily understandable LS texts do not strictly adhere to LS rules, Harbusch and Steinmetz (2022) have explored possibilities for extending those rules to include the syntactic constructions that LS readers are likely to use in putting their thoughts into words. As a result, Harbusch and Steinmetz (2022) defined *Extended Leichte Sprache* (ELS), a native extension of LS based on observations from a corpus study. We will introduce ELS in the following.

2.4 Extended Leichte Sprache

In the previous section, we introduced the core LS constructions: Only main clauses are included in LS. In main declarative clauses, the canonical word order is subject–verb–object. All sentences should be phrased in the active voice, indicative mood, and present or present perfect tense.

The comprehension difficulty of a range of individual syntactic LS constructions was systematically evaluated with LS readers as part of the recent *LeiSA study* (Bock, 2019). Through well-established comprehension tests constructions – not only within but also beyond the scope of LS – were classified according to the error rates measured in experiments with participants with intellectual disabilities and low literacy skills. That study identified a number of constructions that were evaluated as being easy to comprehend but which fell beyond the definition of LS. For example, the comprehensibility of negations – which are forbidden by all major LS rulebooks (cf. Table 2.2, Rule 5.1) – was closely examined. The *LeiSA study* found that negations using *nicht* ‘not’ (e.g., *Der Bus fährt nicht*. ‘The bus isn’t running.’) are easy to comprehend for the target group ($0 \leq \text{error rate} \leq 5.9\%$) whereas negations using *weder–noch* ‘neither–nor’ (*Das Paket ist weder groß noch schwer*. ‘The package is neither big nor heavy.’) show a medium error rate ($12.5\% < \text{error rate} \leq 37.5\%$).

Inspired by the findings of the *LeiSA study*, we sought to identify the full range of syntactic structures that LS readers are likely to use when putting their thoughts into words. Therefore we analyzed authentic LS texts. These results were previously published in Harbusch and Steinmetz (2022). To highlight the joint nature of the treebank study published in Harbusch and Steinmetz (2022), we cite both authors instead of writing ‘we’ when referring to the corpus study.

Syntactic constructions that do not adhere to the LS rules are not hard to find in published LS documents. This suggests the advisability of inspecting a broad collection of LS texts and analyzing the constructions found therein. As not all of the constructions found in LS texts were evaluated with LS readers in the *LeiSA* project, Harbusch and Steinmetz (2022) employed a treebank study as an alternative to the evaluation of example sentences by LS readers and compared the frequencies of constructions in LS to those in spoken and written standard German. Harbusch and Steinmetz (2022) argue that constructions with high frequencies in spoken German are easy to produce due to the time-pressured nature of speech production. In a written text, the author is able to embellish the text in revision cycles, replacing simple constructions with more complex ones. Thus, the written corpus served as a baseline. The appearance of a given construction in spoken language with frequency higher than or equal to its appearance in written language is indicative of an easy-to-understand or unavoidable construction. Conversely, higher frequencies in written text imply difficult constructions employed under

the non-time-critical conditions of revision and editing.

For the quantification of syntactic constructions in standard German, Harbusch and Steinmetz (2022) used TüBa-D/S (also called VERBMOBIL), a treebank of spoken German, and TüBa-D/Z, a treebank of written German. In the VERBMOBIL project (see, e.g., Stegmann et al., 2000 or Wahlster, 2000), more than 400 spontaneously produced spoken dialogues (concerning appointment scheduling) were transliterated and syntactically annotated. The Tübinger Baumbank des Deutschen/Zeitungskorpus (TüBa-D/Z; see, e.g., Telljohann et al., 2009) is a syntactically annotated corpus based on the German newspaper *die Tageszeitung (taz)*. To obtain the frequencies of syntactic constructions of LS, Harbusch and Steinmetz (2022) created a syntactically annotated corpus of LS texts, the *Leichte Sprache Treebank (LST)*.

For LST, Harbusch and Steinmetz (2022) assembled 245 LS texts with more than 300,000 word forms a variety of sources freely available on the internet, spanning the years 2018–2021. To build a representative data set of LS texts of sufficient variety, they selected a broad spectrum of institutions, authors, and validators: according to the credits, at least 153 authors, 116 validators, and 53 institutions were involved in the creation of these texts. Each corpus graph in LST provides a feature with detailed source information of the original text. Originally, Harbusch and Steinmetz (2022) planned to use this information to distinguish the following two subcorpora:

- LST-WithP, comprising only those texts that were proofread by LS readers; and
- LST-NoP, including texts without explicitly mentioned LS-reader participation.

As the two subcorpora do not differ with respect to the number of violations of LS constraints, Harbusch and Steinmetz (2022) omitted the planned step of investigating differences between their construction frequencies.

Harbusch and Steinmetz (2022) identified that a majority of the texts follow the LS rules. These texts not only deal with simple topics (e.g., fairytales) but concern many spheres of life, including patient decrees, voters' rights, financial matters, and laws of succession. The implication is that the conformity of a text to LS rules does not depend on the complexity (or simplicity) of the topic but on whether or not its authors are aware of *best practices* (e.g., rephrasing “if” sentences as questions).

Harbusch and Steinmetz (2022) studied the frequencies of a wide range of syntactic constructions in LST. Constructions that fulfill one of the following conditions were added to ELS:

- **Constructions covered by the LeiSA study:**

According to the LeiSA study, the construction has at most medium-level difficulty for LS readers and it occurs frequently in LST.

- **Constructions NOT covered by the LeiSA study:**

The frequency of the construction in LST compares favorably to its frequency in spoken German and the construction cannot be easily transformed into a pure LS construction.

Harbusch and Steinmetz (2022) categorize the syntactic phenomena according to their level of construction complexity: (1) word-related, (2) phrase-related, and (3) clause type-related constructions. Within each level, they refer to the phenomena examined by the *LeiSA study*. Additionally, they discuss typical simple constructions that are beyond the scope of pure LS and were not evaluated in the LeiSA study. This way, they systematically assessed the whole range of simple syntactic constructions.

In the following, we summarize their most important findings (for further details and the exact frequencies, see the full paper of Harbusch and Steinmetz (2022)).

Negations. Negation is forbidden in LS; however, it is difficult to avoid completely (Bock, 2017). Harbusch and Steinmetz (2022) analyzed the frequencies of several negation words. As mentioned, *nicht* ‘not’ is easy for LS readers to understand according to the LeiSA study and accordingly added to ELS. According to the high frequency of its occurrence in LST (comparable to that in VERBMOBIL), *kein_{inflected}* is also added to prevent forcing a reformulation with *nicht*. All other constructions including negation (e.g., *nie(mals)* ‘never’, *niemand_{inflected}*, or *weder – noch* ‘neither-nor’) are very infrequent in both LST and VERBMOBIL.

Prepositions. According to the rules, prepositions with an agglutinated definite article (APPRART) should not occur in LS (Table 2.2, Rule 2.6); nevertheless, this construction is frequently used in LS text. Half of the cases occur in the idiomatic prepositional phrase (PP) *zum Beispiel* ‘for example’, which should not be abbreviated in LS. In practice, the use of APPRARTPPRART makes sentences shorter. Moreover, using *zu dem Beispiel* for *z.B.* ‘e.g.’ sounds odd in German. Harbusch and Steinmetz (2022) therefore add APPRART to ELS.

Under LS rules, all but genitive-taking prepositions (absent in LST) are included; therefore, Harbusch and Steinmetz (2022) do not suggest further extensions for prepositions.

Nouns. Harbusch and Steinmetz (2022) investigated the morphological features *number=plural* and *case=genitive*, which were studied in the LeiSA project, in LST and in TüBa-D/Z. Plural forms are of low difficulty. They occur slightly

more frequently in LS than in TüBa-D/Z (no data are available for VERBMOBIL). This construction is included in LS. The frequency of genitives, forbidden in LS, is very low; therefore, Harbusch and Steinmetz (2022) do not include genitives in ELS, although they note that such constructions are often used in idiomatic expressions, e.g., for the names of institutions.

Pronouns. The LeiSA study excluded pronouns from its investigations. Harbusch and Steinmetz (2022) provide a detailed analysis of the frequencies of all German pronoun types (e.g., substituting demonstrative pronouns, substituting indefinite pronouns, substituting possessive pronouns, or reflexive personal pronouns). Their comparison of pronoun frequencies places LST somewhere between the written and spoken corpora. However, the high numbers of pronouns is unexpected for the following reasons: The three major LS rule sets Netzwerk Leichte Sprache (2013), Inclusion Europe (2009), and BITV2.0 (2011) insist on consistent naming, i.e., using exactly the same word for the same thing/person throughout a text (cf. Table 2.2, Rule 6.1); additionally, the LS rule set of Inclusion Europe (2009) forbids pronominal resumption, favoring nominal resumption (Table 2.2, Rule 6.2). In particular, many occurrences of substituting indefinite pronouns that characterize abstract referents (e.g., *man* ‘one’, *jemand*_{inflected} ‘somebody’, *etwas* ‘something’, and *alle*_{inflected} ‘all’) resemble those in written text. Due to their frequency, to shorten the resulting sentences, and to enable the use of abstract referents to circumvent passive constructions, Harbusch and Steinmetz (2022) include all pronouns in ELS.

Verb forms. Before taking a look at phrases and clauses, we summarize the findings of Harbusch and Steinmetz (2022) on verb forms.

Auxiliaries and modals, which are included in LS, can dominate non-finite verb forms to build the present perfect tense and specify the modality of other verbs, respectively. Contrary to expectation (that finite verb forms would be most frequent in LST), the frequency of finite verb forms in LST is between the frequencies in VERBMOBIL and TüBa-D/Z. This finding can be attributed to the presence of a higher number of present perfect tense constructions in LST than in the non-LS written corpus. The frequency of infinitives (V.INF and V.IZU, i.e., infinite with *zu* ‘to’, as in *Man hat so versucht*_{FIN} *Corona aufzuhalten*_{IZU} ‘One has tried to stop Corona in this manner’) is similar in LST and VERBMOBILBIL. However, the number of modals that are likely to entail a lexical verb is higher in LST (as a variety of written text) than in VERBMOBIL. As mentioned above, specific lexical verbs can dominate non-finite verbs with POS = V.IZU. The verb lemma *versuchen* ‘to try’ occurs 16 times in LST, 26 in VERBMOBIL, and 472 in TüBa-D/Z. The frequency in LST resembles that in VERBMOBIL, and the infinitive construction is similar to that of modals. We therefore add complement-taking verbs that belong to the restricted LS vocabulary to ELS. However, constructions with *um zu* ‘for the purpose of/in order that’ (KOU) are not added to

ELS. The frequency of KOU1 is 50% lower in LST than in VERBMOBIL. Moreover, the construction can straightforwardly be segmented into: ‘for the (following) purpose/thereby/!...!’ and a main-clause construction, without obstructing the train of thought. For example, the sentence *Es braucht_{FIN} Zeit um_{KOU1} sich zu erholen_{INF}* ‘It takes time to recover’ is divided into the following three lines: *Es braucht_{FIN} Zeit. // Damit/Bis(:) // Sie erholen_{FIN} sich (wieder).*

The preterite tense, passive voice, and subjunctive mood are forbidden in LS. The preterite occurs very infrequently in LST, and most of the 800 cases pertain to auxiliaries and modals. The few lexical verb cases can be replaced by present perfect tense forms (included in LS) without the meaning being changed. To support this argument, we searched VERBMOBIL for preterite forms of the three most frequent lexical verb lemmas according to Kempen and Harbusch (2019). For *sehen* ‘to see’ and *machen* ‘to make’, no incidences were found. Forms matching *ging/-st/-t/-et/-en* of *gehen* ‘to go’ occur 31 times; however, only half of the matches are related to preterite forms. These all occur in the idiomatic phrase *das ging schnell/gut* ‘that went quickly/well’. The other cases match subjunctive mood forms referring to potential time slots/connections/etc. As all verb forms of auxiliaries and modals appear with high frequency, and sentences are shortened by the use of the preterite (instead of present perfect) tense, we include the preterite for auxiliary and modal verbs as finite form in ELS. The frequency of passive constructions is low in all three corpora. Given that such constructions are of medium difficulty according to the LeiSA study, and that it is often hard to find a simple reformulation in the active voice that conveys the same nuance of meaning, Harbusch and Steinmetz (2022) include passive constructions in ELS.

Harbusch and Steinmetz (2022) noticed that nearly all subjunctive mood cases in LST are forms of auxiliaries or modals (e.g., *wären* ‘would be’, and *möchten* ‘would like’). As these are frequent word forms in German, Harbusch and Steinmetz (2022) include the subjunctive forms of auxiliary and modal verbs in finite form in ELS. Constructions with double objects are of medium difficulty, according to the LeiSA study. Such constructions are equally rare in all three corpora examined in this study. However, common verbs like *geben* ‘to give (somebody something)’ qualify for this construction. As these verbs are included in LS, no extension is required.

Phrase and sentence complexity. A treebank search allows for very detailed syntactic specifications. However, it is necessary to keep in mind that the three inspected treebanks are differently encoded in this respect. For example, a noun phrase (NP), i.e., *cat=NX* in VERBMOBIL and TüBa-D/Z, covers constituents that differ from the nodes at the ends of edges labelled *subj*, *obja*, *pn*, etc. in PARZU (Sennrich et al., 2013). In an *NX*, adverbs can be seized. In PARZU, adverbs – if not in the frontfield – belong to the sentential level. For example, the phrase *auch schon viele barriere-freie Gebäude* ‘also already many barrier-free

buildings’, occurring in the midfield of a clause, is assigned to three constituents on the sentence level in PARZU, whereas in the two other treebanks the phrase is assigned to one NX node. There is no simple solution to this problem without manually inspecting all adverbs in LST. Hence, not entirely accurately, but in line with the characterization in the LeiSA study, Harbusch and Steinmetz (2022) define phrase complexity by a simplistic dichotomy with respect to length: They distinguish *short* (up to three words) from *long* (more than four words) constituents. This concept translates to sentence-complexity calculations: Harbusch and Steinmetz (2022) define *short* sentences as containing up to nine words (assuming that these occur in no more than three constituents, each comprising no more than three words) and *long* sentences as containing more than nine words. (Note that punctuation is excluded from consideration here because any phrase-level search refers to an inner node of a corpus graph. In each of the three treebanks, punctuation is governed by the root node – cf. the example trees provided in the supplementary material. The discontinuous positions of punctuation symbols in the surface word order can be accessed by the list of word forms in the TIGER-XML format.) For example, *Bis zu diesem Gehalt zahlt man Beiträge* ‘Up to this income, you pay contributions’ is a short sentence, but *Die Krankenkasse oder die Agentur für Arbeit zahlt die Beiträge für sie.* ‘The health insurance or the employment agency pays the contributions for you’ is a long one. On average, this simple distinction identifies complicated constructions: in our examples, a sentential modifier PP and an NP-modifying PP in an NP coordination, respectively.

In LS, no explicit length restriction for phrase complexity is stated. However, the LeiSA study qualifies as easy only those phrases with no more than three elements; therefore, no new constructions are added to ELS.

Next, we summarize the investigations of Harbusch and Steinmetz (2022) of sentence complexity. They quantified the frequency of the canonical SVO word order in main declarative clauses. They restricted the search to frontfields at the onset of a sentence to abstract away from elided constituents. (*Forward Conjunction Reduction* and/or *Gapping* are the only ellipsis phenomena that can elide the left periphery, and only in the second conjunct of a coordinated sentence; see Ross, 1967). Moreover, according to Temperley (2019), the most complex constructions occur at the onset of a sentence because more mental capacity is available here.

In line with LS, the LeiSA study allows only SVO word order; even mild forms of topicalization were judged to be very difficult for LS readers. Unexpectedly, the frequency of the canonical word order is found to be very similar in all three treebanks; constructions with the SVO word order comprise only half of all constructions in the LS corpus, i.e., the other half are very complicated for the target readers. Clearly, the standard German writers of LS texts adhere to the standard rules of German discourse structure. In mild cases, a one-word constituent occupies the frontfield. However, Harbusch and Steinmetz (2022) also found complex

frontfield fillers, such as conditional clauses, sentential subjects, and objects. For example, in sentence (1), the fronted object is interpreted as the argument of the finite verb form *wollen* until *barriere-frei* occurs (i.e., this is a garden-path sentence). Given the difficulties arising from simple deviations from the canonical word order, no deviations are included in ELS.

- (1) Das_{ACC} wollen_{FIN} wir in NRW barriere-frei machen_{INF}.
 This want we in NRW barrier-free make
 ‘These things we want to make barrier-free in NRW.’

For the canonical word ordering, the average length of the frontfield is longer in LST and TüBa-D/Z, the two written varieties – as expected. The subjects in VERBMOBIL are extremely short due to the use of personal pronouns in dialogue. (Note that VERBMOBIL often has discourse markers, self-repairs, etc. at the onset of a sentence. Therefore, the total numbers for LST and VERBMOBIL diverge more here than in other tables.)

For subordinate clauses, Harbusch and Steinmetz (2022) distinguish between relative clauses and subordinate clauses starting with a subordinating conjunction (KOUS). Both constructions are forbidden in LS. According to the LeiSA study, subject-relative clauses are of low difficulty for LS readers, whereas any other type of relative clause (i.e., a relativizer in the dative or accusative case due to its grammatical function or dominating preposition, respectively) is extremely difficult. Surprisingly, both types occur with approximately equal frequency in LST and TüBa-D/Z. As expected, the relative clauses in LST are considerably shorter than those in TüBa-D/Z. Given their frequency and the LeiSA evaluation, Harbusch and Steinmetz (2022) opt to include subject-relative clauses in ELS (although this adds a verb final (VF) construction to the included word order patterns); however, they suggest that such clauses are not discontinuous and that they should be short.

All types of subordinate clauses are forbidden in LS. In the LeiSA study, subordination with the temporal conjunctions ‘while/after’ is considered of low difficulty. Although ‘while’ does not occur in LST, *nachdem* ‘after’ is used five times. The conditional conjunction *wenn/falls* ‘if’ is by far the most frequent. Other conjunctions used in LST (with their respective frequencies) include: *als* ‘when’ (8); *bevor* ‘before’ (18); *bis* ‘until’ (37); *damit* ‘so that’ (5); *indem* ‘by’ (2); *nachdem* ‘after’ (5); *ob/obwohl* ‘whether’ (204); *seit* ‘since’ (1); *solange* ‘as long as’ (3); and *weil* ‘because’ (180).

Rather than licensing specific subordinate clause types, Harbusch and Steinmetz (2022) suggest adding all subordinating conjunctions to ELS. However, the conjunction and the sentence should be presented in two separate consecutive lines, and the trailing sentence should have main clause word order. The same construction works with the synonymous subordinating conjunction *weil*.

In VERBMOBIL, the subordinating conjunction *weil* ‘because’ occurs in half of the cases with SVO order (Kempen and Harbusch, 2016). (This phenomenon is widely studied as the *weil*-V2 phenomenon in spoken standard German; see Reis (2013) for a broad overview.) This strategy also covers the highly difficult construction of dependent *that* clauses, which occur in LST slightly less often than in VERBMOBIL. This construction can straightforwardly be avoided by replacing *that* by a colon. The content of the *that* clause is presented as a main clause in the canonical SVO order. Thus, Harbusch and Steinmetz (2022) do not include this construction in ELS.

Coordination and ellipsis. Coordination and ellipsis are of medium difficulty, according to the LeiSA study. However, the tested examples are very simple. Coordinations in LST consist of very long lists. Often, formal definitions are replaced by long lists of examples, probably to avoid the use of overcomplicated technical terms. Therefore, although Harbusch and Steinmetz (2022) do not add any new constructions, they recommend using only short lists of coordinated constituents.

The same holds for ellipsis. The use of ellipsis in spoken and written text (see corpus studies into VERBMOBIL (Harbusch and Kempen, 2009) and the TIGER treebank, another syntactically annotated German newspaper corpus (Harbusch and Kempen, 2007)) goes beyond the scope of very limited Forward Conjunction Reduction restricted to the subject, which prevails in LST, and which was the only type of ellipsis evaluated in the LeiSA study. As this construction is judged to be of medium difficulty, and it can be circumvented by explicitly repeating or pronominalizing the subject, Harbusch and Steinmetz (2022) choose not to add it to ELS.

Table 2.3: Summary of constructions in *Extended Leichte Sprache* that extend *Leichte Sprache*.

Phenomenon	Usage
Negation	Restricted to <i>nicht</i> as verb modifier and <i>kein_{inflected}</i> as determiner or substituting pronoun
POS=APPRART	Agglutination of the definite article following the preposition
POS=PRO	noun phrase filled with a substituting pronoun
Preterite tense	Restricted to auxiliaries and modals
Subjunctive mood	Restricted to auxiliaries and modals
Passive voice	Restricted to forms of <i>werden</i>
Complement-taking verb	Restricted to verb lemmas in the vocabulary of LS that dominate a nonfinite verb form with the word ordering S V _{finite} O V _{nonfinite}
Subordination	<i>dass</i> ‘that’ is replaced by a colon; any other conjunction obtains a separate line; the constituents of the subordinate clause are presented with V2-word order in a new line
Relative clause	Restricted to subject-relative clauses, however, not discontinuous

As a result of their corpus study, Harbusch and Steinmetz (2022) defined *Extended Leichte Sprache*. As LS shows more similarities to proximity language (spoken German) than to conceptual written language (Maaß and Bredel, 2016, p.58), ELS is a natural extension of LS. Constructions that show low error rates in the *LeiSA* comprehensibility tests are included by ELS. Table 2.3 sums up all extensions included in ELS that Harbusch and Steinmetz (2022) propose.

Next, we present the state of the art in writing support tools for low-literate users with IDD and/or CCN.

Chapter 3

Writing support systems for low-literate users with intellectual or developmental disability and/or complex communication needs

In this chapter, we discuss state of the art of writing support systems for low-literate users with IDD and/or CCN for the German language. Section 3.1 focuses on symbol-based writing systems from the area of Augmentative and Alternative Communication to examine the linguistic scope and literacy support for low-literate AAC users. In turn, Section 3.2 discusses text-based writing support. Finally, Section 3.4 addresses systems for teaching text-production.

3.1 Symbol-based writing systems

Augmentative and Alternative Communication¹ offers a wide range of support to people with complex communication needs – an essential part of inclusion, i.e., self-determined participation and expression.

A common instrument in AAC is the use of graphic symbols (e.g., pictures, drawings, pictograms) as visual representation of a word or idea. Different symbol collections fit the diverse needs of AAC users. For example, people with visual impairments may use high-contrast symbols. Two symbol sets widely used in Germany are *METACOM*² and the *Picture Communication Symbols (PCS)*³ (Johnson,

¹For detailed information on AAC, see the *International Society for Augmentative and Alternative Communication* (www.isaac-online.org).

²METACOM website: www.metacom-symbole.de

³PCS website: goboardmaker.com/pages/picture-communication-symbols



(a) A page of a symbol book.
(LIFEtool, 2022)



(b) A symbol board.
(ARASAAC, 2022)

Figure 3.1: Two examples of non-technical aids for symbol-based AAC: A communication book with PCS-symbols (a) and a communication board with ARASAAC-symbols (b) composed of multiple symbol strings that can be reassembled into new contents.

1987). Both are commercially licensed. The ARASAAC symbol set (ARASAAC, 2022) is a free-to-use alternative distributed under the Creative Commons License (BY-NC-SA).

There are various communication tools for symbol-based AAC. Non-technical aids include symbol boards or communication books, as seen in Figure 3.1. Technical AAC solutions for symbol-based communication range from simple concatenation of symbols for needs-based, functional communication with small vocabularies to complex (commercial) systems (see, e.g., Lancioni et al. (2019) for a detailed survey). Currently, technical AAC solutions are evolving rapidly and are increasingly available on mainstream devices like smartphones and tablets (Light and McNaughton, 2012; Ascari et al., 2018; Farzana et al., 2021).

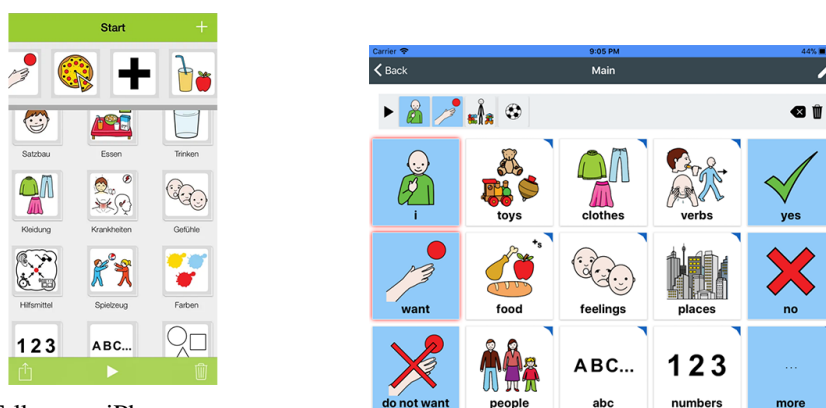
Here, we focus on AAC systems that go beyond functional communication usable for users with low literacy skills supporting the expression of personal thoughts in the context of social closeness and sharing information (cf. Light et al. (2019) and Waller (2019)). Whenever possible, we review systems for our target language German, where a rich morphology and relatively free word order complicate the generation of useful and grammatically correct suggestions⁴.

All popular free AAC apps for German allow users to access a large customizable vocabulary of symbols. However, they do not provide well-founded linguistic support for sentence construction and/or text production (cf. *LetMeTalk*⁵ and *SymboTalk*⁶, shown in Figure 3.2). It is important to mention that these systems aim

⁴For an up-to-date overview of communication aids for German see the database by *Rehamedia* (rehamedia.de/kernprodukte/kommunikationshilfen/) and the database by the German foundation *barrierefrei kommunizieren!* (www.barrierefrei-kommunizieren.de/datenbank/)

⁵*LetMeTalk*: apps.apple.com/us/app/letmetalk/id919990138

⁶*SymboTalk*: www.symbotalk.com



(a) LetMeTalk on an iPhone
(LetMeTalk App Store page
(AppNotize UG, 2022),
29.01.2022)

(b) SymboTalk on an iPad
(SymboTalk website
(Elram, 2017), accessed 29.01.2022)

Figure 3.2: Snapshots of two examples of popular free AAC apps for symbol-based communication using ARASAAC symbols: (a) *LetMeTalk* (b) *SymboTalk*.

at direct (face-to-face) communication between conversation partners and are not designed as writing systems.

In comparison, commercial systems for symbol-based AAC offer a significantly larger scope of linguistic support. The commercial systems *Gateway*⁷, *Mind Express*⁸, and *TD Snap Core First*⁹ offer a representative sample of widely provided features in complex, commercial symbol-based AAC systems that go beyond functional, needs-based communication. Primarily these systems enable users to actively take part in real-time conversations. In addition, they aim to help users develop written language skills by enabling them to increase their diversity of expression in terms of both the grammatical scope and the vocabulary. For writing, they provide basic linguistic support such as adaptive word prediction and automatic inflection for simple constituents. The more complex the linguistic variety, the more grammatical knowledge is required to create correct sentences. For instance, the user has to manually select word endings or correct errors caused by incorrect predictions.

In the following, we consider the writing support of these systems in detail before summarizing their commonalities in writing support and interface design in Section 3.1.

⁷Gateway: www.gatewaytolanguageandlearning.com

⁸Mind Express: www.jabbla.com

⁹Snap Core First: www.tobiidynavox.com/pages/snap-core-first

Gateway



Figure 3.3: Foto of *Gateway* running on an iPad (Elram, 2022).

Gateway (Bruno, 2020) arranges the symbols in a grid layout, which is typical for symbol-based AAC aids (see Figure 3.3). By grouping words by context and category, the symbol pages in *Gateway* are designed to provide efficient access to the vocabulary. Consistent application of the *Fitzgerald Encoding*¹⁰ to the words, word categories, folders (solid colored with a white border) and buttons (solid colored with a colored border) helps the user navigate the grid layout and eases the transition to symbol pages of a higher level of complexity. Additionally, the grid layout can be customized to the user's physical and visual needs. For example, the size and arrangement of the grid elements can be customized. For users with spelling skills, *Gateway* offers pagesets containing an alphabetical keyboard, word endings, words and symbol predictions¹¹. *Gateway* aims to enable users to express themselves freely and develop their language skills. For example, it offers its users successive levels of growing vocabulary and evolving syntactic variety (*Gateway to Language and Learning*, 2022). Beginners communicate mainly in 1-2 word sentences of simply concatenated symbols. They use nouns, verbs and a selection of adjectives and adverbs in their basic form. Then they learn to combine the symbols into multi-word sentences. In this step, users can learn from to use word endings, articles, and prepositions to form syntactically complex sentences.

¹⁰The *Fitzgerald key format* is a color-coding system for word categories commonly used in the AAC context. For example, people are represented in yellow, things in orange and places in purple. Verbs are colored green, descriptive words (adjectives, adverbs, and time words) blue and little words (prepositions, articles, and conjunctions).

¹¹For details on the alphabet pages and their configuration, see: de.tobiidynavox.com/blogs/aufgezeichnete-webinare/compass-mit-gateway, last accessed 12.01.2022

Here, automatic adjustments of the word forms associated with the symbols can be made by Gateway or manually selected by the user. With increasing skill level, the grammatical correctness of the sentences plays an increasing role. One feature available to form syntactically correct sentences is *dynamic morphology*¹². It includes for example:

- personal pronouns automatically inflecting subsequent verb forms,
- offering participles after the choice of an auxiliary verb (e.g., *ich bin gegangen* ‘I have walked’),
- offering infinitive forms after modal verbs (e.g., *ich muss gehen* ‘I must go’)
- automatic transformation of object pronouns (e.g., *ich gebe es ihm* ‘I give it to him’),
- and displaying grid elements for manual selection of other grammar features needed in the context of the sentence (e.g., the choice of comparative and superlative forms or a more precise choice of morphological endings).

A detailed video demonstration of the system¹³ shows that with increasing complexity of syntactic and morphological variety, users have to make more manual selections in order to express themselves in a grammatically correct way.

¹²For a demo video of Gateway’s *dynamic morphology*-feature see www.gatewaytolanguageandlearning.com/resources/video-clips.html: ‘Learning Dynamic Morphology’, last accessed 12.01.2022

¹³Recording of webinar about gateway: de.tobiidynavox.com/blogs/aufgezeichnete-webinare/compass-mit-gateway, accessed 17.02.2022.

Mind Express

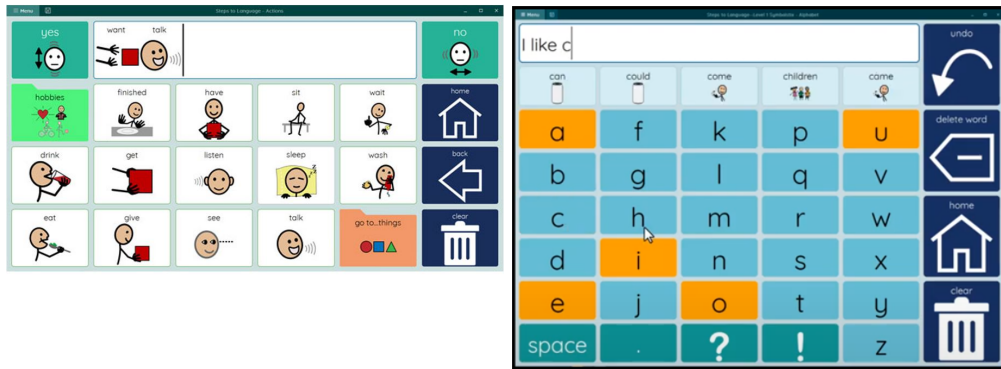
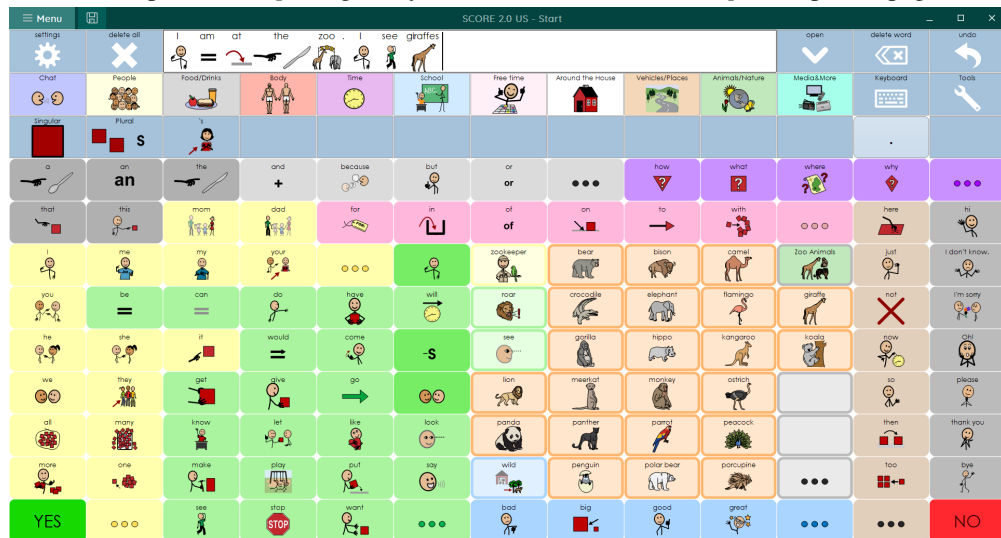
(a) A simple *Mind Express* grid layout.(b) A *Mind Express* alphabet page.(c) A complex *Mind Express* grid layout.

Figure 3.4: Layouts of different styles and complexity for *Mind Express*: Snapshot (a) shows a simple grid layout (Jabbla, 2022a), (b) an alphabet page offering letters and symbols to access words (Jabbla, 2022b), and (c) a complex grid layouts with color coding for different word types and categories (Jabbla, 2022c).

Mind Express (Jabbla, 2022a), like *Gateway*, offers several ways to design grid layouts. For example, Figure 3.4a shows a simple grid layout and Figure 3.4c a complex layout for advanced users. Users with basic spelling skills can to use a mixture of letters and symbols to choose the words, (partially) hide the symbols or use so called *Alphabet pages* (Jabbla, 2022b) on demand. Alphabet pages, see Figure 3.4b, are operated with an alphabetical keyboard and show next word predictions together with symbols.

To enable users to write grammatically correct sentences, *Mind Express* dynamically displays grammar functions depending on the current part-of-speech. These grammar functions include, for example:

- automatic inflection for simple subjects,
- *tense buttons* (cf. Jabbla (2022b)) to change the tense of a verb or sentence to past or future tense,
- *dynamic labels* (cf. Jabbla (2022)) to automatically adjust word forms displayed in combination with symbols (alternatively the adjusted word form is only shown in the sentence box),
- and *grammar popup pages* (cf. Jabbla (2022a)) allow to access additional word endings and word forms when selecting a verb for advanced users who write sentences of a complexity for which *Mind Express* no longer automatically provides correct word forms.

TD Snap Core First

In addition to expanding the core and fringe vocabulary, *TD Snap Core First* (tobii dynavox, 2018) wants to help users develop their literacy skills. For this purpose, *TD Snap Core First* provides some similar grammar features as *Mind Express* and *Gateway*. For example, it automatically adjusts verbs to match a previously selected pronoun and offers participles after forms of *sein* ‘to be’ have been selected. Entering the *Wortwandler* ‘word forms’ enables the user to access different forms (tense, number, etc.) of the last word in the message window located at the top of the choice grid. The user manually selects the desired word form to update the word in the message window accordingly.



(a) A simple *Snap Core First* symbol grid.

(b) The ABC area of *Snap Core First*.

Figure 3.5: Two layouts of *TD Snap Core First* (tobii dynavox, 2022): a simple symbol grid (a) and an ABC area (b)

Similar to *Mind Express* and *Gateway*, *TD Snap Core First* provides keyboard pages operated using a mixture of symbols and letters. Additionally, *TD Snap Core First* offers a so-called *ABC vocabulary area* (see Figure 3.5b). It sorts words by their first and second letter and provides phonetic feedback when one or more letters are selected¹⁴.

Users who can already read and write without symbol support or are in the process of transitioning to written language, *TD Snap Text*¹⁵ offers a text-based communication solution that builds on the design of *TD Snap Core First*.

Conclusion: Writing support in symbol-based AAC systems

The systems described above aim to enable users to communicate efficiently in real time for active participation in conversations. Additionally, the systems aim to help users develop their written language skills by enabling them to increase the diversity of expression (grammatical scope and vocabulary).

For writing, the systems provide basic linguistic support such as adaptive word prediction and automatic inflection for simple constituents. The more complex the linguistic variety, the more grammatical knowledge is required on the side of the user. For instance, the user has to manually select word endings or correct errors caused by incorrect predictions.

As shown in Figures 3.3, 3.4 and 3.5, the systems are organized in customizable grid layouts of varying complexity, suitable for different access methods like scanning¹⁶ (i.e., the system iterates sequentially through all options until the user instructs the system to stop and make a selection), eye control or touch. Grid cells can contain symbols, word and letters as well as function-buttons (i.e., ‘delete last word form’, ‘enter settings menu’). Accordingly, activating a grid cell can select a word or lead to another grid page containing more words of a certain category or access grammatical functions. Users with basic spelling skills can use a mixture of letters and symbols to choose the words, as seen in Figure 3.4 (b). Considering the complexity of the grid layouts, it is important to mention, that these systems presented are foremost used in the context of year-long learning and communication strategies. They need to be individually taught and practiced.

As representative layout examples the snapshots in Figures 3.4, 3.3, and 3.5 show that the current sentence is commonly located prominently at the top. Previ-

¹⁴A demonstration of the *ABC vocabulary area* of *TD Snap Core First* can be found here: www.youtube.com/watch?v=YlGs8TJx2uE&ab_channel=TobiidynavoxDeutschland, accessed 17.02.2022.

¹⁵*TD Snap Text* homepage: de.tobiidynavox.com/pages/td-snap-text, accessed 19.01.2022

¹⁶For an introduction to different scanning techniques see praacticalaac.org/tag/scanning/

ous sentences or a textual context are only visible to advanced users (see the complex *Mind Express* keyboard in Figure 3.4 (c), where two subsequent sentences are displayed in the white sentence box). Of course, participating in (spoken) communication in real time has different requirements than writing texts. We suppose this to be one reason why the writing support of all systems is by design focused on the sentence level. Features for writing coherent texts that take audience design into account are not available. Neither do the currently available AAC systems exploit the full potential of linguistic processing by computer (Waller, 2019).

In the following, we consider the writing support that text-based writing systems for low-literate users with IDD offer on the sentence level and beyond.

3.2 Text-based writing support

Writing instruction with appropriate technology has a positive impact for people with IDD (Smith et al., 2020). Modern text editors implement barrier-free access by features like read-aloud functionality. The database of the German foundation *barrierefrei kommunizieren!* lists systems for users with disabilities: standalone systems like *Kurzweil3000*, *Penfriend*, and *MULTiTEXT*, and next-word predictors like *WoDy*, *EMU* and *FTB-TippFixx* that can be integrated with MS Word and other text editors to support the user.

Text-based writing support for our target group suits users with a certain amount of computer skills who can write short sentences in a (simplified or customized) text editor. A variety of visual highlighting and color encodings (e.g., color keys for different word types, parts of a sentence or punctuation symbols) aid orientation in the text. Flexible read-aloud functions reproduce the written text letter by letter, word by word or sentence by sentence (with or without punctuation marks), respectively, providing memory support and spelling assistance. On demand, all systems employ grammar checkers. Adaptive word predictions (partially for customizable vocabulary) are usually offered in the form of lists that can be controlled using hotkeys for quick selection. However, all systems present the users with an empty page. The textual organization of the train of thoughts and aspects of audience design are not supported in these tools.

There is an increasing demand for language support through linguistic processing by computer. An increasing variety of writing-support systems based on natural language generation attract attention. For a discussion of these approaches, see, e.g., Dale and Viethen (2021) or Gatt and Kraemer (2018), who illustrate the potential of NLG systems. In response to the increasing demand for easy-to-read content, tools using different natural language processing techniques to support

authors in creating LS texts have also been developed¹⁷. The *LangueTool*¹⁸, for instance, uses a rule-based approach to check texts for violations of LS rules. For automatic text simplification in German, Suter et al. (2016) present a rule-based approach, and Ebling et al. (2022) introduce a sentence-based neural machine translation approach. *SummAI*¹⁹ uses artificial intelligence to assist authors in translating existing texts into LS and creating new LS information. However, all of these approaches aim at authors who are proficient in Standard German. They are not designed to help our user group write in LS.

3.3 Deep learning-based writing support

Deep learning methods based on large language models for writing support have recently attracted considerable attention (for survey, see, e.g., Otter et al. (2021)). *OpenAI*'s²⁰ *GPT-3* (Generative Pre-trained Transformer 3) stands out in particular (see, e.g., Dale (2021)). It can create, summarize, simplify or translate texts. The model succeeds in generating content that is sometimes very difficult to distinguish from human-authored texts. In the following, we will discuss to what extent this technique can be used (at this point in time) to support our target group in writing.

GPT-3 is a deep learning model for neural networks that can be trained with virtually any type of text. It uses an approach called sequence-to-sequence (seq2seq) machine translation, where an input sequence (text) is used as a prompt to generate an output sequence (text). The model creates new text based on the training data's structure, style, and patterns. A detailed discussion of *GPT-3* is out of scope for this work, and we refer interested readers to Floridi and Chiriatti (2020) instead. Instead, we want to discuss specific aspects that present challenges for using *GPT-3* – representative of similar methods – for writing support for our target group. Therefore, we consider the aspects training data, input, and output in the following.

Training data *GPT-3* is already able to (re-)produce certain text styles, presumably by mimicking similar patterns from the training data. It is an open question how/whether one could get *GPT-3* to consistently produce “LS-style” texts. Certain individual prompts may be able to achieve this, but the reaction of *GPT-3* to these prompts can vary, making the results inconsistent and not always sticking

¹⁷For approaches for the English language, see, e.g., paperswithcode.com/task/text-simplification.

¹⁸*LangueTool* for LS: languagetool.org/en/leichtesprache/

¹⁹*SummAI* website: www.summ-ai.com

²⁰*OpenAI* website: www.openai.com

to the rules of LS. Alternatively, one could re-train or fine-tune GPT-3 on LS text to make it stick only to LS text. However, the amount of computing needed for this is immense, and as of now, there is no sufficiently large database of LS text available. (It is estimated that there are currently about 1000 German text pairs of standard texts and their equivalents in LS (Dejffal and Horst, 2021)). Using existing LS texts to create a database, additional aspects have to be considered: As described in Sections 2.2, 2.4, text in LS is generally produced by authors proficient in standard German (Maaß and Bredel, 2016, p.25) and then evaluated for ease of comprehension by people with intellectual disabilities or learning difficulties (BITV2.0, 2011; Netzwerk Leichte Sprache, 2013). Although the participation of the target group to check texts for their comprehensibility is anchored in the LS rules (Netzwerk Leichte Sprache, 2013) and is perceived as an integral aspect of the work of plain language (Dejffal and Horst, 2021), this step is often omitted in practice. Many LS texts are not checked by the target group. (For almost half of the LS texts used to create the *Leichte Sprache Treebank* introduced in Section 2.4 it is unknown if the target group was involved in testing the texts for comprehensibility.) In addition, the rules for LS have been developed on the basis of practical experience. It has not yet been fully researched which of the constructions licensed by the LS rules are really easy to understand for the LS target group (cf. Bock (2019)). Moreover, many texts do not follow the LS rules (cf. the LS corpus study by Harbusch and Steinmetz (2022) presented in Section 2.4).

Besides the linguistic realization of the texts, the content of the training data also influences the language model. LS texts are often of an explanatory nature due to the information function of LS (cf. Section 2.1). They contain an above-average number of difficult words to be explained in the text. Accordingly, we'd expect GPT-3 generated LS text to contain many technical terms. This could be prevented by modifying the cost functions used during training of large language models such that they discourage the use of difficult words. However, avoiding difficult words *in general* may not be a good idea either, since some difficult words may in fact be known to the individual user and should therefore actively be used by the system, as not doing so could limit the wealth of a user's expression. Therefore, it would in fact be desirable to customize the language model for the individual user, which is something that these models do not support at this point in time.

Presumably, due to the target group's low literacy skills, a large text base that provides information about how the target group would express themselves in writing does not (yet) exist to the best of our knowledge. Because our target group is very heterogeneous, many factors depend on the individual user's personal use of language. (The strengths of GPT-3 lie, in contrast, in generating texts in a particular style for a specific domain, such as technical blog posts. Here the model can infer a general style from the training data).

Assuming that feeding the language model with LS-text would be the most effective way to train the model to create LS-text, appropriate training data for a large language model would first need to be created. Ideally, the training data would be created in collaboration with the target group in pursuit of the guiding principle of the *Convention on the Rights of Persons with Disabilities*: “Nothing About Us Without Us” (Charlton, 1998). Suggestions by the system that are automatically extracted from given LS text might not be perceived as helpful – some might even be perceived as irritating, let alone unintentionally patronizing. This aspect should also be evaluated in collaboration with the LS target audience. To circumvent the lack of suitable training data, for the time being, one could advise the model only to use short, simple sentences and easy words for text realization (e.g., at CEFR level A1) This approach is used by Reiter et al. (2009) and Tintarev et al. (2016) in their English approaches for NLG storytelling for language-impaired AAC users.

Input The system needs information in the form of a textual input sequence to generate content. The more precisely the user defines the prompt for the system, the more targeted the output. Parameters are, for example, text type (e.g., story or report), domain, topic, target audience, type of content (e.g., information or entertainment), tonality, and mood. Some of these parameters could be preset. At the moment, however, the user still needs to instruct the system on what to write about in a precise manner. This is a challenging, possibly overwhelming task for users with low literacy skills and cognitive challenges. We would expect the text input of our low-literate user group, who are ranging between Alpha-Levels 1 and 4 (cf. Section 2.2), to be too short and erroneous to create meaningful output in a targeted manner.

In their approach using an NLG system to create a personal narrative for answering the question “How was school today?”, Tintarev et al. (2016) try to circumvent this challenge by using a data-to-sequence approach. Their system automatically creates personal stories for low-literate AAC users from sensor data and other media like voice recordings collected as the school day progresses (an approach not feasible for our user group). To filter the data, the user selects the events of the day they want to talk about (e.g., Math class). As a challenge Tintarev et al. (2016) name enabling the user to adequately control the system in terms of both content and expression, as they aim to “assist the user in communication, not replace him or her with an NLG communicator”. This leads to our next point – the generated output.

Output Even given precise instructions, there are certain limitations of the GPT-3 technology. Ultimately, its statements are based on a fortuitous alignment of

textual statistics drawn from the training data (Dale, 2021). The generated results are arbitrary, not personal or specific. This is a problem when it comes to expressing a personal narrative or opinion, because the system does not know the user. Even if the system was trained with additional personal data like in the approach of Tintarev et al. (2016), the system cannot know what the user thinks about a certain topic (e.g., a news article) or wants to express at the moment. The user could either run a prompt until the system generates a pleasing result by chance or adjust the generated text to fit what they wanted to express. To do this, however, they have to read and understand the generated text, decide to discard or keep it, and – in the latter case – decide which elements to change to achieve the desired message. This cognitively demanding task is not suitable for our user group who – due to low literacy – cannot carefully check whether or not the text expresses the intended meaning nor edit it to their liking. Users with intellectual difficulties, memory or cognitive flexibility limitations, and spontaneous decision-making problems are additionally challenged.

In summary, there is no recent NLG-based writing aid tailored to the needs of our target users to the best of our knowledge (cf. the pioneering approach by Demasco and McCoy (1992) aiming to assist users with severe motor impairments). In addition to the lack of suitable training data, deep learning approaches based large language models like GPT-3 present challenges in the process of creating input as well as reading and editing the output and are (not yet) suitable for our user group. Existing textual writing support systems for the target group require a certain amount of writing skills. Additionally, they confront the user with a blank page. The textual organization of the train of thought is not supported. Next, we look at current approaches to teaching text production for the target group addressing the latter aspects.

3.4 Systems for teaching text-production

In the school context, the method of the *Schreibwerkstatt/Schreibkonferenz* ‘writing work-shop’ (see, e.g., Reichardt et al. (2014) for a broad survey) is widely applied. The pupils learn to introduce all protagonists of a story in a way that allows the reader to identify them as the story progresses. The appropriate use of elements of text coherence, discourse structure, and audience design is taught. On the sentence-formulation level, students are instructed to integrate sets of short, choppy sentences into longer, more effective ones (cf. Nordquist (2018) for an online introduction to sentence-combining techniques; Ney (1980), for the history, and Saddler and Preschern (2007), for the context in school). Beside computer systems for the above-mentioned topics, there is a wide range of NLG systems for automatic text production, such as parameterized interactive storytelling by Lukin

and Walker (2019), or interactive story modeling using recurrent neural networks by Fortuin et al. (2018). However, none of these systems is available in German. Moreover, there is no straight-forward way to equip any of these systems with an interface suiting the target group.

Chapter 4

Developing EasyTalk using human-centered design

This chapter is dedicated to the human-centered design process framing the design and development of our writing system. In Section 4.1, we first introduce the HCD process in terms of *EasyTalk*. The following sections are dedicated to the individual phases of human-centered design: specifying the context of use (Section 4.2), specifying the requirements (Section 4.3), creating design solutions (Section 4.4), and evaluating design solutions (Section 4.5).

4.1 Introduction to human-centered design

EasyTalk aims to support our target audience in the complex process of text writing in a simple, easy-to-understand way. In developing the system for the spectrum of low-literate users with IDD and/or CCN, it is particularly important to focus on the needs of this very heterogeneous target group. A well-established method for tailoring a system for a specific user group is *human-centered design* – for the current international definition see the norm *ISO 9241-210*¹. (In the to the context of software development it is often called *user-centered design* (UCD) – introduced by Norman and Draper (1986)).

The key principle of HCD is to put the user at the center of a system’s design and development process in order to make the design easy to use and fit the users’ needs. In addition to the users’ needs, the interests and perspectives of all stakeholders are considered in the development process.

¹ISO 9241-210:2019 – *Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems*: www.iso.org/standard/77520.html

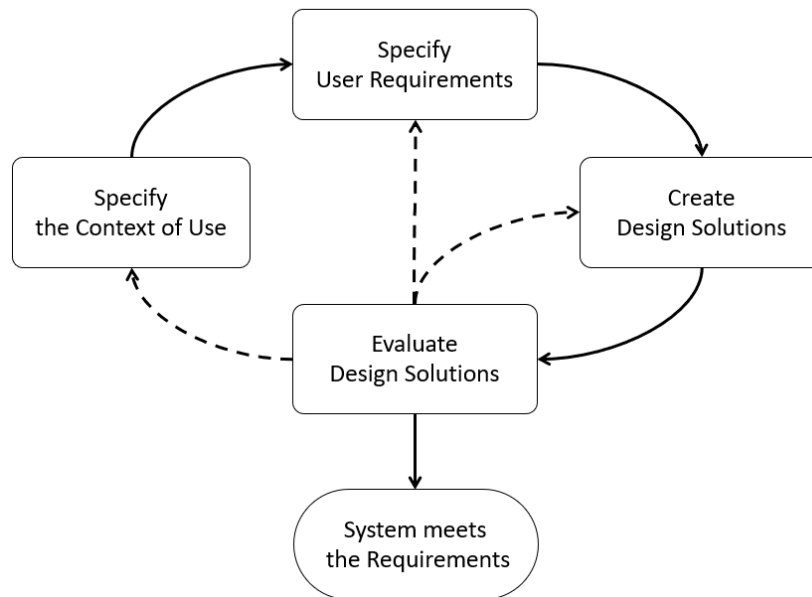


Figure 4.1: The iterative process of human-centered design according to ISO 9241-210. The dotted arrows indicate to iterate where appropriate.

HCD is generally described in four iterative phases:

1. Specifying the context of use
2. Specifying the user requirements
3. Creating design solutions
4. Evaluating designs

As depicted in Figure 4.1, the phases of HCD are interconnected and to be iterated where appropriate until the system meets the user requirements.

In the following, we will detail on each phase of the HCD process in terms of *EasyTalk* individually. As the HCD process frames the development of the assistive writing system in response to our research questions, each phase relates to particular sections of our research.

4.2 Specification of the context of use

In order to develop a system focusing on the needs of its users, it is essential to understand who the users are, their objectives, and in which context they will use

the system. For the specification of the context of use, according to *ISO 9241-210*, the following four factors must be considered:

1. Who are the users? What other stakeholders are there?
2. What are the characteristics of the users/the user group?
3. In which social, cultural, and technical environment will the system be used?
4. What are the goals and objectives of the users?

We address these factors in the following.

The target users of our system belong to the spectrum of people with IDD and/or CCN with low literacy skills. Historically, they are the primary target group of LS (cf. Section 2.1). In recent years, extensive research has focused on this target group and their needs in terms of easy-to-understand language. The description of our users according to the above mentioned HCD criteria is based on these findings, which we detailed on in Sections 2.1 and 2.2.

Our user group is very heterogeneous in various factors. For example, the target group contains users of different age groups (e.g., teenagers or adults), life situations (e.g., school students or employees), and German skills (native speakers or people with migration background). In addition, the personal abilities and limitations of each user may be individually pronounced. To provide a more tangible representation of the target audience, we created the personas in Table 4.1, which represent five examples of prototypical users, each with individual prerequisites and goals.

The environment of use may vary from user to user, as the personas in Table 4.1 illustrate. While some users already use technical aids – possibly adapted to their personal conditions, such as motor or visual impairments – other users may have limited computer skills or not use technical aids at all. Accordingly, the individual conditions for using the system may vary from user to user.

All potential users of our writing system have a common goal: their writing skills are low, but they want to express themselves in writing. We aim to support them in telling their personal stories beyond needs-oriented communication and the repetition of short, familiar phrases (e.g., “I am OK. And you?”). In order to provide useful writing support to our users at their personal level of proficiency, we need to accurately assess the literacy skills of our target group. According to the state of research depicted in Section 2.2, people who can successfully read and understand LS texts have acquired at least basal reading ability and at least basal knowledge of German. For L2 learners of German this translates to at least CEFR level A1.

Table 4.1: Five personas representing examples of prototypical users of *EasyTalk*. Out of consideration for the vulnerable target group, we refrain from using images, as these may reinforce prejudices.

An LS reader in special education	
Personal data	Sandra, female, 24 years old, autism spectrum disorder.
Profession	Sandra is training as a gardener and landscaper through a special school.
Writing context	She learned to read and write in school but it has always been difficult for her. She has a small laptop with the free writing software <i>Open Office</i> installed.
Pain points	When she opens the writing software she is confronted the white page and finds it hard to formulate sentences.
Motivation	At the facility, there is LS newspaper which she likes to read. She would also like to write an article or a reader's letter for the newspaper.
A low-literate talker user with CCN	
Personal data	Seku, male, 19 years old, Non verbal.
Profession	Seku lives and works at a facility for assisted living.
Writing context	He uses a symbol-based <i>talker</i> to communicate. He already knows many communication symbols and has basic spelling skills. He can write short sentences with his talker.
Pain points	Writing with his talker is very slow. Additionally, Seku has a hard time choosing the right word endings when writing. Often this is too tedious for him, so he uses most words in their base form.
Motivation	He wants to have a tool that helps him better with writing so he can write messages for his friends and family.
An LS validator	
Personal data	Heinrich, male, 52 years old, intellectual disability.
Profession	Heinrich is employed in a workshop for people with disabilities.
Writing context	He can read a little but writes only one or two words sentences. So far, he is not familiar with computers.
Pain points	Writing by hand is very difficult for him because of his motoric handicaps.
Motivation	In his spare time, Heinrich is a validator in an LS validation group. Most times, he likes the texts but would express some things differently. Additionally, he would like to create his own texts.
A low-literate worker on the regular labor market	
Personal data	Christian, male, 48 years old, learning disability.
Profession	Christian is a production helper who assembles and packages orders.
Writing context	He has never really learned to write. He can read a little bit – for example the safety regulations at work written in LS. His two children have already moved out and email him regularly. He reads their emails on the tablet computer that his children gave him.
Pain points	Because he finds it difficult to write, he usually replies to his children's mails very briefly with "OK" or "All good. You?". He feels like he cannot adequately express himself in writing.
Motivation	He would like to answer his children's mails properly.
Pupil in special education	
Personal data	Ben, male, 15 years old, learning difficulties.
Profession	Ben is a pupil.
Writing context	He has little experience with computers and feels like regular writing software is too complicated for him. He became familiar with some AAC symbols in school, although he does rely on them to communicate.
Pain points	Generally, he enjoys going to school. However, writing is very difficult for him. This often frustrates him. Whenever he wants to write something, he needs help in formulating the sentences, which he does word by word with the assistance of his teachers and parents.
Motivation	Ben wants to write on his own without always having to ask for help.

To characterize the writing skills of our target users, we use the so-called *Alpha-Levels*, a method to categorize the writing skills of illiterate and low-literate persons introduced by Grotlüschen and Riekmann (2011, 2012). For writing, the Alpha-Levels are defined as follows:

- Persons at *Alpha-Level 1* can write at *letter level*. They are able to associate phonemes (sounds) with graphemes (letters). (This could be, for example, users with CCN who communicate mainly with symbols but know the initial letters of words.)
- At *Alpha-Level 2*, writers reach the *word level* and can spell short, familiar words.
- At *Alpha-Level 3*, the *sentence level*, writers can form short sentences.
- At *Alpha-Level 4*, individuals reach the *text level* in writing. They can write using familiar, everyday vocabulary so that the meaning of their texts can roughly be understood. However, they still write very slowly and incorrectly at the sentence and text level, making mistakes even in common words and phrases.
- Persons at *Alpha-Level 5* can write complex texts and have acquired morphematic strategy, i.e., the ability to derive words from building blocks (e.g., *die Bestellung* ‘the order’ from *bestellen* ‘to order’).
- At *Alpha-Level 6*, persons can write complex text using cross-word strategy, i.e., the ability to consider complex linguistic aspects when writing sentences and texts, such as part of speech to determine capitalization and punctuation.

Based on the assumption that, presupposing sufficient computer skills, people at Alpha-Level 5 and 6 can already use conventional writing support, we locate the majority of our potential users between *Alpha-Level 1* and *Alpha-Level 4*.

In addition to the target users, there are two other main groups of stakeholders involved in our assistive writing system: persons who help users with the system and readers of the user texts.

The user’s helpers may, for example, be caregivers, teachers, or family members. For this group, the system must be easy to access, set up, maintain, understand, and explain. For users in need of assistance, the helpers are essential to accessing and successfully using the system.

For the readership, text understandability is most important. Contributing factors include grammatical correctness of the texts, text coherence, and the addition of important information – such as the place and time of the event described – that

increase the comprehensibility of the text for the reader, who cannot ask questions as in a real-time conversation.

The considerations described lead to the user requirements defined in the following.

4.3 Specification of the user requirements

Creating an assistive writing system for the target group, two main challenges arise: First, the system must provide users with the adequate linguistic means to express themselves. Second, the system must support the users in the writing process – i.e., the system must take care of factors that are otherwise difficult for the users, such as spelling, grammar, and sentence construction. At the same time, the system must be simple and intuitive. Expressed from the user’s point of view: “As a user, I want a system that makes writing easy so that I can write what I want.”

Since LS was developed for our target audience, we expected the linguistic scope of LS to be appropriate for the target audience. Therefore, we examined the relevant LS rules for text generation in Section 2.3 and analyzed the linguistic scope of LS. We realized, that the LS rules do not license common constructions from spoken German - such as using the simple past for modal verbs instead of the perfect (i.e., using *Ich wollte Eis essen* ‘I wanted to eat ice cream.’ instead of *Ich habe Eis essen gewollt*. ‘I have wanted to eat ice cream’). Based on a corpus study based on authentic LS texts, we set the linguistic scope of *EasyTalk* to ELS (cf. Section 2.4), a slightly extended variant of LS, that includes linguistic constructs beyond the core LS rules that our users are likely to be familiar with and want to use when writing – like the common formulations mentioned above.

Considering the state of the art of writing systems for the target group in Chapter 3, we identified factors that might prevent our target group from using these systems. Being aware of these factors is important in order to design a suitable system for the target group.

In summary, we identified the following reasons for our target group not to use these systems:

- The systems are too complex. Their interfaces and functions overtax the user.
- The systems require too high a level of computer skills for inexperienced users.
- The effort to learn the systems is too high for the users.

- The Alpha-Level required for successful use is too high. In writing-based systems, for example, the user is usually confronted with a blank page and does not receive step-by-step writing support at the sentence or text level.

In the case of symbol-based AAC systems supporting literacy, additional factors contribute to the fact that users, who do not rely on AAC for direct communication, usually do not use AAC systems for writing – such as:

- The systems are inaccessible to individuals unfamiliar with AAC symbol systems.
- AAC communication systems are optimized for the specifics of conversational communication in real-time. While they may support text writing, they do not focus on it.
- Complex AAC (grid) systems require extensive training and have high learning demands.
- Advanced AAC systems are usually costly.

In order to develop the system in line with our research questions, each research question stated in Section 1.3 is translated into a core requirement for the system below. To specify the user requirements in more detail, we add to each core requirement the insights from the context of use described in 4.2 and the considerations above as a list of sub-requirements. (We also include insights from the iterative design/evaluation cycles. These are detailed on in the corresponding sections in Chapter 6).

***EasyTalk* must offer linguistic constructions the user likes to use. (RQ1)**

- *EasyTalk* must offer the linguistic scope of ELS.
- The user must be able to use personal terms.

***EasyTalk* must support the user to write correct, understandable, coherent text at their personal level of proficiency. (RQ2)**

- *EasyTalk* must take care of the grammatical correctness of a sentence.
- *EasyTalk* must offer support at the letter-level, word-level, sentence-level, and text-level.
- *EasyTalk* must support the user in creating text coherence.
- *EasyTalk* must support the user in adding important information for the reader.

***EasyTalk* must transform concepts from text-production theory into intuitive on-demand dialogues. (RQ3)**

- *EasyTalk* must phrase linguistic concepts in an easy-to-understand way.
- *EasyTalk* must guide the user through the text-writing process.
- All text elements in the interface must be short, easy-to-read and easy-to-understand.

***EasyTalk*'s interface must be accessible and its interaction patterns must be supportive, intuitive and easy-to-use. (RQ4)**

- The system must provide good usability.
- The *EasyTalk* interface must be clearly structured.
- On demand, *EasyTalk* must read aloud text elements.
- *EasyTalk* system must provide personal AAC symbols.
- The selection of words must be easy.
- *EasyTalk* system must be usable for users with little computer knowledge.
- *EasyTalk* system must be usable without extensive instruction or training.

We are aware that developing a system for our target group presents additional challenges. As mentioned in Section 4.2, the target users are very heterogeneous in their individual abilities, limitations, and life situations (cf. the personas in Table 4.1). For example, some users may have motor disabilities that require the use of special input methods to control the writing system, such as adapted keyboards or eye control. However, it is essential to first tackle the core use case of *EasyTalk* enabling users with IDD and/or CCN with limited writing skills to write correct, understandable text.

Considering requirements above, we created the design solution described in the following.

4.4 Creation of the design solution

This section presents the basic design ideas of our system, *EasyTalk*, for fast, correct, and reader-centered writing that we developed considering the context of use (cf. Section 4.2) and the user requirements (cf. Section 4.3). Essentially,

the design of *EasyTalk* is inspired by AAC methods (cf. Section 3.1), spoken dialogues, and school teaching techniques (cf. Section 3.4).

To provide support at the letter and word-level and to support users with CCN, we make use of communication symbols. By default, *EasyTalk* presents words and sentences in combination with AAC symbols. This representation of sentences as a string of word-symbol combinations is common in symbol-based AAC (see, for example, the symbol book in Figure 3.1a), the symbol board in Figure 3.1b, or the symbol based communication systems in Section 3.1).

In spoken dialogues, information and context emerge naturally through a back-and-forth of questions and answers from the conversational partners. The following conversation snippet between person A and person B shows this by example:

- (1) A: “What do you want to do?”
 B: “I want to shop.”
 A: “What do you want to buy?”
 B: “Clothes.”
 A: “OK. Where?”
 B: “In the mall.”
 A: “When do you want to go?”
 B: “Tomorrow.”

Similarly, in a widely-applied method from school teaching, the concept of sentence components is taught by asking wh-questions to identify sentence components and their functions (for example, asking ‘Who?’ identifies the subject of a sentence). Inspired by this, the process of sentence writing in *EasyTalk* into is broken down into writing sentence components. Each component is associated with a wh-question indicating its grammatical function. Applying this concept to the example above, the system takes the role of person A who asks conversation partner B for information. Component by component, system and user compose a sentence in a dialog-like manner:

- (2) System: Who? Does? What? Where? When?
 Writer: I want to shop clothes in the mall tomorrow
 Resulting Sentence: ‘I want to shop clothes in the mall tomorrow.’

This process enables the system to maintain the syntactic correctness of the construction the user is typing in so-called *scaffolded writing* (Harbusch and Kempen, 2011). Using a natural-language paraphrase generator implementing the grammar of ELS, *EasyTalk* can suggest correctly inflected word forms, pursue the overall correctness and completeness of the sentence, and provide the correct German word ordering. (For details on the technical process, see Section 5.2.) Addition-

ally, the conversation-like writing process emphasizes audience-design concepts (Bell, 1984), i.e., it reminds the user to add information that's important for text understandability by its target (i.e., the reader) who cannot seek clarification – unlike the listener in face-to-face communication.

To create coherence over the entire text, the user is invited to clarify the discourse structure by adding connectors (inspired by *Rhetorical-Structure Theory* (RST); see Hovy (1988) and Mann and Thompson (1988)), thus explicitly marking the relationship between the simple clauses. (SVO order is mandatory in declarative main clauses of ELS). For example, the sentences in (3-a) express the train of thought much better than the staccato phrases in (3-b), thanks to the use of coherence cues (therefore/so, tomorrow, and the colon) and modifiers offering additional context (for you):

- (3) a. Du hast Corona. Darum: Ich will Dir helfen: Ich gehe
 You have COVID. Therefore: I want you to_help: I go
 morgen für dich Lebensmittel einkaufen.
 tomorrow for you grocery shopping.
 'You have COVID. Therefore: I want to help you: I will go grocery shopping for you tomorrow.'
- b. Du hast Corona. Ich helfe Dir. Ich kaufe Lebensmittel ein.
 You have COVID. I help you. I buy groceries.
 'You have COVID. I will help you. I will buy groceries.'

This resembles elements of the widely applied method of the *Schreibwerkstatt / Schreibkonferenz* / 'writing workshop'² used to teach German elementary school children the concepts of written text production (see, e.g., Reichardt et al. (2014) for a broad survey).

According to the iterative nature of the HCD process depicted in Figure 4.1, the design of *EasyTalk* was matured in multiple design and evaluation cycles. Details on the changes applied to the individual versions of the *EasyTalk* prototype can be found in the respective evaluations sections in Chapter 6; Section 6.5 provides a summary. The concrete implementation of the described design ideas of *EasyTalk* is presented with the system in chapter 5. For the sake of clarity, we present only the latest version of *EasyTalk* in this thesis.

²This technique is comparable to sentence-combining exercises in the Anglo-Saxon language area that teach students to integrate sets of short, disconnected sentences to form longer, more effective ones (see Nordquist (2018) for an online introduction, Ney (1980) for the history, and Saddler and Preschern (2007) for the school context).

4.5 Evaluation of the design

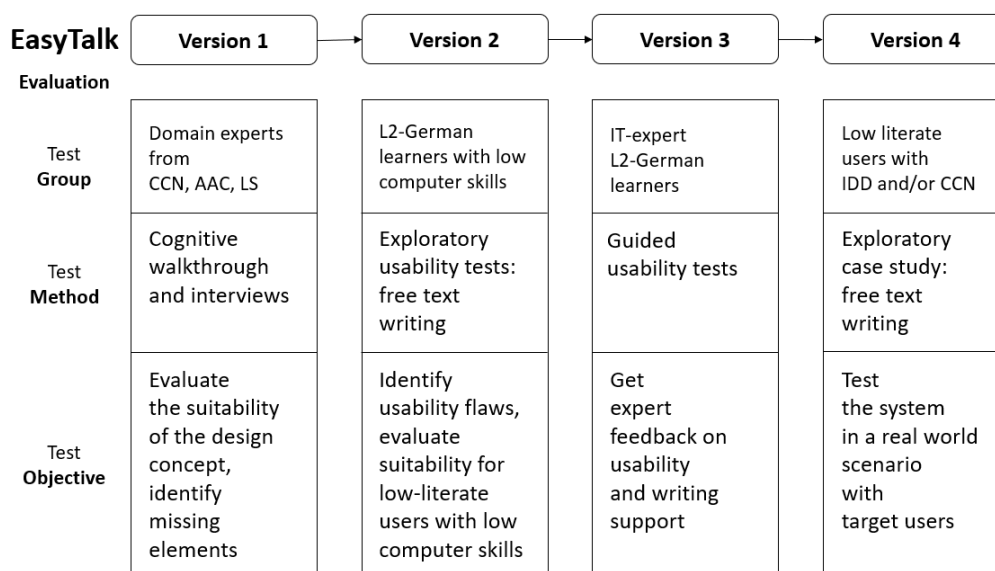


Figure 4.2: Overview of the participants, methods and objectives of the four evaluation cycles conducted during the development of the *EasyTalk* prototype.

In human-centered design, it is best practice to evaluate every design solution with real users. This is particularly important for user-sensitive, inclusive design of accessible, disability-aware software (cf. Henry (2007); Newell et al. (2011); Nganji and Nggada (2011)). Testing software with people with disability presents special challenges and organizational overhead (see, e.g., Henry (2007) or Lazar et al. (2017)). A particular challenge in testing with people with IDD and/or CCN is to obtain good feedback on complex, possibly abstract, questions. In general, it is best practice to identify and correct usability flaws in software before it is made available to the user (see, e.g., Holzinger (2005)). For the target group, however, the initial impression of a software system is particularly important – many AAC solutions are abandoned due to avoidable interface flaws (see, e.g., Fager et al. (2006); Schmidt (2006); Waller (2019)). For this reason, we made targeted use of substitute user groups during the evaluation of *EasyTalk* to mature the system before testing it with the target users in real writing sessions. Additionally, consulting the substitute user groups enabled us to include the perspectives of all stakeholders (i.e., users, their helpers, and potential readers of the texts) in the development process.

Figure 4.2 presents an overview of the four evaluation cycles we conducted. Pursuing to the HCD process sketched in Figure 4.1, we refined the user require-

ments (cf. Section 4.3), evolved the design (cf. Section 4.4) and matured the system (cf. Chapter 5) *EasyTalk* based on the results of each evaluation iteration.

After the first design iteration, we sought feedback from various perspectives as to whether the concept of our system is suitable as a writing aid for the target group. Therefore we evaluated an early-version prototype of *EasyTalk* by conducting cognitive walkthroughs combined with interviews with domain experts in CCN, AAC, LS, and the concerns of people with disabilities (cf. Section 6.1).

As the results showed that our prototype was heading in the right direction, we exploratorily tested free text writing with L2 learners with low German skills (also a target group of LS – see Section 2.2) with low computer skills (cf. Section 6.2). Here, we aimed to identify potential usability flaws in the overall system and tested the suitability of the interface and interaction patterns for users with low computer skills. Based on the results, we improved the design of our system.

Originally, we planned to conduct the third iteration with users of our target group. However, tests with the target group had to be stalled due to the health precautions and local safety restrictions due to COVID-19, which particularly affected our target audience (cf. for example, Rödler (2020) and Portal et al. (2021)). Instead, we conducted guided usability tests with a group of IT-expert L2 learners of German (cf. Section 6.3). We aimed to further improve the usability of our system by seeking expert-level feedback. Additionally, aimed to elicit clues where to extend the linguistic support our target group might benefit from.

Finally, we tested the matured prototype of *EasyTalk* with in an exploratory case study with users of our target group (Section 6.4). Aiming for a real world application scenario, we visited a writing workshop for adult users with severe or multiple disabilities. In this setting, target users performed free text writing sessions using *EasyTalk*. As described by Nganji and Nggada (2011) and Henry (2007), we encountered aspects that require special consideration in testing with the target group, such as very heterogeneous user characteristics (e.g., CCN in the form of being non verbal or problems with spontaneous decision making) and the need for personalized interfaces (e.g., due to visual impairments) or assistive components in form of specialized keyboards or mice (e.g., due to motor impairments). Here, we focused on testing the core functionality against the basic system requirements (see 4.3) of the high fidelity prototype of *EasyTalk*. To perform the tests as distraction-free as possible, we assisted the participants as needed in the test sessions – for example, with spelling or controlling the system.

The results of final test iteration reflect the current state of the prototype as presented in Chapter 5.

Chapter 5

EasyTalk

Our goal is to help the LS target audience become LS writers by providing them with an assistive writing system. In this chapter, we describe *EasyTalk* – our scaffolded-writing system for LS, designed to enable our target users from the spectrum of low literate people with IDD and/or CCN to formulate correct and coherent ELS texts through linguistic processing by computer, interactive grammar guidance and elements of AAC. *EasyTalk* helps users to practice text production at the personal skill level by writing sentences of varying complexity and prompting the user to add important information for the reader and create text coherence.

The chapter commences by introducing the natural language generation concepts employed in *EasyTalk*. Next, we describe the theoretical background of the system’s most important functionality focusing on the sentence-writing process (Section 5.2). In turn, an introduction to the system follows (Section 5.3), providing an overview of its workflow and features. Then, Section 5.4 details on the text-writing support of *EasyTalk* on different levels of the text-production process. Next, Section 5.5 presents a concept for teaching text production in form of a teacher mode within *EasyTalk*. Finally, Section 5.6 describes additional *EasyTalk* features, like options for system customization and text functions.

5.1 Concepts of text-production based on Natural Language Generation employed in EasyTalk

In this section, we introduce the natural language generation concepts we employ in *EasyTalk*. (This passage has previously been published in Harbusch and Steinmetz (2022)). Therefore, we outline the steps from a speaker’s intention to a context-sensitive utterance to allow us to highlight the NLG concepts used in *EasyTalk*.

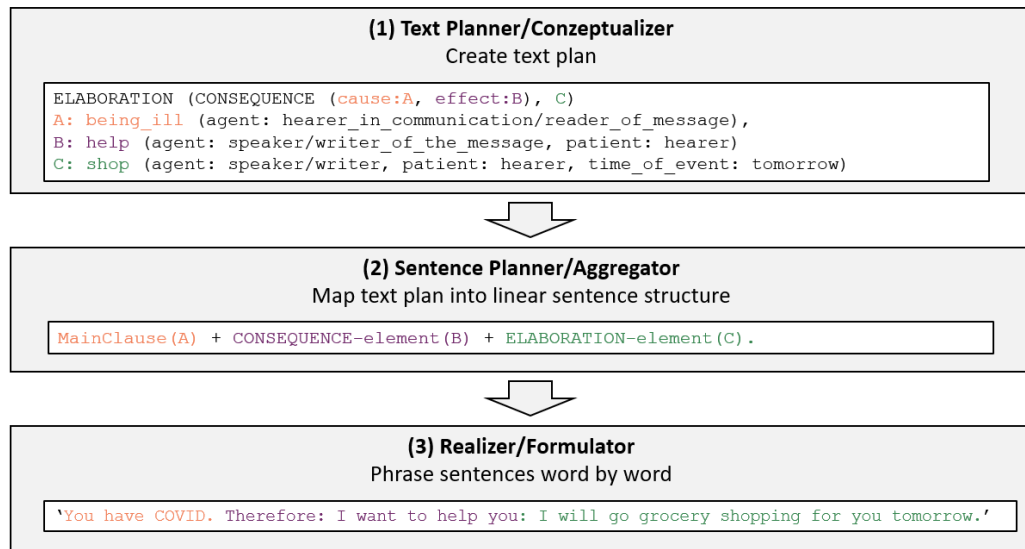


Figure 5.1: Basic elements of a natural language generator (bold), illustrated with our example (in text boxes).

Figure 5.1 shows the typical three-stage pipeline architecture of a declarative text generation system (see, e.g., Reiter and Dale (2000)), illustrated in terms of our example (3-a) from the introduction.

The overall input to the Text Planner/Conceptualizer (Figure 5.1 (1)) is encoded as the speaker’s *goal*:

```

INFORM(S, H, KNOW(H, shopping(agent_of_action_shopping: S, object:
groceries, time: tomorrow))).
  
```

S refers to the speaker, and *H* to the hearer. The goal is an INFORM-speech act, i.e., *S* wants *H* to know something that *S* assumes *H* does not yet know. (In a REQUEST-speech act, *S* wants to obtain new information that *S* assumes *H* knows, often resulting in a question like “*Can you please tell me how I can help you tomorrow?*”.) Roughly speaking, the goal highlights the discrepancy between the knowledge bases of *S* and *H*, which should be removed through a series of speech acts, i.e., the not-yet-verbalized conceptual messages (propositions dealt with in the Conceptualizer). Ideally, after delivery of the message by *S*, *H* knows all communicated facts (and facts that can be inferred by *H*). This task requires separate representations of the speaker’s and the hearer’s knowledge about the current situation and about their presupposed *world knowledge*, respectively (including the implications of/inferences from all the facts).

In the example, we assume that *S* notices that *H* is ill. *S* wants to indicate to *H* that *S* is aware of this fact. By explicitly informing *H* that *S* knows that *H* is ill,

the speaker enriches the utterance with a known fact to make the context/intention of the utterance clear, thereby creating an overall *discourse structure*. Because of the close personal relationship between *S* and *H*, *S* decides to help by buying groceries for *H* the next day. This plan results from the world-knowledge fact that relieving an ill person of a task helps that person to rest and recover.

The Conceptualizer decides which information should be communicated. In our example, the facts *A*, *B*, and *C* are selected (the propositions are rendered here in the form of sentences, abstracting away from the logic representation and detailed argumentation; instead, each proposition is supplemented by the intended interpretation of the hearer):

A: *Du hast Corona* ‘You have COVID’: the context of the utterance
B: *Ich will dir helfen* ‘I want to help you’: reason for a proposed action
C: *Ich kaufe morgen Lebensmittel für dich ein* ‘I will go grocery shopping for you tomorrow’: communication of the planned action

Importantly, propositions do not stand in isolation, but in relationship, in order to express the discourse structure/the speaker’s intention. A widely used technique for this purpose is that of Rhetorical Structure Theory (RST; see Hovy (1988) and Mann and Thompson (1988)). Two important examples of relations between propositions are ELABORATION and CONSEQUENCE. The resulting hierarchical structure of interrelated propositions is called the *text plan*. Text plans are handed over to the Sentence Planner/Aggregator module (Figure 5.1 (2)), which has the task of linearizing the hierarchical structure. The linearization process involves, among other things, the insertion of coordinating and subordinating conjunctions and other lexical items that instantiate RST relations (although not all RST relations need to surface explicitly in the final text). In terms of our example:

- *A* is realized as the main clause;
- CONSEQUENCE(*B*) is realized by the causative adverb *therefore* preceding the main-clause realization of *B*, and
- ELABORATION(*C*) is realized by a colon preceding the main-clause realization of *C*.

The Realizer/Formulator (Figure 5.1 (3)), provides the subsequently generated text – one of the many realization options of the overall generation system.

In the NLG nomenclature, all writing support systems mentioned in Chapter 3 are restricted to formulator problems.

In the following Chapter, we illustrate the potential of using concepts from all three generation steps in *EasyTalk*.

5.2 Theoretical system description

In the previous section, we introduced the NLG concepts employed in *EasyTalk* in the form of a three-step process using a Text Planner, a Sentence Planner and a Formulator (cf. Figure 5.1). Here, we detail on the theoretical background of our system’s NLG component used for sentence planning and sentence formulation in the sentence-writing process.

First, in Section 5.2.1, we detail on the base concepts deployed in *EasyTalk*. Then, we describe this aspect of our system roughly based on the *4+1 View Model of architecture* introduced by Kruchten (1995) in a top down manner. In Section 5.2.2, we depict the main *scenarios* in the form of use cases. We focus on the use cases of sentence writing. We distinguish which ones are handled by the user and which are handled by the system. Next, in Section 5.2.3, we consider the actions of sentence writing from the *process view* by describing the activities in their chronological order *EasyTalk*. Then, in Section 5.2.4, we describe the information exchange between the system and the user in the course of these activities from the *development view*. In Section 5.2.5, we highlight the NLG core component from the *logical view* and take a detailed look at its structure. Finally, in Section 5.2.6, we explain the processes of its core functions in detail.

5.2.1 Base concepts

Rich inflection and *free word order* are two characteristics of the German language that complicate the generation of correct suggestions in the context of linguistic writing support by computer (cf. Figure 5.2). Since these two phenomena are much less pronounced in other languages, such as English, we now briefly introduce them to illustrate resulting challenges for our NLG component.

Due to *rich inflection*, word endings in German vary depending on the grammatical context. Figure 5.2a illustrates this with six examples of symbol-word combinations common in AAC. In each example, the user has written two words. The first word, the subject, varies in each example (cf. *ich* ‘I’, *du* ‘you’, *wir* ‘we’, ...) and is displayed with different symbols accordingly. While the lemma of the second word, the verb *geben* ‘to give’, is consistent and thus always shows the same symbol, its form is inflected according to the number of the subject to satisfy the subject-verb agreement (cf. the 6 underlined word forms in the dashed boxes).

With increasing sentence length, the challenge of always providing the user with the correct word form grows – especially since word order is free in German. This means that the sentence components can take different positions in the sentence, and writing systems cannot rely on the user to always enter the sentence components in the same order.

Typed symbols:		
Computer-produced sentence:	Ich <u>gebe</u>	Wir <u>geben</u>
English translation:	'I give'	'We give'
Typed symbols:		
Computer-produced sentence:	Du <u>gibst</u>	Ihr <u>gebt</u>
English translation:	'You give'	'You give'
Typed symbols:		
Computer-produced sentence:	Sie <u>gibt</u>	Sie <u>geben</u>
English translation:	'She gives'	'They give'

(a) An example illustrating *rich inflection* in the German language in combination with AAC symbols: The same symbol can produce different word forms in a sentence. The form of *geben* 'to give', dashed frame, changes with the subject associated with the first symbol.

1	Typed symbols:	
	German sentence:	Die Frau gibt dem Mann das Buch.
	English Translation:	'The woman gives the man the book.'
2	Typed symbols:	
	German sentence:	Die Frau gibt das Buch dem Mann.
	English Translation:	'The woman gives the man the book.'
3	Typed symbols:	
	German sentence:	Das Buch gibt die Frau dem Mann.
	English Translation:	'The woman gives the man the book.'
4	Typed symbols:	
	German sentence:	Der Frau gibt der Mann das Buch.
	English Translation:	'To the woman, the man gives the book.'
5	Typed symbols:	
	German sentence:	Der Mann gibt der Frau das Buch.
	English Translation:	'The man gives the woman the book.'

(b) Five example sentences illustrating free word order in German. The *subject* of each sentence is marked green, the *direct object* (Whom?) blue and the *indirect object* (To whom?) orange. In Sentences 1-3 *die Frau* 'the woman' is the subject. In Sentences 4 and 5, *der Mann* 'the man' is the subject. This is recognizable by the articles associated with the nouns, which vary with the grammatical function.

Figure 5.2: Two figures illustrating the phenomenons *rich inflection* (a) and *free word order* (b) in the German language.

Figure 5.2b demonstrates this with five examples and shows, that the order of the symbols in a sentence cannot be used to infer its meaning: In Sentence 1-3, the subject (*the woman* ‘the woman’), the direct object (*the book* ‘the book’), and the indirect object (*the man* ‘the man’) are the same, even though they are located at different positions in the sentence. In Examples 4 and 5, *the man* (now subject) and *the woman* (now indirect object) switch grammatical functions. This is indicated by the changed articles before the nouns, which vary according to case, grammatical gender, and number of a noun in German. The comparison of Sentences 1-3 and Sentences 4 and 5 shows the impact of articles in German for the meaning of a sentence. Different articles and word endings can easily be overlooked by low-literate users and lead to unintended meanings.

Generative methods, such as the machine-learning based approaches we presented in Section 3.3, create complete sentences and texts. The user must read, understand, and edit these to detect and correct deviations from the intended meaning. This task is beyond the capability of our target group of low-literate users with IDD and/or CCN. The Augmentative and Alternative Communication systems and assistive text editors presented in Section 3.1 and 3.2 rely on users either choosing word endings themselves or correcting false suggestions as needed. Accordingly, the systems require a certain level of grammatical knowledge and attention on the part of the user.

With respect to the goal of supporting users with very low writing skills while taking into account the challenges of cognitive flexibility and memory limitations, we aim for a guided step-by-step process that shifts responsibility for the grammatical correctness and completeness of the text to the system. At the same time, the choice of words remains in the hand of the user. Therefore, we employ a variant of an NLG formulator (cf. Section 5.1). Its goal is to build up a derivation tree based on the rules of a syntactic grammar (here, ELS constructions; if desired, the declarative grammar can easily be restricted to pure LS rules – cf. Section 5.6.2 for system customizations) so that syntactic correctness is automatically maintained. Based on this representation, *EasyTalk* produces correctly inflected word forms. In NLG, the formulator usually administers only the best sentence representation; there is no user interface (UI) enabling the selection of another option (paraphrase). We adopt a slightly more flexible formulator approach, developed for L2 learners of German. *COMPASS* (Harbusch et al., 2007, 2014) is a natural-language paraphrase generator that constructs the sentence the user has in mind in a step-by-step dialogue. The process is called *scaffolded writing*, in reference to the fact that the system is able to maintain the syntactic correctness of the construction the user is typing after the user has specified its grammatical function (Harbusch and Kempen, 2011). *COMPASS* is based on the grammar rules in *Performance Grammar*, a psycholinguistically motivated grammar formalism (Harbusch and Kempen, 2002; Kempen and Harbusch, 2002).

The separation into distinct dominance and word order rules in Performance Grammar enables a flexible sentence-production process in COMPASS to suit the user's preferences. For example, the user can enter all arguments first to empty the short-term memory and then fully concentrate on arranging the constituents according to the intended discourse structure. Revisions made throughout the sentence at any point in time are retained.

EasyTalk's NLG component reuses parts of COMPASS. In the following, we describe which concepts and data we have adopted in *EasyTalk*. In contrast to COMPASS, *EasyTalk* implements a linear step-by-step writing process. Hence, *EasyTalk* requires less flexibility in the sentence-production process than COMPASS. Once the user has entered an argument, it remains at its position in the sentence. Therefore, we do not need to adopt the concept of late linearization and the required implementation of a complex derivation tree. It is sufficient for us to extract the rules for the word order and structure of the individual sentence components from the grammar of COMPASS. The resulting set of declarative rules is restricted to ELS constructions (cf. Table 2.3). For example, the range of verb forms is restricted to the active voice, indicative mood, and present and present perfect tenses. For auxiliaries and modals only, the preterite and the subjunctive mood are also offered (e.g., *Wir wollten uns heute in der Stadt treffen*. 'We wanted to meet in the city today.'). The system favors non-inversion word order. See Table 5.1 for the order in which the constituents are presented in main declarative sentences. To provide an intuitive UI, it is crucial to avoid linguistic terms; therefore, we use cues in the forms of interrogative pronouns¹, as outlined in Column 1, to communicate with the user about grammatical functions and maintain scaffolded writing. (As described in the design stage in Section 4.4, we decided on this technique as it is reminiscent of gathering information in a natural dialog and resembles widely applied school exercises for identifying grammatical function fillers in a sentence.) In return, the system is enabled to propose correctly inflected forms.

In addition to the rule set, *EasyTalk* and COMPASS share the same lexicon. In essence, the overall lexicon covers the German *CELEX* (Gulikers et al., 1995). To obtain a reasonable suggestion list, this lexicon is restricted to L2-learner level A1/A2 in *EasyTalk*. The lexicon can also be adapted to the user's personal vocabulary (e.g., to include proper names of protagonists or places; for details on the process, see Section 5.6.2).

In the following, we detail on the sentence-writing process in *EasyTalk* to describe how the system employs the base concepts described above.

¹For the direct object and the indirect object, we intentionally use the cues *Wen* and *Wem*, which are reserved for animate fillers, to make case assignments easier.

Table 5.1: List of constituents in a main declarative sentence (in the top panel, subject and finite verb forms are obligatory; the second panel enumerates further arguments/valency-frame fillers of the finite verb; and in the lower panel, adjuncts/modifiers are enumerated). Column 1 provides the cue words to communicate the grammatical function of the component to the *EasyTalk* users.

Cue	Automatically inflected filler
Wer ‘who _{nom} ’	Elements of the SUBJ ect in nominative case
Tut ‘does’	FIN ite verb form in active voice, present tense, coinciding in person and number with the subject
Wem ‘whom _{dat} ’	Elements of the Indirect OBJ ect in dative case
Wen ‘whom _{acc} ’	Elements of the Direct OBJ ect in accusative case
P_f was ‘P _f what’	Elements of the Prepositional OBJ ect in the case P_f , the instantiated preposition requires
Was tun ‘what to do’	Past Participle in case the finite verb form is an auxiliary or INF initive in case the finite form is a modal or Infinitive_with_ZU in case the finite form is a complement-taking verb
Wann ‘when’	Elements of MOD ifier_time
Wo ‘where _{loc} ’	Elements of MOD ifier_location
Woher/-hin ‘where _{dir} ’	Elements of MOD ifier_direction from/to
Wie ‘with what’	Elements of MOD ifier_instrument

5.2.2 Use cases involved in sentence writing

This section describes the main use cases of sentence writing. We distinguish which ones are handled by the user and which use cases are handled by the system.

Figure 5.3 depicts the main use cases of *EasyTalk* in the form of an *UML use case diagram*. The diagram focuses on the process of writing a sentence in *EasyTalk* by splitting it into further use cases. It associates the use cases with their actors and illustrates the distribution between user interaction tasks and autonomous system tasks – here restricted to the NLG component. The use cases of the two stakeholders directly interacting with the system (i.e., *user* and *helper* – cf. Section 4.3) are colored white. The use cases handled by the system are colored gray.

To reduce the need for writing skills, the system is responsible for providing the user with all the necessary information for writing a correct, complete sentence. The system makes word order suggestions according to the sentence type selected by the user (e.g., declaration or question). It provides correctly inflected word forms for the current grammatical context. The user selects word forms from these suggestions to add to the sentence.

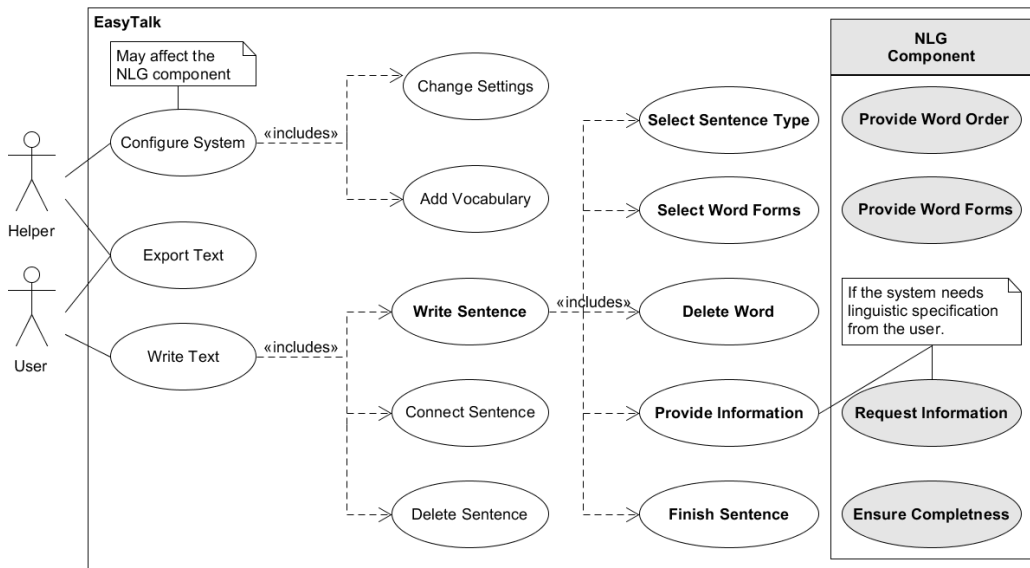


Figure 5.3: An UML Use Case Diagram showing the main use cases of *EasyTalk* in association with the actors. The use cases of the stakeholders *user* and *helper* are colored white, the ones of the *system* gray. The diagram focuses on process of writing a sentence (bold), depicting the main actions involved.

To make corrections, the user can delete word forms. If the user wants to finish the sentence, the system checks whether all mandatory components (e.g., subject and verb) are filled and thus ensures the completeness of the sentence. If linguistic ambiguities arise during writing, the system requests the necessary information from the user in a simple manner. In summary, *EasyTalk*'s NLG component aims at fast and correct writing by supporting the user by providing the correctly inflected word forms in any sentential context, mentioning all obligatory sentence components, and maintaining the correct word ordering according to the sentence type.

5.2.3 Sentence writing activities

The sentence-writing process of COMPASS and *EasyTalk* differ significantly. While COMPASS allows the user to rearrange sentence components on the fly, *EasyTalk* guides the user through the sentence in a linear manner.

In the following, we illustrate linear scaffolded writing with *EasyTalk* using the *UML activity diagram* in Figure 5.4. Like Figure 5.3, it distinguishes between user activities (on the left, white) and activities of the system's NLG component (on the right, gray). For simplicity, we omit the use case of deleting a word form from the sentence, considering it the *undo action* of adding a word form.

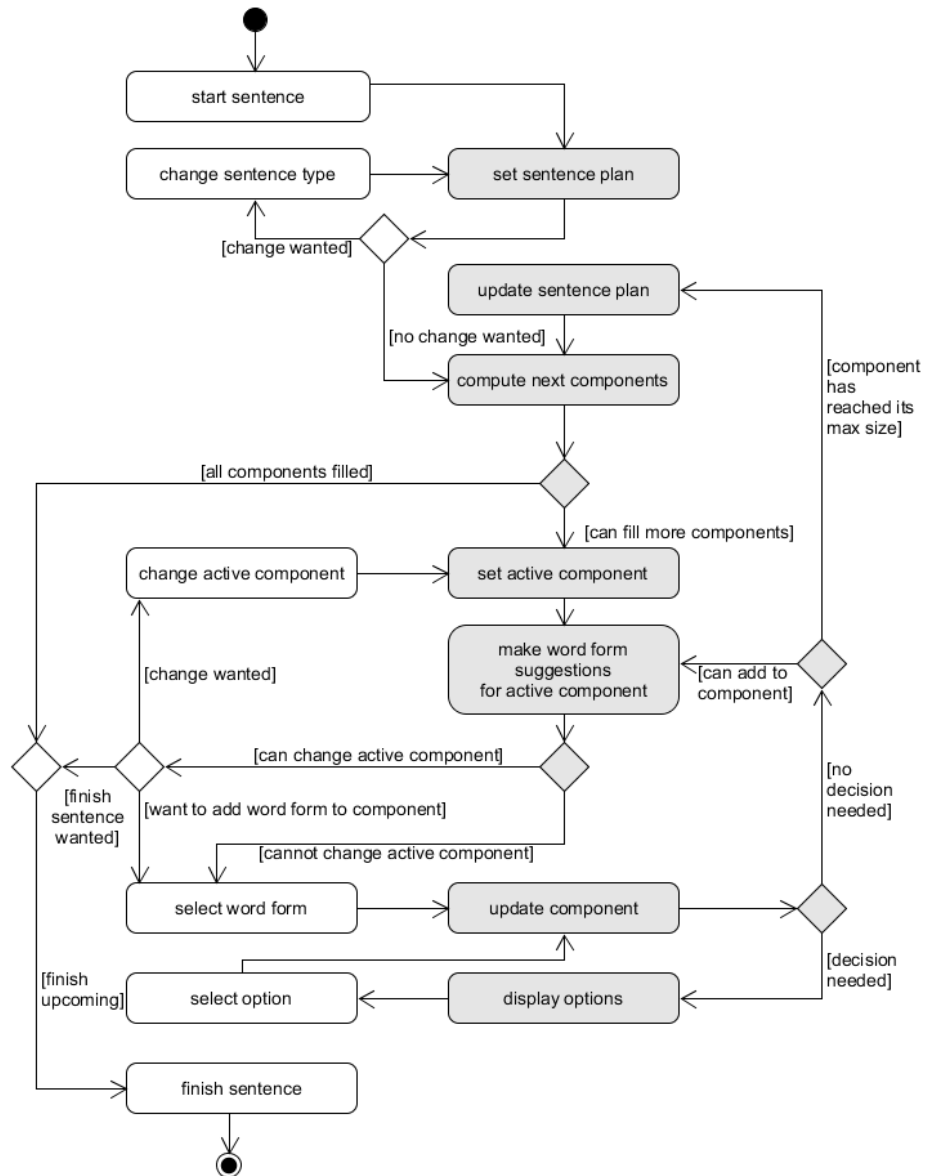


Figure 5.4: An UML activity diagram illustrating the sentence-writing process in *EasyTalk*. The user actions are placed on the left side and colored white. The system actions are placed on the right side and colored gray.

The first user action is starting the sentence. By default, *EasyTalk* chooses the *sentence plan* describing the canonical word order of a main declarative sentence (cf. Table 5.1). As long as the sentence is empty, the user can change the type of the sentence – for example to write a question. If the user changes the sentence type, *EasyTalk* changes to the corresponding sentence plan. The system computes the next possible components according to the active sentence plan. In the main declarative clause presupposing SVO word order, for example, the user must fill the subject first. Once the verb has been entered, the arguments according to its valency frame and modifiers like time and place can be filled. According to the usual German word order, *EasyTalk* suggests the user which component to fill next – setting it the active component. For this component, it offers inflected word form suggestions. If the user wants to deviate from this order (e.g., to emphasize a component), they can activate another component and *EasyTalk* adapts the word form suggestions accordingly. In two cases, the user cannot change the active component: In the first case, a component must be completed first in order to continue the sentence (e.g., the verb must be entered before its arguments). In the second case, a component that has been started is incomplete (e.g., “the” would be an incomplete argument and “the woman” would be a complete one). Agreeing with the active component, the user adds word forms to it by selecting them from the suggestions made by *EasyTalk*. After each word form, the system updates the component. Here, it may need more linguistic information. If the user, for example, has entered an auxiliary verb, *EasyTalk* needs to know if the user wants to continue the sentence in present tense or present perfect tense. The system presents the choice options to the user, who makes a selection. If the system has all information, it checks, whether the user can still add to the component (e.g., *dir und* ‘you and’) or the component has reached its maximal size (e.g., *dir und mir* ‘me and you’). In case the user can still add to the component, the word adding process is repeated. A component can be finished in two ways: In the first case, the user switches to another component. This is only possible, if the current component is complete. In the second case, the component cannot be further expanded. Once a component is finished either way, the system updates the sentence plan. It removes the finished component from the list of components to be filled, updates the list of next components and proposes the user with the next component to fill. At the latest, when all components are filled, the sentence is finished. Of course, the user can finish the sentence beforehand. This is possible, as soon as all obligatory elements have been added to the sentence and the current component is in a complete state.

5.2.4 System components for sentence writing

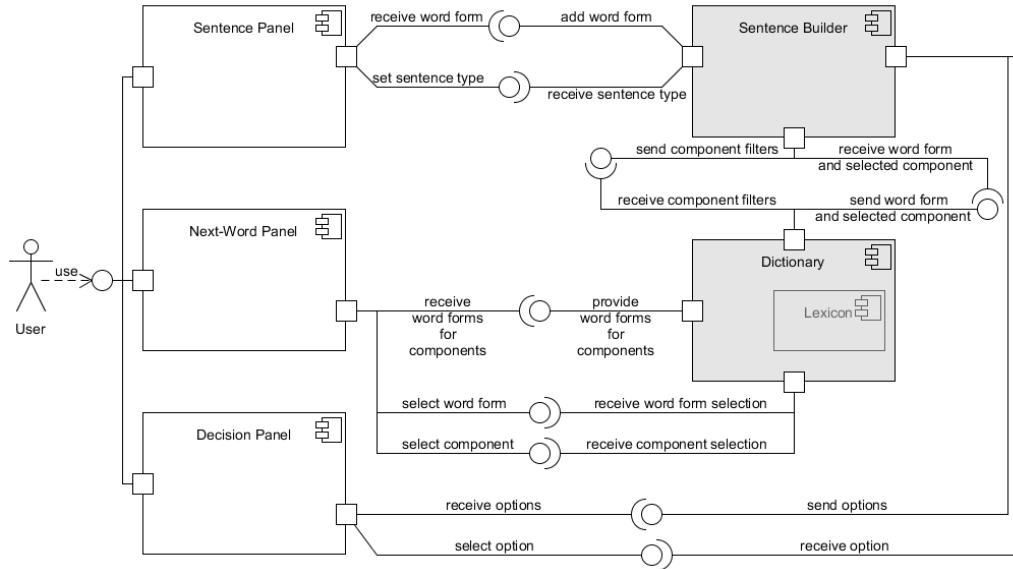


Figure 5.5: An UML component diagram showing the system components involved in the sentence writing process. The NLG components of *EasyTalk* are rendered gray, the user interface components are filled white.

Now that we considered the user and system activities of the writing process in detail, we take a look at the corresponding system components.

Figure 5.5 shows the system's structure by using an *UML component diagram*. It visualizes the interactions between the elements of the user interface (on the left, white) and the NLG components (on the right, gray). During the sentence-writing process, the user interacts with the system using three panels: the *Sentence Panel*, the *Next-Word Panel*, and the *Decision Panel*. We detail on these panels in the system introduction in Section 5.3 and its following sections. Here, we focus on the data flow. As Figure 5.5 illustrates, the NLG component provides the interface with all choice options. It passes the user's choices back to the NLG component where they are processed. We divide the NLG component into two main sub-components: the *Sentence Builder* and the *Dictionary*. The Sentence Builder is responsible for the correctness and completeness of the sentence. The Dictionary holds the lexicon (in the form of an SQL database containing the COMPASS version of CELEX). Accordingly, the Sentence Builder tells the Dictionary which word forms to send to the interface in the current sentential context. This is communicated in the form of *Component Filters*. A Component Filter consists of the inflections codes for the current Component Type (e.g., subject, verb, or direct object) needed to extract the corresponding word forms from the lexicon.

5.2.5 Natural language generation core components

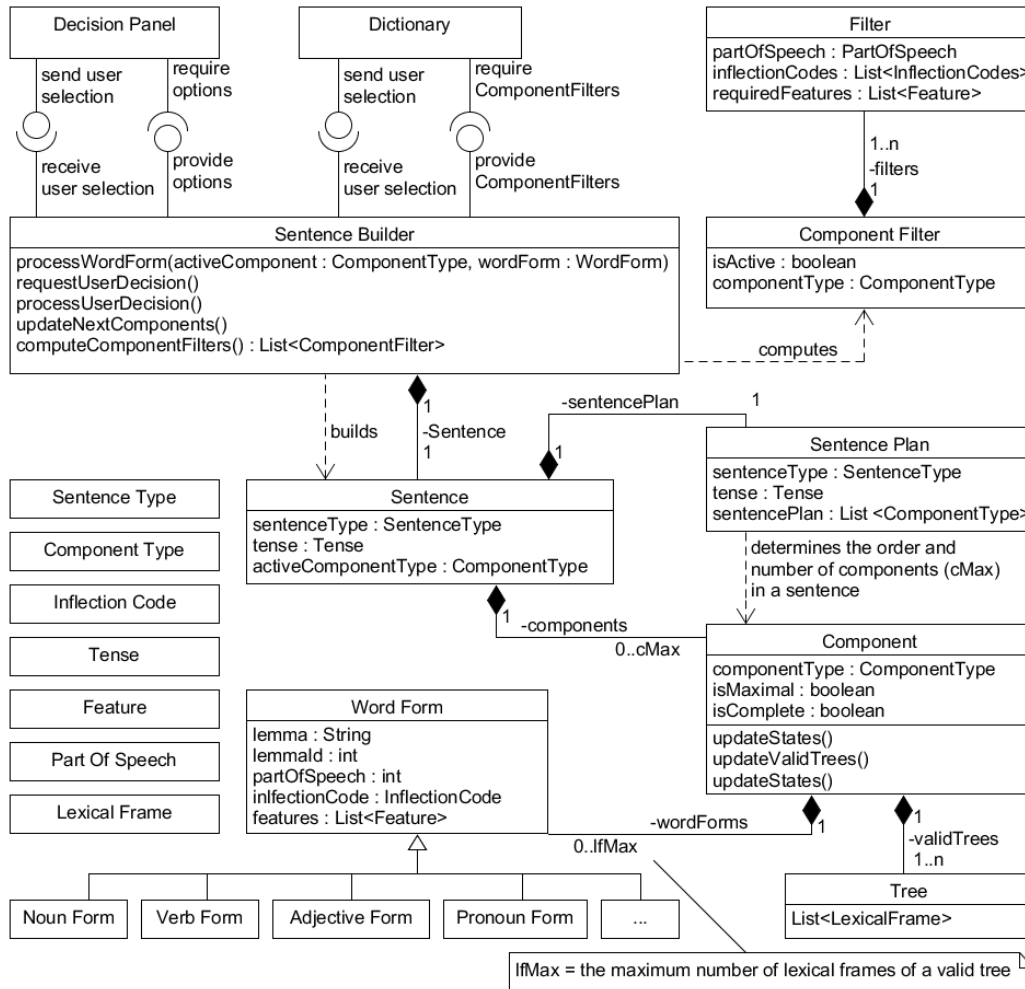


Figure 5.6: An UML class diagram of *EasyTalk*'s NLG core component. A detailed description is provided in the text.

The calculation of the Component Filters is the core task of our NLG component and takes place in the Sentence Builder, which we present in detail below.

Figure 5.6 illustrates the structure of *EasyTalk*'s NLG core component in the form of an UML class diagram. For the sake of clarity, it contains only the most important classes, attributes and functions. In the upper left corner we see the Decision Panel and the Dictionary, the two system components that communicate with the Sentence Builder. Through these components, the Sentence Builder provides the choice options to the user. In turn, they communicate the user's selection to the Sentence Builder. We focus on the most complex task, the computation of

the Component Filters, which are used to provide the user with correctly inflected word forms for the current sentence context, taking into account the word order of the current *Sentence Type* (e.g., declaration or question). The Sentence Builder holds a *Sentence*. The Sentence uses a specific *Sentence Plan* according its Sentence Type, which contains a set of slots to be filled with *Components* of a certain *Component Type* (e.g., *Subject*, *Verb*, *Direct Object*, ...). The Components of the Sentence Plan are ordered according to the word order of the Sentence Type and additional factors, such as the *Tense* of the Sentence. Each Component can be filled according to a set of *Trees* corresponding to its Component Type. To obtain the Trees, we used the COMPASS linear order rule set reduced to the scope of ELS². Example Trees are:

- {*pers.pro*}
a personal pronoun
e.g., *ich* ‘I’,
- {*poss.pro, adj, n*}
a possessive pronoun followed by an adjective and a noun
e.g., *mein kleiner Hund* ‘my little dog’,
- {*art, n*}
an article followed by a noun
e.g., *der Mann* ‘the man’,
- {*adj, n*}
an adjective followed by a noun
e.g., *kleine Hunde* ‘small dogs’,
- {*art, adj, n*}
an article followed by an adjective and a noun
e.g., *der kleine Hund* ‘the small dog’,
- {*art, crd, adj, n, prep, dem.pro, adj, n*}
an article followed by a cardinal number, an adjective, a noun, a proposition, a demonstrative pronoun, an adjective and a noun
e.g., *der eine leckere Kuchen von meiner lieben Oma* ‘the one tasty cake of my beloved grandma’

²An exact definition of the component size is not given in LS. Using the LS rules ‘only one thought per sentence’ and ‘keep sentences short’ as well as insights of the LS corpus study by Harbusch and Steinmetz (2022), we define that a component Tree may consist of a maximum of two subtrees linked by or/and, each of which may have a prepositional attribute associated with it – as commonly used to express relationships in LS, such as *die Katze von der Frau* ‘the cat of the woman’.

As the examples show, each Tree provides a linear sequence of *Lexical Frames* within a component in a grammatically correct order. Each Lexical Frame can be filled with an inflected *Word Form* of a *lemma* (i.e., the *base word*) matching the *Word Type* of the Lexical Frame. To be grammatically correct, this Word Form must meet specific conditions within the Component and the context of the Sentence. Within the Component, attributes such as grammatical gender, case, and number of the *preceding* Word Forms determine which Word Forms can follow. For example, a Component starting with *kleine* ‘small’ requires the following Word Forms to be either *nominative, plural* (e.g., the subject *kleine Hunde* ‘small dogs’) or *accusative, plural* (e.g., the direct object *kleine Männer* ‘small men’). Depending on the Word Type, or *Part of Speech*, different attributes come into play. For adjectives, for example, the declension type has to be considered. In *EasyTalk* we call these attributes *Features*.

5.2.6 Computing word form suggestions

The concepts described before are used to compute the word form suggestions for the current sentential context.

The rich morphology and word order rules of the German language make providing correct Word Forms a complex task. Two more examples for filling in the last and most complex example Tree in the list above highlight the rich morphology:

- (1) *die drei kleinen Hunde von meinem schönen Nachbar*
the three little dogs of my beautiful neighbor
- (2) *der eine kleine Hund von meinen schönen Nachbarn*
the one little dog of my beautiful neighbors

With the exception of the cardinal number, both examples use the same lemmas. The different choice of Word Forms determines two varying meanings: In Example (1), three dogs of unknown gender belong to one male neighbor. In Example (2), one male dog belongs to an unknown number of multiple neighbors of unknown gender.

In addition to the meaning, the Word Forms may influence subsequent Components in the sentence context by invoking additional rules, such as word order rules or the subject-verb agreement. If the user chooses example (1) to fill the subject, the NLG component must propose *Verb Forms* in *third person plural* (e.g., ‘play’) to satisfy the subject-verb agreement. In the case of example (2), the Verb Form must be *third person singular* (e.g., ‘plays’), accordingly.

In the following, we describe the most important algorithms that determine the creation of the Component Filters using (high level) pseudo code.

Algorithm 1: processWordForm

```

Data: wordForm, componentType
1 if needAdditionalInformation then
2   |   userDecision = requestUserDecision();
3   |   processUserDecision(userDecision);
4 end
5 activeComponent = Sentence.getActiveComponent();
6 if activeComponent.getComponentType() != componentType then
7   |   activeComponent = Sentence.addNewComponent(componentType);
8 end
9 activeComponent.addWordForm(wordForm);
10 activeComponent.updateValidTrees();
11 activeComponent.updateStates();
12 sentence.updateNextComponents();
13 computeComponentFilters();

```

Every time the user chooses a Word Form in the corresponding interface element, the Dictionary extracts all Word Form information from the lexicon and passes it along with the active Component Type to the Sentence Builder. (For performance reasons, the Dictionary keeps only the minimum necessary information in memory.) In turn, the Sentence Builder executes the *processWordForm*-routine described in Algorithm 1. First, the Sentence Builder evaluates if it needs additional contextual information to be able to correctly process the Word Form (Lines 1-4). If this is the case, the Sentence Builder provides the users with choice options and waits for the user's response before it continues. In the next step (Lines 5-8), the Sentence Builder checks, if the Component Type chosen by the user matches the Component Type of the currently active Sentence Component. If they differ, the Sentence Builder finishes the current Component and adds a new Component of the chosen Component Type to the Sentence. Then, the selected Word Form is added to the active Component (Line 9) and its valid Trees are updated (Line 10). A valid Tree matches the Lexical Frames of the Word Forms added to the Component so far (e.g., $\{art, n\}$ and $\{art, adj, n\}$ are both valid Trees for a Component starting with the article *der* 'the'). Next, the states of the active Component are updated (Line 11; for details, see Algorithm 2). Depending on these states, the next Components are computed (Line 12; for details, see Algorithm 3). Finally, the Sentence Build can compute the Component Filters (Line 13; for details, see Algorithm 4) and pass them to the Dictionary so that it can provide the user with the options for the next Word Form.

Algorithm 2: updateStates

```

1 componentSize = activeComponent.getNumberOfWordForms();
2 for all validTrees do
3   | treeSize = tree.getNumberOfLexicalFrames();
4   | if componentSize = treeSize then
5   |   | activeComponent.isComplete = true;
6   |   | break;
7   | end
8 end
9 for all validTrees do
10  | if tree.hasEmptyLexicalFrame() then
11  |   | activeComponent.isMaximal = false;
12  |   | break;
13  | end
14 end

```

Algorithm 2, *updateStates*, describes how the two states, being *complete* and being *maximal* of the active Component are determined.

To decide if a Component is complete, the number of Word Forms is matched against the sizes of all valid trees. If one Tree is an exact match, the component is complete. For example, the tree $\{pers.pron, conj., pers.pron\}$ is valid for the subject containing the Word Forms *ich und* ‘me and’. However, the Component is not yet complete. Adding a second personal pronoun would complete the Tree and the Component. After that, the user would not be able to add any more words to the component since no other tree is valid for this sequence of lemmas. Hence, the Component *ich und du* ‘me and you’ is complete and maximal. In other words, a Component is being maximal as soon as only one valid complete Tree is left.

Depending on the states of the active Component, the Sentence decides which Components can be filled next. This process is described in Algorithm 3, *updateNextComponents*. As long as the current Component is incomplete, the user must add to it. In this case, the Sentence Builder computes a Component Filter only for this Component (Lines 1-3) and passes it to the Dictionary. If the Component is complete, the user can opt to switch to another Component (Lines 4-9). The options for this are computed using the Sentence Plan (Line 4). Only if the active Component is not yet maximal, it is added to the set of next Components itself (Line 7).

Algorithm 3: updateNextComponents

```

1 if activeComponent.isComplete = false then
2   | nextComponents.add(activeComponent);
3 else
4   | components = SentencePlan.getNextComponents();
5   | nextComponents.add(components);
6   if activeComponent.isMaximal = false then
7     | nextComponents.add(activeComponent);
8   end
9 end

```

Algorithm 4: createComponentFilters

Data: nextComponents**Result:** componentFilters

```

1 componentFilters;
2 for all nextComponents do
3   | featureCollection;
4   | componentsOfInfluence = component.getComponentsOfInfluence();
5   for all relevantComponents do
6     | wordForms = componentsOfInfluence.getWordForms();
7     for all wordForms do
8       | features = wordForm.getFeatures();
9       | featureCollection.add(features);
10    end
11    | unifiedFeatures = unify(featureCollection);
12  end
13  for all valid Trees do
14    | nextLexicalFrame = getNextLexicalFrameInTree();
15    | wordType = nextLexicalFrame.getWordType();
16    | filters = mapToInflectionCodes(wordType,unifiedFeatures);
17    | componentFilter = createComponentFilter(componentType,
18      | filters);
19    | componentFilters.add(componentFilter);
20  end

```

Finally, the Sentence Builder computes the Component Filters for all next Component options according to Algorithm 4. For each next Component option, it considers the Components of influence (Line 4) according to the grammar rules. Of the influencing Components, it collects the relevant Features of all Word Forms (Lines 5-10). In turn, it unifies the Features and creates distinct Feature Sets (Line 11) valid for the current sentence context. The list below shows three examples:

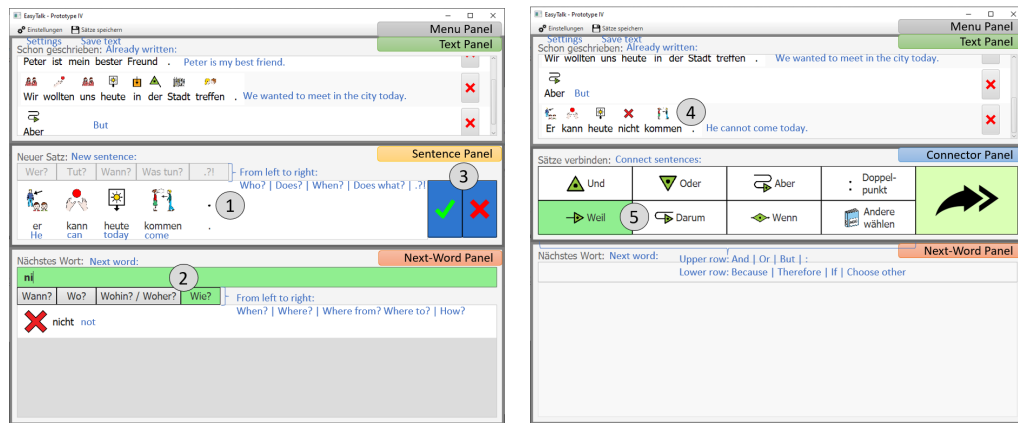
1. Component: Subject
 First Word Form added: *kleine* ‘small’
 Valid (incomplete) example Tree: $\{adj, n\}$
 Resulting Feature Set:
 $\{\{female, nominative, plural\}, \{male, nominative, plural\}, \{neuter, nominative, plural\}\}$
2. Component: Direct Object
 First Word Form added: *kleine* ‘small’
 Valid (incomplete) example Tree: $\{adj, n\}$
 Resulting Feature Set:
 $\{\{female, accusative, plural\}, \{male, accusative, plural\}, \{neuter, accusative, plural\}\}$
3. Component: Subject
 First Word Form added: *Ich* ‘I’
 Valid (complete) example Tree: $\{pers.pron\}$
 Resulting Feature Set: $\{\{firstPerson, nominative, singular\}\}$.

Next, the routine iterates all valid Trees (Lines 13 - 19). For each Tree, it determines the Word Type of the next empty Lexical Frame (Lines 14,15). According to the Word Type, it maps the Features to the corresponding *Inflection Codes*, which are used to extract Word Forms from the dictionary. For example, the Inflection Code used for Nouns in the singular nominative case is *nS*; the Inflection Code for perfect participle Verb Forms (e.g., *gegangen* ‘gone’) is *pA*. The Inflection Codes for all subsequent Lexical Frames of all valid Trees are then added to the Component Filter currently being computed (Line 18). Finally, the routine returns the Component Filters for all possible following Components.

In the next step, the Sentence Builder then passes the Component Filters to the Dictionary, which, in turn, forwards the corresponding Word Forms of the Lexicon to the User Interface. Now, the system awaits the next user action.

After this detailed description of the architecture and theoretical aspects of the NLG component of *EasyTalk*, the following sections describe the system depicting its user interface.

5.3 System introduction



(a) Step 1-3.

(b) Step 4 and 5.

Figure 5.7: Two consecutive snapshots outlining the basic writing process of *EasyTalk* in five steps:

- 1) Choose the sentence type by selecting the corresponding punctuation mark in the empty Sentence Panel. “./?!” is provided (click button to change; declarative is the default);
- 2) Add words to the Sentence Panel by selecting word by word from the suggestion list in the Next-Word Panel offering inflected forms;
- 3) Choose the checkmark button to finish the sentence (X deletes last word). In parallel, the Sentence Panel flips to the Connector Panel (cf. step 4);
- 4) The completed sentence moves to the Text Panel (it can be read out loud and/or exported for further use);
- 5) Select the connection of the next sentence in the Connector Panel. Consecutively, the system flips back to the Sentence Panel for the next sentence (cf. step 1).

Before we detail on the individual features of *EasyTalk*, we introduce its interface and outline the basics of the writing process in *EasyTalk* in five steps as depicted in Figure 5.7.

EasyTalk's user interface comprises three layers embedded in the *Menu Panel* (labeled gray): The *Text Panel* (labeled green) is located at the top. In the middle, the *Sentence Panel* (labeled yellow) and the *Connector Panel* (labeled blue) are shown in alternation (see the two snapshots in Figure 5.7 depicting that either the Sentence Panel (cf. Figure 5.7a) or the Connector Panel (cf. Figure 5.7b) is active). The *Next-Word Panel* (labeled orange) is located at the bottom.

As seen in Figure 5.7, all words in *EasyTalk* can be supplemented with a symbol from the users' preferred AAC symbol set. All words and commands in *EasyTalk* can be read aloud on demand.

Starting a new sentence, the user chooses the sentence type by selecting the corresponding punctuation mark in the empty Sentence Panel (step 1). *EasyTalk* supports declarative, indicative and imperative sentences. Declarative sentences, indicated by ‘.’, are the default. To change the sentence type, the user clicks the button labeled with the punctuation marks ‘./?!’. The chosen punctuation mark automatically remains sentence-final at the end of the sentence during the process of typing the sentence. In Figure 5.7a, the user has already written two sentences (‘Peter is my best friend. We wanted to meet in the city today.’) and chosen a sentence connector (‘But’). Currently, the user is adding words to the sentence, ‘He can come today’, depicted in the Sentence Panel (step 2) by selecting word by word from the suggestion list in the Next-Word Panel that offers inflected forms for the current sentence component. In our example, by selecting the modifier cue *Wie? /* ‘how’ (displayed in green), the user decided to add *nicht* ‘not’ as a modal specification that the reader should know. At the point shown, the user has typed “*ni*” in the text-input field of the Next-Word Panel. Accordingly, a choice list presenting only modal expressions is retrieved from the lexicon matching the current input string. In the figure, one item qualifies as matches for “*ni*”. In case of multiple matches, the user navigates the completion list by scrolling to the intended form. Once the user selects the word form, it is automatically inserted at the correct position in the sentence (i.e., before the verb *kommen* ‘to come’ at the end of the sentence). To delete the last word, the user clicks the X-button at the right-hand side of the Sentence Panel. Finishing the sentence currently in Figure 5.7a by clicking the checkmark-button (step 3) updates the Text Panel in Figure 5.7b): The completed sentence is attached to the previously written text displayed in the Text Panel (step 4). (It can be read out loud and/or exported for further use (cf. Section 5.6.1).) In parallel, the Sentence Panel flips to the Connector Panel prompting the user so select the connection of the next sentence (step 5). Once the user chose a connector, the Text Panel is updated and the system flips back to the Sentence Panel in order to start the next sentence (cf. step 1).

Next, we detail on the individual writing features of *EasyTalk*.

5.4 Text-writing support

This section describes the text-writing support in *EasyTalk*. First, we detail on the letter- and word-level support (cf. Section 5.4.1). Then, we describe the sentence-level support (cf. Section 5.4.2) and finally, the text-level support (cf. Section 5.4.3).

5.4.1 Letter- and word-level support

At the letter- and word-level, *EasyTalk* supports the user with customizable communication symbols, read-aloud functionality and word form suggestions. We describe these aspects in the following.

5.4.1.1 Symbol support

In Augmentative and Alternative Communication, it is an well-established technique to combine symbols with words (cf. Section 5.3). In *EasyTalk* all words can be augmented with personal symbols. By default, *EasyTalk* uses *ARASAAC symbols*³ – a free-to-use symbol collection offering more than 11.000 descriptive and schematic symbols ARASAAC (2022). For AAC users who are on the way to literacy, it is essential that their familiar symbols are available to them. Accordingly, in *EasyTalk* the symbols can be easily customized (see Section 5.6.2 for details on the process). In case the symbols feel distracting to (presumably non-AAC) users, the symbols can be hidden. We assume, however, that even users who are not familiar with AAC symbols may find the combination of symbols and words helpful in recognizing words more easily – in particular, unknown or longer words. In case of irregular word forms with a spelling that differs from the base word (e.g., *I read* ‘I read’ vs. *she reads* ‘she reads’), the symbols can facilitate the recognition of the base word for users with low spelling skills.

5.4.1.2 Read-aloud functionality

Read-aloud features are a common tool for software accessibility. Like the symbol-based and text-based systems presented in Chapter 3, *EasyTalk* offers a read-aloud function for all text elements using the text-to-speech module *MaryTTS*⁴ licensed under the *LGPL*⁵. To give their users a personal voice, read-aloud functions for *Speech Generating Devices* (SGDs) for people with CCN usually provide many customization options, such as tone of voice or speech rate. Despite our awareness of the its importance, this feature is is less pronounced than in professional SGDs, as we focus on the core functionality of our *EasyTalk* prototype. Nevertheless, *EasyTalk* offers basic customization possibilities, which are described in Section 5.6.2.

³ARASAAC homepage: www.arasaac.org

⁴MaryTTS website: mary.dfki.de

⁵Information on the LGPL: www.gnu.org/licenses/lgpl-3.0.de.html

5.4.1.3 Suggestion list

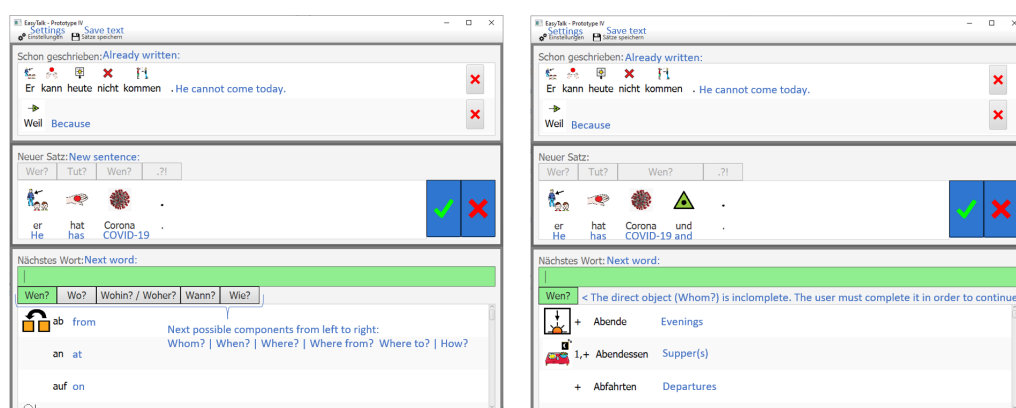
As described in Sections 5.2 and 5.3, *EasyTalk* provides the user with correctly inflected word forms for any sentential context. On the word level, the suggestions prevent spelling errors and provide users with low spelling skills an sense of ease and confidence in choosing word forms. On the sentence level, they guarantee a grammatically correctness. We detail on the sentence-level support in the following section.

5.4.2 Sentence-level support

In this section, we elaborate on the inter-sentential linguistic support of *EasyTalk* introduced in Section 5.2. A particular highlight of the following descriptions comprises arguments for an adequate user interface UI that, in addition to supporting low literacy skills, must compensate for factors such as working memory deficits within our heterogeneous target user group. First, Section 5.4.2.1 describes the sentence-production process in interaction with the user. Then, Section 5.4.2.2 details on verbs with separable verb prefix. Section 5.4.2.3 presents the mechanism for writing complex verb constructions. Section 5.4.2.4 is dedicated to writing questions and Section 5.4.2.5 to writing in the past tense. Finally, Section 5.4.2.6 presents the free writing mode of *EasyTalk*.

5.4.2.1 Fast and correct ELS sentence production

In this section, we describe how the system supports the user in writing a sentence in an easy and intuitive step-by-step manner. First, the system presents the cues for the sentence-initial components according to the canonical word order of the selected sentence type. In a declarative main clause presupposing SVO word order, the subject (Who?) is entered first. (A question begins with either a verb or an interrogative pronoun followed by a verb – cf. Section 5.4.2.4). Based on the subject's number and person features, the system provides only correctly inflected verb forms for any typed word prefix managed by the cue *Tut* 'does' in the Sentence Panel. After the verb is entered, the system keeps track of the overall valency restrictions/arguments provided in the lexicalized grammar, and every word form is supplemented with the appropriate syntactic structure. In the Next-Word Panel, the system presents – from left to right – the cues for the (possibly obligatory) arguments followed by adjunct/modifier cues. This default execution strategy based on the common German word order offers guidance to the user. The left-most cue is active and highlighted green. Accordingly, the suggestion list in the Next-Word Panel displays word forms matching the component.



(a) The Next-Word Panel offers the next possible components.

(b) The Next-Word Panel shows only the current, incomplete component.

Figure 5.8: Two snapshots from *et* illustrating how the system communicates the next possible components to the user. In Snapshot (a), the user has started the direct object (‘Whom?’) with ‘COVID-19’. As the direct object is complete, *EasyTalk* offers several options to continue the sentence in the Next-Word Panel according to the Sentence Plan. The direct object is active and highlighted green. In Snapshot (b), the user has added the word ‘and’ to the direct object. Now, the component is in an incomplete state. Therefore, the corresponding ‘Whom’-cue is the only one shown to the user in the Next-Word Panel. To continue the sentence, the user must complete the component first. To achieve this, they can either add to the direct object or delete the last word (‘and’) to return to the previous complete state.

However, the user can enter the constituents freely in any desired order allowing for variety, personal preference, and emphasis of components. To switch the component, the user tabs through the list of cues or clicks a cue to activate it. Incompletely filled valency frames cannot be finished and moved to the Text-Panel; i.e., only correct sentences can be typed in *EasyTalk*. (In case the user attempts to finish an incomplete sentence, the system activates and highlights the cue for the (first) missing obligatory argument.) Figure 5.8 continues the text written in Figure 5.7 and shows this by example.

In Figure 5.8a, the user has entered three words in the Sentence Panel (*er hat Corona* ‘he has COVID-19’). For the cues (‘Who?’, ‘Does?’, ‘Whom_{acc}?’) the system has collected the following information so far:

Wer? ‘who_{nom}’ = **Subject:** *Er*_{PPER:3rdPers,Sing,Nom}
Tut? ‘does’ = **Verb:** *hat*_{VVFIN:3rdPers,Sing,Pres,Ind,ActiveVoice}
Wen? ‘whom_{acc}’ = **Direct object:** *Corona*_{NN:Neut,SING,Acc}

According to the system’s internal representation of the sentence (cf. Section 5.2.5), all obligatory grammatical functions (subject, verb and direct object) are

Table 5.2: Examples for the two states of a component in *EasyTalk*: being complete and/or being maximal.

Component	English Translation	State
Corona	COVID-19	complete
Corona und	COVID-19 and	not complete
Corona und Fieber	CODVID-19 and fever	complete
Corona und hohes	CODVID-19 and high	not complete
Corona und hohes Fieber	COVID-19 and high fever	complete
Corona und hohes Fieber von der Infektion	COVID-19 and high fever from the infection	complete and maximal

filled and the sentence is currently in a complete state). At this point, the user can either finish the sentence or add to it. The system presents the user with all options to continue the sentence by displaying the wh-cues for the next possible sentence components at the top of the Next-Word Panel. To ensure that each component is complete in itself and does not exceed its maximum size, the system updates an internal tree structure after each word form added (cf. Section 5.2.6). In Figure 5.8a, the user can still add to the direct object in the Sentence Panel. To communicate this to the user, its cue, *Wen?* ‘Whom_{acc}?’, is displayed first in the list of next possible sentence components in the Next-Word Panel and highlighted green to indicate that the suggestion list displays corresponding word forms. Alternatively, the user could start a new component by selecting another wh-cue in the Next-Word Panel. In this case, *EasyTalk* would update the list of word form suggestions accordingly. In the second snapshot of the example shown in Figure 5.8b, the user has added the conjunction ‘and’ to the direct object (now: *Corona and* ‘COVID-19 and’). At this point, the direct object is incomplete. The user has to complete it before proceeding with a new component or ending the sentence. I.e., they can either add to the direct object or delete the last word(s) to return to a previous complete state. To signal this to the user, *EasyTalk* shows only the ‘Whom_{acc}’-cue in the Next-Word Panel. Once a component has reached its maximum size, it cannot be expanded further and its cue disappears from the Next-Word Panel. For a better understanding, Table 5.2 illustrates different options for expanding the direct object in Figure 5.8b along with their states.

Modifier/adjunct cues are facultative. Displaying them in the Next-Word Panel should prompt the user to supplement the sentences properly with audience-design information, such as the time and place of an event (cf. the example sentence of Figure 5.7: *Wir wollten uns heute_{When?} in der Stadt_{Where?} treffen.* ‘We wanted to meet in the city_{Where?} today_{When?}.’; for a strategy teaching users how to write for an audience in *EasyTalk*, see Section 5.5.)

The typing speeds of all users, not only target users with little computer skills, are supported by prediction/completion lists (cf. typing on reduced keyboards on cell phones). The structure of the Next-Word-Panel borrows this concept. For any string prefix – even an empty one – *EasyTalk* displays a suggestion list according to the active cue. In Figures 5.8a and 5.8b, the text input field is empty. Accordingly, all word form suggestions are displayed in their alphabetical order. In Figures 5.8a, the list shows word forms matching the context of a direct object containing a noun ('COVID-19'). The three word forms visible – *ab* 'from', *an* 'at', *auf* 'on' – can be used to add an prepositional attribute to the direct object. In Figure 5.8b, the suggestion list displays all word forms available to continue the direct object after the conjunction ('and') has been added. (For details on the computation of the suggestions according to the grammatical context, see Algorithms 1-4 in Section 5.2.) Words not visible in the suggestion list can be accessed by scrolling through the list, or by starting to type a word's prefix – given that the user knows the spelling. To select a word form, the user navigates to the desired list item and confirms the selection. Directly pressing 'Enter' quickly selects the topmost list item. *EasyTalk*'s inflected suggestions speed up typing by unifying the two-stage process of selection and manual morphological adaptation. Hence, not only is syntactic correctness maintained, but spelling errors are also prevented providing users with low spelling skills an additional sense of ease and confidence.

5.4.2.2 Verbs with separable verb prefix

In German, verbs with separable verb prefix (SVP) are common. In a declarative main clause, the SVP is positioned at the end of the sentence.

Figure 5.9 shows how *EasyTalk* handles verbs with SVP in three consecutive snapshots continuing the writing session from Figures 5.7 and 5.8: In the first snapshot (5.9a), the user has written the subject Ich 'I' in the Sentence Panel and has typed the prefix 'ru' in the text-input field. The Next-Word Panel shows two inflected verb forms for the first person singular: *rufe* of the lemma *rufen* 'to shout' and *rufe an* of the lemma *anrufen* 'call'. As depicted by the example, *EasyTalk* displays inflected verb forms together with their SVP in the suggestion list in the Next-Word Panel. Adding the verb its SVP in one step reduces the mental load on the side of the user, who does not have to remember to add the SVP separately at the end of the sentence. In the second snapshot (5.9b), the user has selected the verb form *rufe an* with SVP. *EasyTalk* has added two units to the Sentence Plan – one for the inflected verb form (*rufe* 'call') and one for its SVP (*an* 'to'). The verb and its SVP are visually linked by displaying the same symbol (cf. the cellphone in Figure 5.9). The SVP is automatically moved to its sentence-final word order position every time the user adds more word forms to the sentence. This is depicted in the third snapshot (5.9c): the user has added



Figure 5.9: Three consecutive snapshots showing how *EasyTalk* handles verbs with separable verb prefix. In (a) the user is about to choose the verb form *rufe an* ‘call (to)’. In (b), the system added two units to the Sentence Panel: one for the verb form and one for its SVP. The user is about to add the direct object *ihn* ‘him’. In (c), the direct object has been inserted before the SVP in the Sentence Panel and the user is about to select the time modifier *nachher* ‘später’, which will also be added before the SVP in the Sentence Panel.

the direct object *ihn* ‘him’, which was inserted between the verb and its SVP. Now, they are about to add the time modifier *nachher* ‘later’, which will be added between the direct object and the SVP. By automatically updating the SVP’s position, *EasyTalk* keeps the word order correct.

5.4.2.3 Complex verb constructions

A challenge with respect to the linear order of building a sentence is presented by complex verb constructions such as verb clusters including auxiliary, modal or complement taking verbs (such as, e.g., *gehen* ‘to go’). In order to satisfy the

valency requirements of these verbs, our system asks the users if they want to add a another verb and updates the list of arguments accordingly. Striving for a correct and complete clause, we hereby knowingly urge the user to think in an unusual word order: In German, the arguments of the full verb often precede the verb, thus our system needs to know this verb in order to determine and support the formulation of its extensions. The grammar rules permit recursion here, i.e., the user can produce a verb chain (e.g., *Ich will einkaufen gehen*. ‘I want to go shopping.’). As soon as the verb chain ends, the user fills the arguments of the last entered verb in the usual manner.

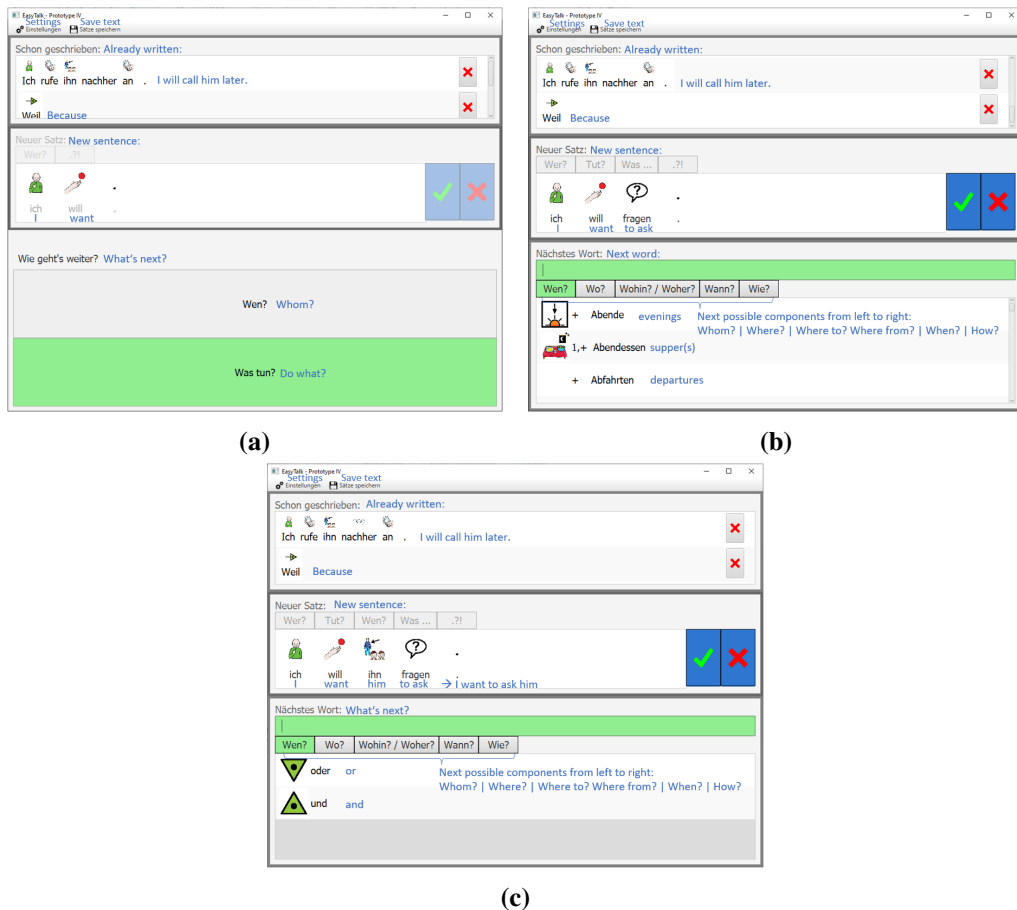


Figure 5.10: Three snapshots showing how *EasyTalk* handles complex verb constructions. In (a), the user has just entered a modal verb. The system needs to know if the user wants to fill its arguments or add a second verb. It presents the user a decision dialog with both options. In (b), the user chose to add a second verb in the infinitive form and is about to fill its arguments. In (c), the user added the direct object and *EasyTalk* automatically moved the second verb to its sentence-final position.

In continuation of our ongoing writing session, Figure 5.10 illustrates the process of writing a complex verb construction in *EasyTalk* by the example of the modal verb *wollen* ‘to want’ in three snapshots. In the first snapshot, Figure 5.10a, the user just selected the inflected verb form *will* ‘want’ from the Next-Word Panel, and the system added it to the Sentence Panel. The natural language generation component of *EasyTalk* (cf. Section 5.2) computed two options for continuing the sentence. To be able to offer correct suggestions, it requires the user to make a decision: The user may either continue with the argument of the verb (top option, cued *Wen?* ‘Whom_{acc}?’) or choose to add another verb (bottom option, cued *Was tun?* ‘Do what?’). The system presents both options to the user in a decision dialogue which temporarily replaces the word list in the Next-Word Panel. The currently active choice is highlighted in green. Analogous to the word list of the Next-Word Panel, the user can scroll through the options and make a selection by clicking or pressing enter. In Figure 5.10b, the user has chosen the verb *fragen* ‘to ask’, ending the verb chain. The user is now about to fill the argument of the second verb’s valency frame in the form of the direct object. For each next word form added to the sentence, *EasyTalk* automatically adjusts the position of the sentence-final infinitive verb as depicted in Figure 5.10c. By highlighting the current constituent in the Next-Word Panel, the user is not distracted by the verb(s) at the end of the sentence. For another example of this process, see Steinmetz and Harbusch (2020).

5.4.2.4 Writing questions

Now, we detail on writing questions in *EasyTalk*. To write a question, the user needs to change the sentence type to ‘question’ prior to adding word forms to a new sentence. To do so, the user clicks the punctuation-mark element in the empty Sentence Panel. When the question mark is displayed, *EasyTalk* presents the user the word order of interrogative sentences. The user may start a question with an interrogative pronoun (e.g., ‘who’, ‘when’, or ‘where’) or a verb – in case the choice of an interrogative pronoun is omitted. In some cases, the chosen interrogative pronoun replaces an argument or adjunct – e.g., ‘who’ replaces the subject. After adding the verb, *EasyTalk* requires the user to enter the subject (unless the user started the question with ‘who’). The choices in the Next-Word-Panel are restricted to fulfill the subject-verb-agreement. After adding the subject, the user enters all arguments and adjuncts in the usual manner.

Figures 5.11 and 5.12 show two examples of the question writing process in *EasyTalk*. In Figure 5.11, the user starts the question with an interrogative pronoun; in Figure 5.12, the question is started with a verb. In the following, we consider both examples in detail.

In the first example, the user is writing the question *Wie kann ich dir helfen?*



Figure 5.11: Details on writing a question starting with an interrogative pronoun in *EasyTalk*. In (a), the user has chosen the sentence type ‘question’. In (b), the user has added an interrogative pronoun and is about to choose the predicate, which determines the number of the subject in (c) to fulfill the subject-verb agreement. In (d), the user continues the sentence in the usual manner.

‘How can I help you?’. Snapshot 5.12a shows the interface of *EasyTalk* after the user has selected the question mode in the Sentence Panel. *EasyTalk*’s natural language generation component updates the next possible components accordingly (cf. Section 5.2): The Next-Word Panel offers the user to start the question with an interrogative pronoun, cued ‘?’, or with a verb, cued *Tut?* ‘Does?’. (In a declarative sentence, the user would have to fill obligatory subject first). In Snapshot 5.12b, the user has added the question word *wie* ‘how’ to the Sentence Panel. At this point, *EasyTalk* requires the user to choose the obligatory verb. Unlike in declarative sentences, the subject-verb-agreement is not yet determined. Consequently, the suggestion list offers verb forms with inflections for different

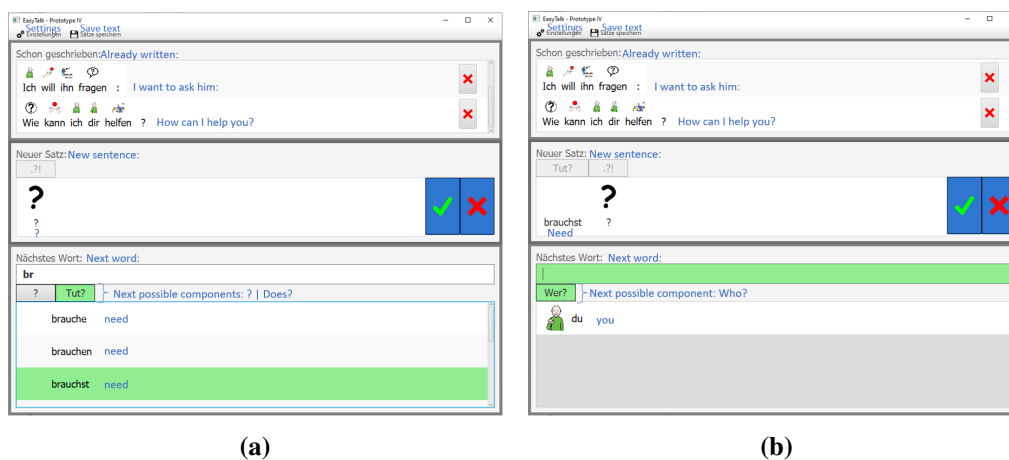


Figure 5.12: Details on writing a question starting with a verb in *EasyTalk*. In (a), the user has selected the verb cue and is about to choose a predicate in the second person singular. Next, in (b), the user must select the subject. In the example, only the personal pronoun ‘you’ can fulfill the subject-verb-agreement. After adding subject, the sentence is continued in the usual manner.

grammatical numbers and persons. In Snapshot 5.12b, the user has entered the prefix “k” in the input filter in the Next-Word Panel. We see two inflected forms of the verb *können* ‘to can’ at the top of the suggestion list: *kann* (first or third person singular, indicative mode, present tense) and *kannst* (second person singular, indicative mode, present tense). Further down the list, all other forms of the verb (e.g., *können* can for we, you and they or simple past forms, like *konnte*, *konnten*, ...) are displayed according to the alphabetical order. After adding the verb to the sentence, the user must fill the subject. To fulfill the subject-verb agreement, the Next-Word Panel displays word forms to form a subject in first or third person singular. Additionally, the internal tree structure (cf. Section 5.2) of the subject is limited, i.e., the user cannot complement the subject with ‘and’ or ‘or’. In Snapshot 5.11c, the user is about to select the personal pronoun *ich* ‘I’. After completing the subject, the sentence is continued analogously to declarative sentences. First, the modal verb is handled with the mechanism for verb clusters described in Section 5.4.2.3. Then, the user fills the sentence components in the usual manner. In Snapshot 5.11d, the user has added the infinitive verb *helfen* ‘to help’ to the sentence and is about to add the personal pronoun *dir* ‘you’. For each word form added, *EasyTalk* keeps the correct word order in the Sentence Panel.

In the second example, the user starts the question *Brauchst du etwas aus dem Laden?* ‘Do you need anything from the store?’ with the verb *brauchen* ‘to need’. To do so, the user has activated the verb filter in the Next-Word Panel in Snapshot 5.12a. The suggestion list presents the user with different inflected verb forms

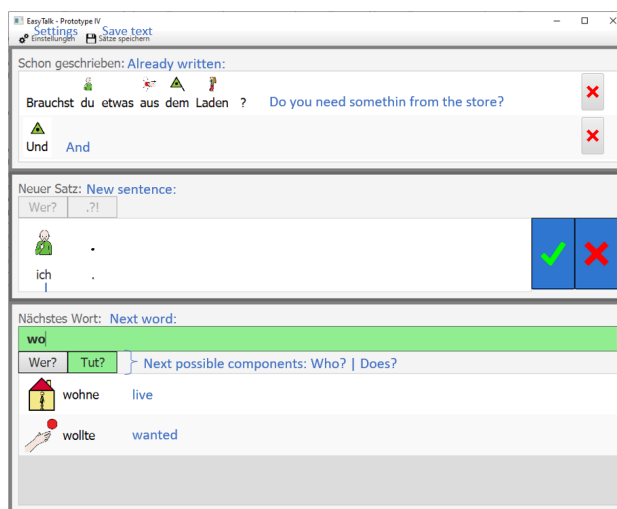


Figure 5.13: Writing in the preterite past in *EasyTalk*: For modal verbs only, the suggestion list contains verb forms in the preterite past.

for different grammatical persons and numbers matching the filter prefix “br”. The user chooses the 2nd person singular, *brauchst*. In this case, the subject-verb agreement is unambiguous. Accordingly, the personal pronoun *du* ‘you’ is the only available choice for the subject in Snapshot 5.12b. After choosing the subject, the user is presented with all components in the usual manner.

5.4.2.5 Writing in the past tense

Supporting the constructions of Extended Leichte Sprache, *EasyTalk* allows the user to write in present tense, present perfect tense and, for modal verbs, preterite past. In the following, we explore writing in preterite past and present perfect tense in *EasyTalk* by example.

Preterite past In Extended Leichte Sprache, the preterite past is only licensed for modal verbs – due to its frequent use in spoken language and LS texts (cf. Section 2.4). Using preterite past in *EasyTalk* is straightforward: As Figure 5.13 shows, the inflected verb forms are included in the choice list of the Next-Word Panel for the verb component. The mechanism for complex verb constructions described in Section 5.4.2.3 applies to modal verbs in the present and simple past tense.

Present perfect tense In German, the present perfect tense is formed by an auxiliary verb in the present tense in combination with a perfect participle. Therefore,



Figure 5.14: Three snapshots illustrating how to write sentences in the present perfect tense in *EasyTalk*. In (a), the user has just added an auxiliary verb to the sentence. *EasyTalk* presents a decision dialog with the options to continue in the present or present perfect tense. In (b), the user has chosen the present perfect tense, and *EasyTalk* offers suggestions for participle verb forms. In (c), the user fills the arguments of the finite verb form. *EasyTalk* keeps the participle at its correct sentence-final position.

after choosing forms of *haben* ‘to have’ or *sein* ‘to be’, *EasyTalk* presents the user with a decision dialog in order to select the tense of the sentence. The proceeding, shown in Figure 5.14, is analogous to the formation of complex verb constructions (see Section 5.4.2.3): In the decision dialog, the user chooses between the alternatives *Jetzt?* ‘Now?’ for present tense, or *Früher?* ‘Before?’, as shown in Snapshot 5.14a. If the user has chosen *Früher?*, as in Snapshot 5.14b, *EasyTalk* prompts the user to choose a perfect participle in the Next-Word Panel. Afterward, the user adds to the sentence in the usual manner. The system automatically keeps the participle at its correct sentence-final position, as depicted in Snapshot 5.14c.

5.4.2.6 Free writing mode



Figure 5.15: Four snapshots showing the *free writing mode* of *EasyTalk*. In (a), the default mode is shown in contrast to the free writing mode in (b). In (c) and (d), the last word added to the Sentence Panel can be assigned to different sentence components. The system presents the user with the respective options in a decision dialog.

In our previous examples, *EasyTalk* was used in the guided default mode (cf. Section 5.2). In this mode, the system presents the user with an active *wh*-cue in the Next-Word Panel at all times. To change the active sentence component, e.g., to deviate from the proposed default sentence order, the user has to actively select another cue in the Next-Word Panel. The active sentence component is communicated explicitly between the user and the system at all times. The word form suggestions are limited to fit the context of the active component. As a trade-off, the user needs to actively select the next cue to start the next sentence component before the current one is complete and maximal (i.e., all nodes of a

component tree are filled; cf. Table 5.2 for examples or Section 5.2 Algorithm 2 for the procedure). Figure 5.15 illustrates this using four examples. In the first example, depicted in Figure 5.15a, *EasyTalk*'s default writing mode is active. The user has started the sentence with *ich* 'I'. After entering the personal pronoun, *EasyTalk*'s internal ELS tree structure allows the user to add a conjunction to the subject (e.g., *Ich und mein Freund* 'I and my friend'). Alternatively, the user can activate *Tun?* 'Does?' and continue with the verb. The latter is much more likely in simple texts than adding a complement to the personal pronoun (cf. the corpus study of LS texts by Harbusch and Steinmetz (2022)).

To speed up writing, *EasyTalk* offers the *free writing mode* that can be activated in the system settings (cf. Section 5.6.2). In the free writing mode, *EasyTalk* also proposes all the next possible components at the top of the Next-Word Panel. However, the system does not select a component by default. The suggestion list contains all word forms that match one of the displayed next components. The user can either activate a cue to restrict the list to a wh-cue or directly start typing in the text-input field to search for a word form. This is illustrated in Snapshot 5.15b: In this snapshot, the free writing mode is active. Like in Snapshot 5.15a, the user has started the sentence with *ich* 'I'. However, no cue is highlighted green in the Next-Word Panel – i.e., no cue is active in Snapshot 5.15b. The word list contains both complements for the subject and inflected verb forms. The user is able to directly look for a verb form by scrolling or typing a prefix in the text-input field without having to activate the verb-cue first.

When the user adds a word form in free writing mode, *EasyTalk* automatically assigns it to the corresponding sentence component. The assignment process uses feature unification in the context of the sentence (cf. Section 5.2, Algorithms 1 and 3). It computes, which of the next possible components matches the word form. For certain word types, such as verbs, adverbs, or personal pronouns (except for *sie*, which can be both third person singular ('she') and third person plural ('they')), the result of the assignment process is generally unambiguous. For other word types, however, grammatical ambiguities can occur. Two examples of this are shown in Snapshots 5.15c and 5.15d: In Snapshot 5.15c, the user has so far written the sentence *Wir können uns bald treffen* 'We can meet soon' and just added the preposition *in* 'in'. Previously, the system could uniquely identify the component affiliation of all added word forms. However, the *alternating preposition* 'in' can be followed by either dative case (here, for a place modifier) or accusative case (here, for a direction modifier) in German. To be able to create the component the user had in mind, *EasyTalk* presents the user a decision dialog with both available options communicated in the form of wh-cues. Snapshot 5.15d illustrates another example for a decision dialog using the sentence *Ich habe eine Karte für ihn geschrieben*. 'I have written a card for him.' written earlier in our writing session (see Section 5.4.2.5). At the point shown, the user has just filled

the direct object with *eine Karte* ‘a card’ and lastly selected the preposition *für* ‘for’. At this point of the sentence, the free writing mode would invoke a decision dialog for the first time: *EasyTalk* requires the user to decide, whether to add to the direct object (which was the case in our writing session), to add a how-component (e.g., the purpose *für eine gute Sache* ‘for a good cause’), or a time-component (e.g., the timeframe *für eine Stunde* ‘for an hour’)⁶

In summary, the free writing mode allows the user to write with (possibly) fewer selections which speeds up the writing process. In free writing mode, the user can choose suitable word forms for all currently available cues in the Next-Word Panel. They can either select a cue on demand to restrict the list of suggestions in the same way as the default *guided writing mode*, or select a word form directly. The latter requires the user to know which word form fits the desired component. If the user selects a word form without activating a cue, the system automatically assigns the word form to its corresponding component. If a word form cannot be assigned unambiguously, *EasyTalk* uses decision dialogues prompting the user to identify the desired sentence component. The free writing mode can be activated in the system settings depending on the user’s skill level and personal preference.

5.4.3 Text-level support

Writing support in *EasyTalk* is not restricted to intra-sentential items. Text consisting of a series of simple sentences with canonical Subject-Verb-Object order lacks flow, and the writer’s thoughts are only partially communicated. As in the conceptualizer of a natural language generation system (cf. Section 5.1), RST-like cues relating the individual sentences should verbalize the user’s communicative goal. As mentioned in Chapter 1, techniques for exemplifying RST relations are learned in exercises for complex clause construction in school. However, complex clauses with informative conjunctions are not available in either *Leichte Sprache* or *Extended Leichte Sprache*. Yet, the standard German writers of LS texts often resort to subordinate clauses. Studies into the LST corpus (Harbusch and Steinmetz, 2022) comprising more than 29,000 sentences from a variety of LS text from the internet describe another problem. In order to provide text coherence, main declarative clauses deviate in 50 percent of the cases from the SVO order – although any deviation from SVO word order is very hard to understand by our target group (Bock, 2019).

We suggest a very easy (E)LS-conform method to provide coherence cues.

⁶We are aware, that the question words *Wofür?* ‘What for?’ and *Wie lange?* ‘For how long?’ would be more precise here. For reasons of simplicity for our target users, however, we decided to add these modifiers to the *Wie?*- and *Wann?*-cue, respectively.

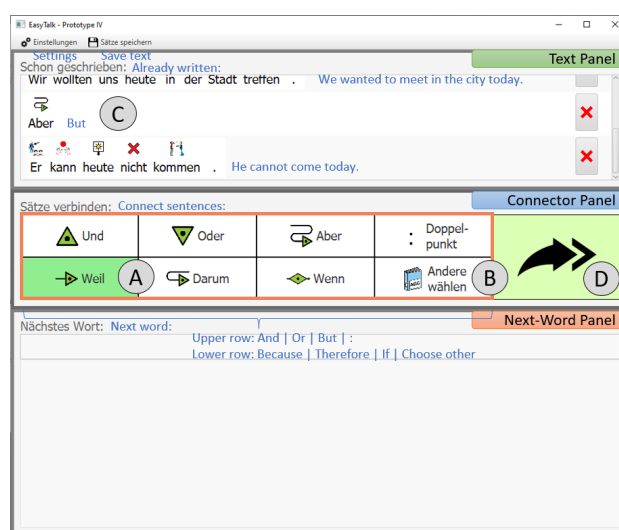


Figure 5.16: Connecting sentences via the Connector Panel of *EasyTalk*; the currently active connector is highlighted green (A). Activating *Andere wählen* ‘Choose other’ (B) opens a choice list of more connector options in the Next-Word Panel. Chosen connectors are added to the Text Panel in a separate line (C). Operating the arrow-button (D) omits the choice of a connector.

The idea is inspired by the German *weil*-V2 phenomenon in spoken German (the subordinating conjunction *weil* ‘because’ is followed by a clause with main-clause V2-word order; cf. Reis (2013) for a thorough survey). Based on audio and transliteration data from spoken German, Kempen and Harbusch (2016) argue that speakers start a new sentence after having uttered the conjunction. We reason that the concept of going on with a main clause after any conjunction or a sentential adverb in the frontfield is a feasible generalization that circumvents subordinating clauses and focused elements in the frontfield position in German without losing the information carried by these items. Looking at this claim from a sentence-planning perspective, any abstract relation known from the Rhetorical-Structure Theory becomes available as a sentence connector between two main clauses. The resulting text reflects the writer’s conceptual message. Thus, the overall discourse structure is conveyed much better than by unconnected sequences of main clauses (cf. the complete text of our writing session Table 5.3 with highlighted connectors preserving the constraints of (E)LS).

Via the Connector Panel introduced in Figure 5.7b, all abstract RST-relations to express the communicative goal of a sentence are made accessible by using an intuitive wording from the target users’ vocabulary (e.g., REASON = because). Figure 5.16 shows the selection process in detail. The menu provides seven connectors (framed orange) – recommended by Netzwerk Leichte Sprache (2013)(cf.

Table 5.3: The complete text of our ongoing *EasyTalk* example writing session. Left column: German text. Right column: English translation. Red: Chosen connectors.

<i>Peter ist mein bester Freund.</i>	‘Peter is my best friend.’
<i>Wir wollten uns heute in der Stadt treffen.</i>	‘We wanted to meet in the city today’
<i>Aber</i>	‘But’
<i>Er kann heute leider nicht kommen.</i>	‘Unfortunately, he cannot come today.’
<i>Weil</i>	‘Because’
<i>Er hat Corona.</i>	‘He has COVID.’
<i>Darum</i>	‘Therefore’
<i>Ich mache mir viele Sorgen.</i>	‘I am very worried.’
<i>Darum</i>	‘Therefore’
<i>Ich rufe ihn nachher an.</i>	‘I will call him later.’
<i>Weil</i>	‘Because’
<i>Ich will ihn fragen:</i>	‘I want to ask him:’
<i>Wie kann ich dir helfen?</i>	‘How can I help you?’
<i>Brauchst du etwas aus dem Laden?</i>	‘Do you need anything from the store?’
<i>Und</i>	‘And’
<i>Ich wollte ihn aufheitern</i>	‘I wanted to cheer him up’
<i>Darum</i>	‘Therefore’
<i>Ich habe eine Karte für ihn geschrieben.</i>	‘I have written a card for him.’
<i>Ich hoffe:</i>	‘I hope:’
<i>Er fühlt sich bald besser.</i>	‘He will feel better soon.’
<i>Und</i>	‘And’
<i>Wir können uns bald wieder in der Stadt treffen.</i>	‘We can meet in the city again soon.’

Table 2.2, Rule 4.3) – for direct access.⁷ We group the elements in the menu according to conjunction type. In the upper row, the coordinating conjunctions *und* ‘and’, *oder* ‘or’, and *aber* ‘but’, and the colon are provided. We realize that the colon is highly ambiguous in LS texts; however, its use is widespread (Maaß and Bredel, 2016, p. 254). We therefore offer this choice to prevent users from having to search for this option. In the second row, the user is presented with the subordi-

⁷In accordance with suggestions made by AAC experts (cf. Section 6.1), we restrict the choice to those forms widely used under LS rules Netzwerk Leichte Sprache (2013) to avoid overtaxing the user.

nating conjunctions *weil* ‘because’ and *wenn* ‘if’, the adverb *darum* ‘therefore’, and a button *Andere wählen* ‘Choose other’. The currently active choice is highlighted green (Figure 5.16 (A)). Initially, we leave the Next-Word Panel empty to avoid additional reading during the decision making for a connector. However, in their corpus study into LST, Harbusch and Steinmetz (2022) observed further variation for sentence connectors. Hence, more advanced users can operate *Andere wählen* ‘Choose other’ (Figure 5.16 (B)) to browse through all conjunctions. For consistency and overall ease of use of the system, the Next-Word Panel then provides a list of conjunction choices with the same selection options as for word forms in sentence typing. In case the option selected – either by button or in the Next-Word Panel – is a word, it is added as a separate line at the end of the Text Panel (Figure 5.16 (C)). The colon is appended to the last sentence in the Text Panel and replaces the previously written punctuation symbol (cf. the sentences in Table 5.3). Activating the arrow button on the right of the Connector Panel skips the selection of a connector (Figure 5.16 (D)).

5.5 Teaching text production in a writing workshop

So far, we have illustrated how users can use *EasyTalk* to write and connect sentences in ELS. On the sentence level (cf. Section 5.4.2.1), *EasyTalk* takes readership-design aspects into account by reminding the user to add information like place/time of an event by presenting wh-question cues. On the discourse level (cf. Section 5.4.3), it prompts the user to add coherence specifications to express the communicative function of the sentences. In this section, we outline an approach on the teaching of text production concepts in *EasyTalk* by wrapping an active control structure, called *EasyText*, around the key components for typing. We previously published this concept in Steinmetz and Harbusch (2021b) and Harbusch and Steinmetz (2022).

Prolific writers know that writing coherent, understandable text emphasizes audience design concepts. A writer’s workshop aims at teaching students the process of text writing through practical application (cf. Graves and Murray (1980) for the history; Hicks (2009) for the digital application of the concept). When *EasyText* is active, the system functions as a teacher taking over the initiative by asking questions at different stages of the text production. For convenience, this mode can be easily ended or reactivated at any point in time.

At the beginning of a text, the user has to answer questions from a checklist (cf. Figure 5.17 for an excerpt; the questionnaire presented to the user can be adapted to specific text genres). Depending on the user’s reading fluency, the questions can be read aloud to them (e.g., by a caregiver) or the read-aloud function of *EasyTalk* can be used to speed up the dialogue. Each checklist of *EasyText* starts with an

Initial story writing checklist		
1	Lieber David!	Dear David!
2	Denk dir eine Geschichte aus!	Think up a story!
3	Bist Du fertig?	Are you ready?
4	Beantworte zuerst die wichtigen Fragen von	First answer important question with respect
5	der Geschichte!	to the story.
6	Weil:	Because:
7	Die Leser*innen wissen viele Dinge nicht.	The readers does not know about many things.
8	Deshalb:	Therefore:
9	Wir schreiben die wichtigen Dinge!	We write the important things.
10	Wer soll deine Geschichte lesen?	Who is the reader of your story?
11	Mache ein Kreuz bei den Passenden:	Mark the fitting ones:
12	• Mama/Papa	• Mum/Dad
13	• Frank, mein Betreuer	• Frank, my caregiver
14	• Peter, mein bester Freund	• Peter, my best friend
15	Wer fehlt in der Liste?	Who is missing in the list?
16	Wir müssen jetzt herausfinden:	We have to find out now:
17	Was wissen die Leser*innen schon?	What do the readers already know?
18	• Wann findet die Geschichte statt?	• When does the story take place?
19	• Wo findet die Geschichte statt?	• Where does the story take place?
20	• Wer spielt mit?	• Who takes part?

Figure 5.17: An excerpt from the checklist that can be adapted to specific text genres and situations presented in the beginning of a writing workshop session in *EasyText*. (For details, see the text; to aid readability, the formatting in *EasyTalk* is omitted here.)

introductory text (cf. Lines 1–9 in Figure 5.17). Lines 10–15 collect background information on the reader in an intuitive manner. As far as possible, the individual questions of the checklist offer a range of alternatives to select from. Where this is not possible, the user types the answer using *EasyTalk*. Based on the currently active user profile (e.g., containing the name of the user (in our case *David*), and names of the caregivers, friends, etc.) and the user’s previously written texts, the system offers predictions. Lines 16–20 show part of the collection of background information for the text the user would like to write. A sequence of questions is asked to characterize all the protagonists in the list of actors (Line 20) so that the reader can identify them clearly. Different options are tested. Does the reader already know the name of the actor(s)? Can they be introduced by name? Can a characterization of the person(s) be added to enable the reader to become familiar with them (e.g., ‘*Peter is my best friend*’, ‘*Frank is my caregiver*’)? Such a session avoids the need for relative clauses (not allowed in (E)LS) to establish new protagonists in the story. Similarly, the background of every sentence is explored through questions referring to the modifier cues in Table 5.1.

When *EasyText* is active, the system asks the user to note down all changes or details unknown to the reader by asking explicit questions. Instead of simply displaying the modifier cues in *EasyTalk* (cf., e.g., Figure 5.7), *EasyText* stipulates the filling of modifiers (e.g., when and where the story takes place). *EasyText* provides default fillers in the selection list of the Next-Word Panel (e.g., time=“now”, place=“user’s home address”). For every new sentence, the system asks whether the current fillers have to be changed. Only in the beginning of the overall story,

and in case of a change, the fillers are added to the text.

This process has various benefits. Not only is the user trained in adding relevant audience-design aspects, but, in addition, the system can actively support the user during sentence production throughout the story. For instance, suggestions of personal pronouns can be made by the system when referring to protagonists during sentence production (e.g., *sie* ‘she’ for Mama). Assuming David is going to write the story we sketched in our ongoing example writing session, the system would stipulate the introduction of the actors (‘Peter is my best friend.’, ‘We’), the time (‘today’) and place (‘in the city’) of the event (‘to meet’) are added to the text – resulting in: “Peter ist mein bester Freund.//Wir wollten uns heute in der Stadt treffen.”//... ‘Peter is my best friend. We wanted to meet in the city today.//...’ This leads the user to provide the reader with necessary context. As described, the temporal and spatial modifiers are prefilled with the initial/most recent filler, ruling out wrong assumptions by the reader. In Figure our example, the second sentence, refers to *heute* ‘today’. However, without active user intervention, the sentence would display the When-cue filled with *yesterday*, as the user wrote the verb in preterite past. This leads the user to notice the clash with their intended content; in the example, the cue is revised to *heute* ‘today’. The question of whether or not the system should actively provide tense suggestions remains open. We hesitate to make our system overly adaptive. Many users – irrespective of their specific user group – do not appreciate non-static user interfaces (Lee and Yoon, 2004).

5.6 Additional features and system settings

This section presents the text functions (Section 5.6.1) and options for customizing *EasyTalk* (Section 5.6.2).

5.6.1 Text functions

The current prototype of *EasyTalk* offers basic text functions to the users. During a session, the user can activate the read-aloud functionality by clicking on a sentence in the Text Panel or a text element of the user interface. Finally, the users can export their finished texts from *EasyTalk* with or without symbols via the option ‘save text’ from the meta-level Menu Panel (cf. the gray arrow in Figure 5.18). In addition, the Menu Panel offers various settings providing further customization options (for instance, extending the vocabulary or changing the symbols) which we discuss in the following section.

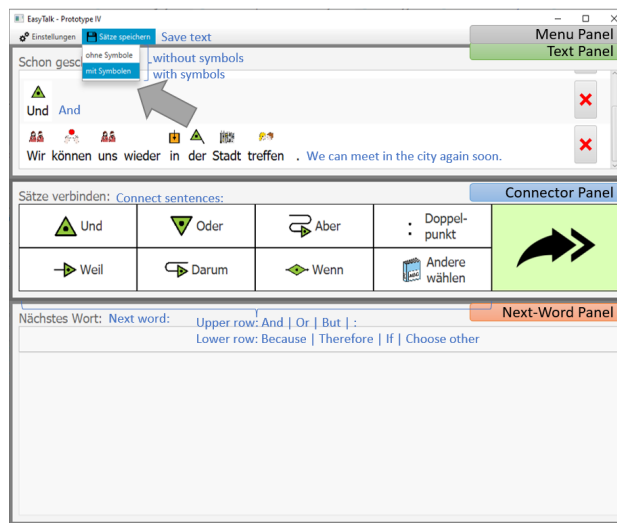


Figure 5.18: The text functions of *EasyTalk*. The written text can be exported from the system with or without symbols via the Menu Panel.

5.6.2 Options for system customization

EasyTalk offers the user several options for system customization, accessed via the Menu Panel (cf. Figure 5.18). In the following we detail on these options which are summarized in the Table 5.4.

As described in Section 5.4.2.6, *EasyTalk* offers two writing modes. The user can choose between the *guided writing mode* and the *free writing mode*. In the settings menu, both options are presented in form of a radio button group. By default, the guided mode is active.

To customize the vocabulary, *EasyTalk* offers two ways. As mentioned, CELEX is the basis of the *EasyTalk* vocabulary. CELEX comprises over 51.000 base forms (for verbs, this is the infinitive form, and for nouns, the nominative singular form) and more than 365.000 inflected word forms. To meet the requirements of the ELS, we have removed some word forms from the lexicon – such as substantive forms that occur only in the genitive case and verb forms in the subjunctive (except for the subjunctive forms for modal verbs and auxiliary verbs). In the system settings, the user may choose between this ‘full’ version of the lexicon and one that is restricted to CEFR L2-levels A2-B1. This option is also offered in the form of a radio button group. To restrict the lexicon, we use a lemma list loading the relevant lemmas by their database id at system startup. In the prototype, *EasyTalk* does not offer a graphical user interface to browse the lexicon and modify this list, i.e., to add or remove word forms. However, this feature is certainly needed in a mature version of the system to provide non-specialists with a simple and fast tool to create and customize personal vocabularies.

Table 5.4: Options for system customization. Left column: category. Right column: customization options.

Category	Customization Options
Writing modes	- Guided writing mode (default) - Free writing mode
Vocabulary	- Select lemmas from CELEX - Add proper names
Symbols	- Exchange symbols - Turn symbols off/on - Show singular/plural markers for nouns
Read-aloud functionality	- Adjust voice - Read out all text elements on demand
Accessibility	- Play alert sound if the suggestion list is empty
System platform	- Windows OS

To add proper names to the lexicon, *EasyTalk* offers a simple three-step dialog, shown in Figure 5.19. First, the user or their helper enters the proper name into a text-input field. In the second step, the grammatical gender is selected in a radio button group. Optionally, in the third step, a symbol associated with the proper name can be loaded into the system via a standard open file dialog.

In general, all symbols in *EasyTalk* are easily exchangeable: the symbols are located in the form of a ‘.png’ or ‘.jpg’ file in the ‘symbols’-folder included in the *EasyTalk* system directory. Each file name corresponds to a lemma. For inflected word forms, the *EasyTalk* displays the symbol associated with the base word. This simple way of linking symbols and words makes it possible to exchange the symbols individually or as a whole set easily. To turn the symbols off, the user can either remove all symbols from the ‘symbols’-folder or deactivate the use of symbols in the *EasyTalk* system settings via the checkbox labeled *Symbole anzeigen* ‘show symbols’. In the German language, some noun forms can have ambiguous numbers. To visually highlight the number of a noun form, the user can display the number of nouns next to each symbol-noun combination in the Next-Word Panel by activating indicators in the system settings. The indicator for the singular is a ‘1’, and the one for the plural is a ‘+’. Both indicators, ‘1/+’, are shown for noun forms matching both numbers.

EasyTalk offers a basic read-aloud functionality. It can be used for the Sentence Panel only or be activated for all text elements. The read-aloud functionality uses the text-to-speech module *MaryTTS*. The current prototype of *EasyTalk* offers only the male default voice for German. However, different male and female

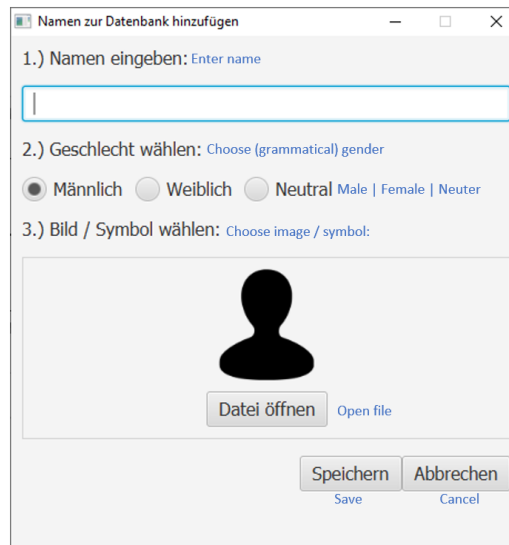


Figure 5.19: The dialog for adding proper names to the vocabulary in *EasyTalk*.

German voices are freely available on the internet⁸.

To notify users with that the suggestion list is empty (e.g., as a result of typing mistakes), the system can play an alert sound. This feature may be particularly useful for users with low computer skills who focus on the keyboard while typing and do not realize their typing mistakes until they look back up to the screen. Playing this alert sound can be activated via a checkbox.

To this point, *EasyTalk* was compiled and run on the Microsoft Windows operating system (version 7, 8, and 10). As the system is programmed in Java, it can be compiled for other operating systems as well.

⁸See, e.g., a public repository offering a female voice with Bavarian dialect: github.com/marytts/voice-bits1-hsmm

Chapter 6

System evaluation

This chapter is dedicated to the evaluation of *EasyTalk*. It reports the results of four test iterations in their chronological order. Each iteration corresponds to one version of the *EasyTalk* prototype (introduced in Section 4.5, Figure 4.2) and is depicted in a dedicated section, which provides information about the test setup, the test procedure, the participants, the insights gained, and the resulting modifications to the tested version of the *EasyTalk* prototype. First, Section 6.1 presents the system evaluation via interviews conducted with domain experts of AAC, CCN and LS. Section 6.2 presents the exploratory writing tests with beginner German learners with low computer skills. Section 6.3 is dedicated to the guided usability tests with IT-expert L2 learners of German. Section 6.4 presents the case study with the target group exploring free text writing with *EasyTalk*. Finally, Section 6.5 summarizes all evaluation results and the resulting changes to the system to illustrate the evolution of *EasyTalk* during the iterative HCD process – finally resulting in the version of the system presented in this work.

6.1 System evaluation in interviews with experts on accessible learning and barrier-free communication

In this section, we discuss the evaluation of the first version of the *EasyTalk* prototype. The group of LS readers is very heterogeneous (cf. Maaß and Bredel (2016, p. 139), see Table 4.1 for personas of prototypical users). Therefore, we talked to experts in the fields of CCN, AAC, and LS who are familiar with the diverse needs, abilities and conditions of our target users. The tests focused on obtaining feedback from various perspectives on whether the system is a suitable writing aid for the target group. At the same time, the experts represent possible *EasyTalk* stake-

holders as potential helpers of our users and readers of their texts. Partially, these test results have previously been published in Steinmetz and Harbusch (2020) and Harbusch and Steinmetz (2022).

6.1.1 Test setup and participants

To evaluate the first prototype of *EasyTalk*, we consulted experts in the fields of CCN, AAC, and LS who are familiar with the diverse needs, abilities and conditions of our user group. The expert group, offering different perspectives, consists of:

- **A male LS reader** with learning disabilities. He regularly reads LS texts, for example the weekly LS-newspaper provided by the facility for assisted living he lives and works at. As a member of an LS review group, he regularly reviews LS texts for their ease of comprehension, and does some writing himself. Correspondingly, he can be qualified as an (advanced) real user of our target group.
- **A male LS writer** leading an LS group for which he regularly writes LS texts and tests them for comprehensibility with a group of people with intellectual disabilities and learning difficulties. In addition, he hosts an inclusive reading club for a foundation for people with disability and has many years of professional experience as a social worker in the field of assisted living for adults with intellectual or multiple disabilities.
- **A female AAC expert.** She now heads an AAC team in an institution for people with multiple and/or severe cognitive or sensory disabilities after several years of professional experience in the field of AACAC. She knows a great number of AAC solutions suitable for clients with different needs and application scenarios, including literacy training for AAC users.
- **A male domain expert** working with people having severe cognitive and physical disabilities. He is a recently retired social worker offering us valuable expertise during a career of more than 35 years. He is familiar with a broad variety of communication methods and witnessed the field evolve – though it has never been part of his job to implement or select a communication method for a client.

For our expert group, we used the case study method¹, conducting exploratory think-aloud probes followed by semi-structured interviews. The tests were performed under normal room lighting on a laptop with 15” display screen resolution

¹See Lazar et al. (2017) for a good overview of the benefits and drawbacks of case studies.

6.1. SYSTEM EVALUATION IN INTERVIEWS WITH EXPERT GROUPS 117

of 1920x1080. *EasyTalk* was set up using the ARASAAC symbols. The test participants operated *EasyTalk* on the provided laptop keyboard with an external mouse or the inbuilt touch pad of the laptop. The test sessions were conducted within the time frame of one hour. During the test session of the LS writer, a personal caretaker was present in addition to the interviewer.

6.1.2 Test procedure

At the beginning of each session, the participants received a short introduction to the system in form of a demo sentence written by the interviewer. Then, as first probe, each participant should write the following sentences of varying complexity:

- *Er ist alt.* ‘He is old.’
- *Ich gebe dir das Buch.* ‘I give you the book.’
- *Ich will ein Eis essen.* ‘I want to eat an ice cream.’
- *Die schöne Frau gibt dem netten Mann das Buch.* ‘The beautiful woman gives the nice man the book.’
- *Die Katze von der Frau schläft.* ‘The cat of the woman sleeps.’

As second probe, the participants should write sentences themselves. In both probes, we tried to elicit the think-aloud method. Moreover, we protocolled the answers to the following open-end questions:

- **RQ1:** What was easy and what was difficult during typing?
- **RQ2:** How do you evaluate the AAC-support features?
- **RQ3:** Do you think the system is useful? Please elaborate on your choice!
- **RQ4:** Would you like to use the system? Please elaborate on your choice!

The transcript of all sessions notes can be found in Appendix A.1.

6.1.3 Test results

In the following, we present important insights from the evaluation sessions with the expert group on accessible learning and barrier-free communication. All experts acknowledged that *EasyTalk* meets the requirements of those users who know the alphabetic characters and have basic spelling skills, but have difficulties writing whole words or complex sentences and coherent texts.

The user interface was generally considered easy to understand. All experts gave positive feedback on the AAC symbols. However, they all asked whether the symbols are customizable, so that the users are not dependent on a specific symbol set but can use the symbols they are familiar with. The LS writer added, that the symbols should be easy to hide for advanced users.

The AAC expert and the domain expert were particularly focused on the accessibility of *EasyTalk*: Both asked how the program can be maintained. They appreciated that each user can use their own (possibly individually adapted) keyboard. They suggested to add the possibility for operating the system by *scanning*². Moreover, the expert group appreciated the read-aloud function for the produced text in *EasyTalk* (activated by clicking on a sentence) and suggested that all elements displayed in the panels should be supplemented with this function, thereby increasing barrier-free accessibility. To take the needs of users with visual impairments into account, two experts noted that the font size should be individually adjustable.

The experts emphasized the importance of being able to add one's own vocabulary – quoting the AAC expert: “People need to be able to describe their world in their own words.” They requested that individual vocabularies should be easy to prepare by the caregivers or teachers of the individual users.

The concept of breaking the process of sentence writing down into answering a sequence of wh-questions was accepted by all experts. They related the concept to parent-child dialogues (cf. Brandt et al. (2016)). Inspired by a practice used in school, the LS reader contemplated color-coding of wordforms according to their part-of-speech to provide additional support.

In terms of the word-selection strategy, the experts liked the filterable word list. They appreciated that it reduces the need for typing and offers correctly inflected wordforms. The experts forecast that the suggestion list provides the users with a feeling of security and speeds up typing at the same time.

Regarding the mechanism for using complement-taking verbs, we received positive feedback from our LS reader and the AAC expert. They described it as a “reasonable way” to access these constructions. However, the experts recommended thorough testing with varying groups LS users.

²Information on different scanning techniques: praacticalaac.org/tag/scanning/

6.1. SYSTEM EVALUATION IN INTERVIEWS WITH EXPERT GROUPS 119

The RST-related aspects of the system were recognized by the experts as a good way to practice connecting sentences. The LS reader expressed a particularly positive opinion: “It is a good way to link sentences without having to think too much”. However, the LS reader and the LS writer both stated, that the ARASAAC symbols for conjunctions did not appear intuitive or understandable for them. Again, they stressed the importance of setting the system up with symbols that are familiar to the individual user to avoid confusion.

The AAC expert expressed thoughts on formulating questions in *EasyTalk* – a function that had not yet been implemented at that time. She suggested that – instead of adding the punctuation mark at the end of the sentence – the punctuation mark could be chosen at the beginning of the sentence to communicate the sentence type and the resulting word order more explicitly. (We use this pattern in the *EasyTalk* version presented in this work.)

We were pleased with the largely positive feedback although, of course, we are aware that the participants knew they were talking to the developer. Moreover, one should not overgeneralize case study results. In general, the case studies revealed substantial interest and enthusiasms in the participants while they were executing the *EasyTalk* probes. We noted many hints to usage scenarios for a product version of *EasyTalk*: Private use (for example to compose emails), collaborative writing of LS text, collaborative writing of weekly logs of residents of facilities for assisted living, teaching in schools and German classes.

6.1.4 Changes to *EasyTalk* based on the test results

Based on feedback from the expert group, we further developed the *EasyTalk* prototype as follows: Expanding the linguistic scope of this early *EasyTalk* prototype, we implemented the support for question writing according to the AAC expert’s design suggestion. Based on the feedback of the expert group, we added the option to expand the read-aloud function to all text elements in *EasyTalk* in the system settings. Additionally, we added an easy-to-use dialog to add personal terms to the dictionary of *EasyTalk* and added a checkbox to en- or disable the use of AAC symbols in the system settings (cf. Section 5.6.2). We took note of the need for personal user profiles to configure the system for individual users for a product version of *EasyTalk*.

6.2 Testing with L2 learners with low German and low computer skills

In this section, we present the evaluation results obtained in tests with beginner L2 learners with low computer skills. The focus of these tests was to evaluate the process of free sentence writing in *EasyTalk* with users with low literacy and computer skills. Partially, these test results have previously been published in Steinmetz and Harbusch (2020) and Harbusch and Steinmetz (2022).

6.2.1 Test setup and participants

Table 6.1: Data on the L2 learners with low computer skills participating in the test session.

Participant	Age	Gender	Mother tongue	L2-German level	Computer skills
P1	28	male	Amharic	A1-A2	Low
P2	37	male	Cotocoli / French	A1	Low
P3	24	male	Tigrinya	A2	Rudimentary

Testing with L2 learners with low computer and low literacy skills in German, we chose the case study method as an appropriate way to gain insights on the usability of the system in free text-writing scenarios. We tested the system with three male L2 learners (cf. Table 6.1) with predominantly oral German language skills at CEFR-L2-learner level A1-A2. They are literate in their native languages (Amharic, Tigrinya, and French/Cotocoli). Their computer skills are rudimentary. They write German only in very short messages, e.g., to make appointments via messenger apps.

Each test session was conducted within the time frame of 30-40 minutes. The tests were performed under normal room lighting on a desktop computer with a 17" monitor and screen resolution of 1920x1080. *EasyTalk* was setup displaying the ARASAAC symbols. All participants used the provided keyboard and computer mouse.

6.2.2 Test procedure

At the beginning of each test session, the participant was given a short introduction to the system. The interviewer wrote one demo sentence with *EasyTalk*. Then, the participants were asked to write their own sentences. Aware that this was a test with novice German learners, the interviewer offered assistance with word finding or sentence wording. We aimed to create a casual situation in our experimental

6.2. TESTING WITH BEGINNER L2 LEARNERS WITH LOW COMPUTER SKILLS 121

<p>S1 Ich habe Zeit. 'I have time.'</p> <p>S2 Ich kaufe das Buch. 'I buy the book.'</p> <p>S3 Ich will das Buch lesen. 'I want to read the book.'</p> <p>S4 Ich mache die Wäschen. 'I do the laundries.'</p> <p>S5 Ich habe gestern gegessen. 'I have eaten yesterday.'</p>	<p>S1 Du antwortest der Frau. 'You answer the woman.'</p> <p>S2 Ich gebe dir das. 'I give you that.'</p> <p>C1 Und 'And'</p> <p>S3 Wir haben das Fahrrad. 'We have the bicycle.'</p> <p>C2 Oder 'Or'</p> <p>S4 Das Auto fährt. 'The car drives.'</p> <p>C3 Darum 'Therefore'</p> <p>S5 Er Arbeitet die Arbeit. 'He works the work.'</p>	<p>S1 Ich antworte dir. 'I answer you.'</p> <p>S2 Ich gebe dir das Buch abends. 'I give you the book in the evening.'</p> <p>S3 Das Angebot ist gut. 'The offer is good.'</p> <p>S4 Ich habe gestern geschlafen. 'I have slept yesterday.'</p> <p>S5 Du achtest deine Mutter. 'You respect your mother.'</p>
---	---	--

Figure 6.1: The texts written in the free writing sessions of the low-literate L2 learners – P1, P2 and P3 – with low computer skills. The participants were asked to write whatever comes to their mind, without being asked to connect their sentences.

set-up to avoid participants feeling that their German language skills are put to the test.

We conducted semi-structured interviews covering the following aspects:

1. How intuitive do Wh-cues guide the sentence construction process to people with little (written) German skills? Do users feel supported in increasing their grammatical skills in German?
2. Does the interface support people with low computer skills, especially those that are unfamiliar with common control patterns?

As the implementer conducted the interviews, we were able to use situational follow-up questions to evaluate possible workarounds when deficiencies were discovered. The full sessions notes are provided in Appendix A.2.

6.2.3 Test results

Here we summarize interesting findings of testing with novice German writers with little computer skills that have led to revisions in the interface.

Figure 6.1 shows the sentences written by the participants during their test sessions. Without over-generalizing, we observe that beginner-level users do neither recognize nor use the full extent of linguistic scope and support of *EasyTalk*. (This led to the design of an active teaching strategy in the writing workshop presented in Chapter 5.5.) At the same time, it showed that users can write according to their personal preferences and skill levels. All participants wrote at least five sentences. Except for one sentence by P2, all sentences were started with personal pronouns. The most complex sentence components consisted of a two-word combination of a noun and its corresponding article. Participants P1 and P3 used adverbs of time

as When-modifiers in their sentences (*abends* ‘at night’, *gestern* ‘yesterday’). P2 chose three connectors in the Connector Panel. (We note that we just asked the participants to try writing sentences using *EasyTalk*. We did not ask the participants to write connected texts.) We asked P2 for his opinion on the Connector Panel. He answered that he felt prompted by the menu to connect his sentences.

P1 wrote a complex verb construction (*Ich will das Buch lesen* ‘I want to read the book.’), operating the decision dialog without asking for help. To get additional feedback on the mechanism for writing complex verb constructions from the other participants, the interviewer demonstrated this feature to P2 and P3 after finishing their free writing session using the same sentence written by P1. The support while entering sentences with complement-taking verbs was highly appreciated. (*EasyTalk* automatically moves the infinitive to clause-final position in German – a different position than in our participants’ mother tongues).

In the prototype version tested, *EasyTalk* automatically adjusts the tense of a sentence according to the chosen When-modifier. For instance, it puts the sentence into the present perfect, the past tense used in LS, when the user selects an adverb of time describing past tense as When-modifier, e.g., *gestern* ‘yesterday’. P1 and P2 used this feature without commenting on it, whereupon the interviewer asked for their feedback on this feature. Both participants stated that they had not noticed their sentences’ automatic tense transformation but appreciated the system’s automatic correction in hindsight. We are, however, aware that unnoticed automatic corrections by the system may lead to unintended meanings in the resulting text. Therefore, the writing process should be changed to become more explicit.

All test subjects liked the support for correctly conjugating the verbs and choosing the correctly inflected wordforms and determiners.

During all tests, the same barriers to selecting word forms from the completion list hindered fast typing: Spelling errors (e.g., spelling *Zeit* ‘time’ as the similar-sounding ‘Seit’) or mistakes in selecting the gender of an article (*der/die/-das_{inflected}* ‘the’) result in an unexpected – possibly empty – completion list. Presumably, due to their low computer skills, our users tended to either focus on the screen or the keyboard. As a result, they continued to type even though they had already made a mistake that resulted in an empty word list. They first noticed the mistake when they looked at the monitor again, which had two effects: First, it slowed down the typing process because the participants had to correct their input before being able to select a word form. Second, the participants could not use the suggestions in the word list to help them spell or find words in the anticipated manner.

Sentence S4 by P1, *Ich mache die Wäschen* ‘I do the laundries.’, shows a weakness of *EasyTalk* for L2 learners: Although the sentence is grammatically correct, the phrasing is uncommon because the word *Wäsche* ‘laundry’ is usually

used in its singular form in German in this context. For word forms of the grammatically female gender, the article *die* is identical for direct objects in singular or plural. Subtleties like this may easily go unnoticed by language learners.

Finally, we received positive feedback regarding the combination of words and visual symbols. Participants emphasized that it helped them recognize and remember words more easily. The users actively resorted to the read-aloud function without being asked as part of the probes. All participants perceived the system as easy to use. This feedback is reflected by the fact that the participants operated ET autonomously after 1-2 demo sentences by the interviewer.

6.2.4 Changes to EasyTalk based on the test results

To create a more conscious and uniform writing process, we applied the mechanism for complex verb clusters also for combinations consisting of an auxiliary verb and a participle (e.g., *Ich habe gegessen* ‘I have eaten’; for details, see Section 5.4.2.3).

In the free-writing sessions, the L2 learners made significantly more use of the delete button than the participants of the expert group in the guided writing sessions with pre-given sentences presented in Section 6.1. In this context, we noticed the need to change the behavior of the delete button: Instead of deleting an entire sentence component, we changed its behavior only to delete the last word added to the Sentence Panel.

Due to our observation that typing mistakes causing the completion list to be empty may go unnoticed by users focusing on the keyboard, we added an acoustic signal to direct the user’s attention back to the text input field. The signal can be easily activated and deactivated in the system settings.

To support L2 learners in recognizing word forms with ambiguous numerus, the current version of *EasyTalk* offers displaying indicators for singular and plural noun forms: In the word list, a ‘1’ for singular and a ‘+’ for plural is displayed next to each noun. For ambiguous forms, ‘1/+’ is displayed. This indicator can be selected and deselected in the system settings.

During the test sessions, we identified the need to improve the focus function of the interactive elements in *EasyTalk*. The focused element should always be the one the user (probably) will use next – for example, when starting a new sentence, the text input window for choosing a word form needs to be in focus. The improved element focus saves clicks and enhances user guidance throughout the writing process.

6.3 Testing with IT-expert L2 learners

In this section, we present the evaluation results obtained in tests with L2 learners with expert computer skills. In these tests, we had IT-experts evaluate the usability of *EasyTalk* and focused in particular on the Connector Panel. Parts of these test results have previously been published in Harbusch and Steinmetz (2022).

6.3.1 Test setup and participants

Table 6.2: Data on the IT-expert L2 learners participating in the test session.

Participant	Age	Gender	Mother Tongue	L2-German level	Profession
P1	31-35	male	Romanian	B1	Software Developer
P2	25-30	male	Arabic	B1	Software Developer
P3	36-40	female	Spanish	A2	IT Solution Consultant
P4	41-45	male	Swedish	A2-B1	IT Business Consultant
P5	41-45	male	Swedish	A1-A2	IT Solution Architect
P6	46-50	male	Swedish	A2	Software Developer
P7	31-35	male	Swedish	A2-B1	IT Project Manager
P8	46-50	male	Swedish	A2	IT Solution Architect
P9	26-30	female	Mandarin	B2	CPQ Product Modeler
P10	31-35	male	Swedish	A1	IT Business Consultant

We conducted the tests with 10 IT experts of different native languages (Arabic, Romanian, Swedish, Mandarin, and Spanish) and German skills ranging from A1-B2 – see Table 6.2 for details. In these tests with expert-user L2 learners, we aimed to elicit clues about where to extend the linguistic support our target group might benefit from and to get detailed feedback on the usability of *EasyTalk* from expert computer users.

Due to ongoing COVID-19 restrictions and health precautions, all test sessions with the IT-expert L2 learners were conducted remotely: *EasyTalk* was running on the interviewer’s laptop with a screen resolution of 1920x1080 and operated by the participants via remote desktop control. As all participants are familiar with this technique from their working environment, this setup was not perceived as distracting. We recorded a screen video and session audio with the participants’ permission. All tests were conducted within the time frame of 25-30 minutes. The transcripts of all sessions are provided in Appendix A.3.

6.3.2 Test procedure

In the tests with IT-expert L2 learners, we apply discount testing (Nielsen, 1989). For obtaining comparable results, we presented each test participant with a picture

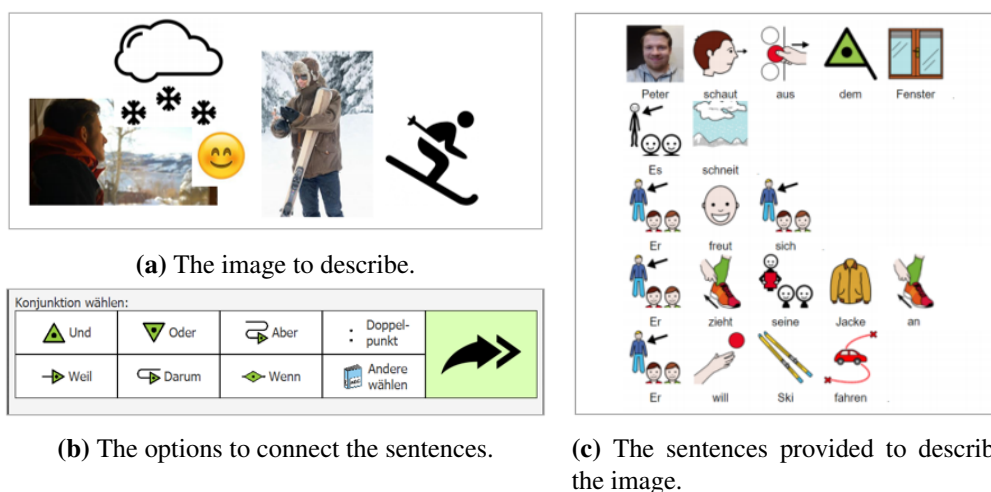


Figure 6.2: The test setup presented to the IT-expert L2 learners: The participants were asked to describe the image in (a) using the options of the *EasyTalk* Connector Panel shown in (b) to connect the sentences depicted in (c) providing the test subjects with German vocabulary augmented with AAC symbols.

supplemented with five sentences providing the German vocabulary (see Figure 6.2)³, allowing the test subjects to focus on typing rather than finding a German word. We asked the participants to perform thinking out loud while typing the story in *EasyTalk* – supposedly to email it to a child who only speaks German. At the end of each session, we conducted a short, semi-structured interview and asked the participants to fill out a user experience questionnaire, the *UEQ*⁴ (Laugwitz et al., 2006, 2008), in English for their convenience. We concluded each session by asking if the participant would recommend or use *EasyTalk* for learning German.

6.3.3 Test results

Testing with IT-expert L2 learners, we focused on the following questions: Which support do L2 learners with expert computer skills with little (written) German skills expect from an advanced writing system? How do expert users rate the usability of *EasyTalk*? In particular, we focused on identifying flaws in the Connector Panel.

The within-sentence support results were mainly positive; all participants felt supported by the system. Moreover, they reported that the system is easy to understand and intuitive to operate. This feedback is reflected in the fact that all learners

³The short story was written in *EasyTalk* and exported from the system with symbols.

⁴UEQ website: www.ueq-online.org

managed to use the system autonomously after a demo where the interviewer produced one example sentence with *EasyTalk* (i.e., the first sentence of the pre-given text displayed in Figure 6.2 (c) – *Peter schaut aus dem Fenster*: ‘Peter is looking out the window.’). Same as the previous L2 learner group (see Section 6.2), the IT people appreciated the support by the combination of symbols and words, the automatic inflection of words, and the word-ordering support. The support for verbs with separable verb prefix received particularly positive comments. Participants appreciated that the SVP is automatically placed in the correct position and that its association with the associated verb is visually highlighted by using the same symbol (see, e.g., sentence 4 in Figure 6.2 (c) – *He puts his jacket on* ‘He puts his jacket on.’). The decision dialog for complex verb clusters (Triggered in sentence 5, *He wants to go skiing* ‘He wants to go skiing.’) was operated by the participants without difficulties, although they were not demonstrated the mechanism beforehand. Four participants voiced that the system made them aware of the German word order.

In terms of challenges for L2 learners, we encountered similar issues to those found in the earlier testing sessions with non-native speakers: Several participants asked for active support in spelling or finding a German word. We need to keep this aspect in mind when testing *EasyTalk* with the target group. Our target group has spelling problems, too. In addition, not all users may be native German speakers. Moreover, some errors typically made by L2 learners may not be recognized by the system as *EasyTalk* does not check the semantics of the written texts. A question from participant P5 highlighted this: When writing the last sentence, S5 – *He wants to go skiing* ‘He wants to go skiing’, P5 asked if he could add a location or time instead of the object (ski). To try, he wrote the sentence again. This time, instead of ‘ski’, he applied the wh-filter for directional modifiers to the suggestion list and added the direction *nach Zoo* ‘to zoo’. While the preposition *nach* ‘to’ is correct for cities or countries (e.g., *nach München* ‘to Munich’), in the chosen context, it should be *in den Zoo* ‘into the zoo’. A similar typical L2 learner error went unnoticed by the system in the previous tests with low-literate German learners (see Section 6.2).

The suggestions for further linguistic support by the L2 learner expert users at the sentence level concerned, in particular, the presentation and selection of vocabulary: Three test participants proposed to add vocabulary suggestions based not only on the literal spelling but as well on the phonetic spelling (i.e., adding suggestions of similar sounding words like *Zeit* ‘time’ and *seit* ‘since’). They also suggested enabling entering words via the microphone. Other participants suggested a mode with lemmas on disposition from a pre-selected topic or domain similar to common AAC grid layouts (cf. the AAC systems presented in Section 3.1). We note that for some AAC users using a familiar grid display (see, e.g., Figures 3.3 or 3.4) may be an advantageous – possibly even necessary – option.

Table 6.3: The choices C1-C4 made by the participants in the Connector Panel to connect the sentences S1-S5 into a short story. In the case of “-” a participant opted to skip the choice of a connector. The first connector, C1, was selected by the interviewer as part of the system introduction. P7 and P10 opted to write individual texts. Their results are excluded from this table.

	Participant							
	P1	P2	P3	P4	P5	P6	P8	P9
S1	Peter schaut aus dem Fenster.							
C1	:	:	:	:	:	:	:	:
S2	Es schneit.							
C2	Und	Und	Darum	Und	Und	Und	Und	Und
S3	Er freut sich.							
C3	Darum	-	-	Und	-	-	-	Und
S4	Er zieht seine Jacke an.							
C4	Weil	Weil	Und	Weil	Weil	:	Und	-
S5	Er will Ski fahren.							

The sentence combining support in the form of the Connector Panel was basically judged intuitive and meaningful by all test subjects. Table 6.3 shows the choices made by the participants in the Connector Panel to connect the sentence into a short story. The results of this more free-form task show that each participant interpreted the sentence contexts slightly differently; no participants chose an identical set of connectors. However, all participants expected the system to provide feedback on the quality of their choice. This request probably originates from the fact that most test subjects indicated that they have experience with fill-the-gap German-language learning-software⁵. Of course, they appreciated the freedom that *EasyTalk* offers compared to these exercises. Nevertheless, help similar to the system’s word order expertise would be welcome here. This problem is difficult for *EasyTalk* as we have no overall content representation to check what is a reasonable/necessary relation to add.

In filling out the UEQ questionnaires, the participants rated *EasyTalk* on 26 items using opposing pairs of adjectives on a scale from negative (-3.0) to positive (+3.0) within a time frame of ~5 minutes. In doing so, they were invited to express their thoughts aloud. Table 6.4 depicts the accumulated results of the UEQ user experience questionnaires created with the *UEQ Data Analysis Tool*⁶ sorted by the categories *attractiveness*, *dependability*, *efficiency*, *novelty*, *perspicuity* and

⁵Several participants compared *EasyTalk* to the language learning app *DuoLingo* (www.duolingo.com).

⁶The *UEQ Data Analysis Tool* is available at www.ueq-online.org.

Table 6.4: The results of the UEQ user experience questionnaires, grouped by category. The first column shows the negative end of the scale, and the second one the positive end. -3 is the lowest, +3 the highest rating. The table shows the values *Mean*, *Variance*, *Standard Deviation (Std. Dev.)* and the number of participants, *No.*, who rated an item.

Negative	Positive	Mean	Variance	Std. Dev.	No.
Attractiveness					
annoying	enjoyable	1.7	1.8	1.3	10
bad	good	2.1	0.8	0.9	10
unlikable	pleasing	1.9	0.5	0.7	10
unpleasant	pleasant	1.9	0.8	0.9	10
unattractive	attractive	1.2	1.5	1.2	10
unfriendly	friendly	2.1	0.8	0.9	10
Dependability					
unpredictable	predictable	0.6	2.8	1.7	9
obstructive	supportive	1.8	1.7	1.3	10
not secure	secure	0.9	1.6	1.2	8
does not meet expectations	meets expectations	1.4	1.4	1.2	10
Efficiency					
slow	fast	x	x	x	0
inefficient	efficient	1.6	0.6	0.7	8
impractical	practical	1.9	0.8	0.9	10
cluttered	organized	1.6	0.9	1.0	10
Novelty					
dull	creative	1.4	2.5	1.6	10
conventional	inventive	1.7	1.0	1.0	9
usual	leading edge	0.8	1.7	1.3	10
conservative	innovative	1.4	0.9	1.0	10
Perspicuity					
not understandable	understandable	1.6	1.4	1.2	10
difficult to learn	easy to learn	2.2	0.4	0.6	10
complicated	easy	1.6	0.7	0.8	10
confusing	clear	1.5	1.6	1.3	10
Stimulation					
inferior	valuable	1.8	0.8	0.9	10
boring	exciting	1.4	2	1.4	10
not interesting	interesting	2.2	0.6	0.8	10
demotivating	motivating	2	0.4	0.7	10

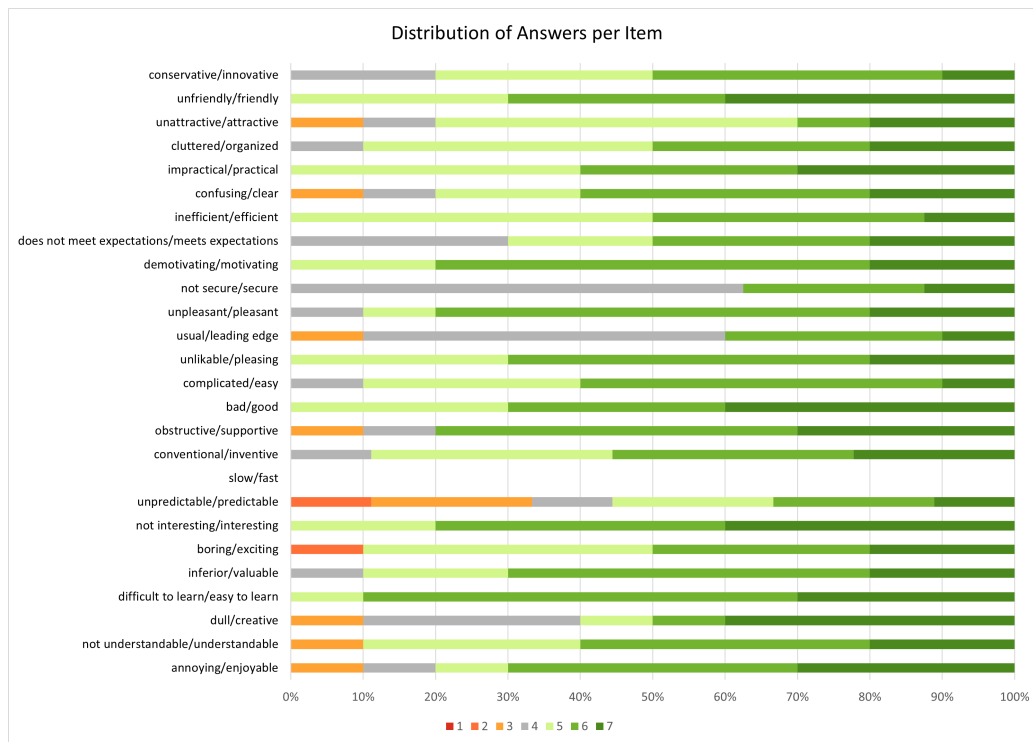


Figure 6.3: The UEQ distribution of answers to the single items in the order presented to the 10 participants. The scale shows the percentage of negative (from red to orange), neutral (gray) and positive (from light green to dark green) judgments. Overall the results are mainly positive with very few polarizing items.

stimulation. Figure 6.3 shows the distribution of answers to the single items in the order in which they were presented to participants. In the following analysis of the results, we focus on points that show a polarization in the answers.⁷

In the category *attractiveness*, the test group perceived *EasyTalk* as very good, very friendly, very pleasing, very pleasant, and very enjoyable. The item *attractiveness*, however, received the lowest rating with a mean value of 1.2/3.0 points. Participants expressed that they did not find the graphical user interface design particularly appealing. Foremost, the IT experts criticized its ‘outdated’ look and feel. Nevertheless, the distribution of answers in Figure 6.3 shows, that only one participant rated the system to be unattractive.

Within the category *dependability*, *EasyTalk* received mixed ratings. 8 of 10 testers found *EasyTalk* supportive. 7 participants confirmed that the system met

⁷We are aware that the expressiveness of the UEQ increases with a larger test group. In combination with the feedback from the usability tests, however, the questionnaire offers valuable insights even with 10 participants. Based on the *UEQ Data Analysis Tool* we assume an precision of 0.5 and an error probability of 0.05 for our number of participants.

their expectations; 3 participants voiced that they had no prior expectations and therefore chose a neutral rating (see Figure 6.3). The item *predictability* is the most polarizing one. It received the highest variance (2.8 points) and the lowest mean value (0.6 points). Figure 6.3 depicts that half of the test group rated *EasyTalk* to be predictable, while the other half perceived the system as rather unpredictable. As mentioned earlier, many participants appreciated that the system allows for creativity through free writing – which, of course, is not predictable. Therefore, the low predictability value is not necessarily negative. On the scale from *not secure* to *secure*, *EasyTalk* received a relatively low mean value (0.9 points). Figure 6.3 shows, that 60% of the ratings were neutral and 40% positive. Two participants omitted rating this item with the comment that looking at the system from an IT security perspective was not relevant. This allows for various interpretations. It is possible that other IT experts also viewed this item from an IT security perspective and/or were unable to place the relevance of the item in relation to *EasyTalk*. However, it could also reflect the feedback from participants that they would have liked more feedback on their choice of connectors. As we can only speculate, we make a note to ask specifically whether *EasyTalk* gives users a sense of security when writing in further tests.

In the category *efficiency*, *EasyTalk* was rated to be efficient, organized and very practical. The evaluation of the writing speed (*slow/fast*) was omitted due to the network delays caused by the remote setup.

In terms of *novelty*, the test group rated the system creative, innovative, and very inventive. On a scale from *usual* to *leading edge*, *EasyTalk* received a mean value of 0.8 points. Again, the ordinary UI design was mentioned. At the same time, participants commented that the system uses familiar operating patterns. We assume that this is one of the reasons for the high *perspicuity* of *EasyTalk*.

In the category *perspicuity*, *EasyTalk* was perceived as very easy to learn (mean: 2.2 points), easy (mean: 1.6 points), understandable (mean: 1.6 points) and clear (mean: 1.5 points). Some participants voiced, that the mechanism for writing complex word clusters, which had not been demonstrated beforehand, had made them halt and think for a moment. At the same time, the participants said that, in their opinion, the decision dialog is easy to use with a little practice.

EasyTalk also received consistently high scores in the category *stimulation*: the participants perceived the system as valuable (mean: 1.8), exciting (mean: 1.4), very interesting (mean: 2.2) and very motivating (mean: 2.0).

The positive feedback and the overall positive ratings in all categories of the UEQ questionnaire are reflected in the participants' positive responses to the final question: 8 of 10 testers would recommend or use the system for language learning. Six of them would prefer to run *EasyTalk* as a smartphone rather than a desktop app.

6.3.4 Changes to EasyTalk based on the test results

All IT-expert L2 learners perceived *EasyTalk* as a good system that is easy to learn, very motivating, practical, friendly, pleasing and very interesting. Due to the positive overall verdict, we made no significant changes to the system after testing with the IT-expert group. However, we have collected some suggestions for later versions of *EasyTalk*, such as:

- Developing a smartphone/tablet version of *EasyTalk*.
- Adding content representation to the system in order to be able to provide more profound feedback on the semantics of the written texts (e.g., to enable the system to identify unusual wording or to provide feedback on the choice of connectors).
- Providing different vocabulary representations to the Next-Word Panel, such as AAC symbol grids or word clusters based on a specific topic or domain.
- Making the suggestion list in the next word panel more robust against spelling errors, for example by displaying words that are literally or phonetically spelled similarly to the user input in the Next-Word Panel.
- Offering the possibility enter words via the microphone.

6.4 Testing with the target group

In this section, we present the test results of our final evaluation conducted with the target group. These results have partially been published in Steinmetz and Harbusch (2022).

6.4.1 Test setup and participants

Testing with people with disabilities presents unique challenges and increased organizational effort (cf. Lazar et al. (2017): Chapter 16, for an overview) – for example, during the test sessions, special precautions needed to be taken in direct contact with the target group which is particularly vulnerable to COVID-19 (cf. Rödler (2020); Portal et al. (2021)). Therefore, we conducted a qualitative case study aiming to uncover the biggest usability challenges for our target group in our software with only a handful of participants (cf. discount testing; Nielsen (1989)). For this purpose, we visited the *Schreibwerkstatt* ‘writing workshop’ of the *Habila Tannenhof Ulm*⁸, an institution for adults with intellectual or multiple disabilities

⁸Website of the *Habila Tannenhof Ulm*: www.habila.de/wohnen-assistenz/ulm/tannenhof-ulm

Table 6.5: Data on the participants (Genders: M = Male, F= Female; Conditions: ASD = Autism Spectrum Disorder, VI = Visual impairments, HoH = Hard of Hearing, CCN = Complex Communication Needs, MI = Motor impairments, IDD = intellectual or developmental disability). P8 opted out of the test on her own wish.

Participant	P1	P2	P3	P4	P5	P6	P7	P8
Age	20-25	20-25	18-20	20-25	20-25	20-25	20-25	18-20
Gender	M	M	F	F	M	M	F	F
Condition(s)	ASD	ASD, VI	HoH, CCN	IDD	IDD, VI	IDD	IDD, MI	IDD, VI
Uses spelling checker	N	Y	N	Y	Y	Y	N	N
Uses a mouse	N	Y	N	N	N	N	Y	N
Regular computer use	N	N	N	N	N	Y	N	Y
Eye tracking recorded	Y	Y	Y	Y	Y	N	N	Y

in southern Germany. Eight German-speaking participants, aged 18-25, with different conditions, writing and computer skills exploratively tested the system in free-writing sessions from 25 to 40 minutes. The heterogeneity of the group of testers is evident in Table 6.5. The tests were performed under normal room lighting on a laptop with 15” display screen resolution of 1920x1080. *EasyTalk* had to be operated in the same setup (e.g., displaying the ARASAAC symbols) by all participants using the inbuilt laptop keyboard and an external mouse.

6.4.2 Test procedure

Predefined tasks – like in a usability study – might exert pressure and frustration on the side of the target group – not directly related to flaws of the communication features to be evaluated. Thus, we aimed to create a casual situation in our experimental set-up to avoid participants feeling that their personal skills were being scrutinized. To provide a feeling of security, the individual caregiver (or the writing workshop leader) and only one person from the evaluation team (the interviewer) were present in each session. They started with a brief warm-up to break the ice.

In previous evaluations (cf. Sections 6.1, 6.2, and 6.3), we used standard evaluation techniques like cognitive walkthroughs, thinking aloud or user experience questionnaires. However, these techniques would overtax the target group. Besides logging the users’ actions, we used eye tracking as far as the participants gave their permission and conditions allowed for recording eye movements with a *Tobii Pro Nano*⁹ portable eye tracker to obtain objective information (cf. Bojko (2005)) and gather insights on viewing/interaction patterns, potential blind spots and stumbling points. If the participant agreed to the eye tracking, the interviewer

⁹www.tobiipro.com/product-listing/nano/

briefly explained and performed the calibration process once ¹⁰ with *Tobii Pro Lab*¹¹.

To explain how the system works, the interviewer wrote one sample sentence in *EasyTalk*: *Die Sonne scheint heute*. ‘The sun shines today.’ The participants could opt for rehearsing the example interactively with the interviewer. Afterwards, all participants were invited to freely explore the system. (Before the experiment, the leader of the *Schreibwerkstatt* had advised participants with spontaneous decision-making problems to think up in advance the sentences they wanted to write during the experiment.) If needed, the participant received help with spelling or interacting with the computer either from the interviewer or the caretaker. At the end of the typing session, the interviewer exported the text from *EasyTalk* with or without symbols according to the participants preference to hand it to them as receipt for participating in the experiment. One final yes/no-question was asked: Would you like to use *EasyTalk* in the writing workshop in the future?

The individual session notes including the written texts and the eye-tracking data can be found in Appendix A.4.

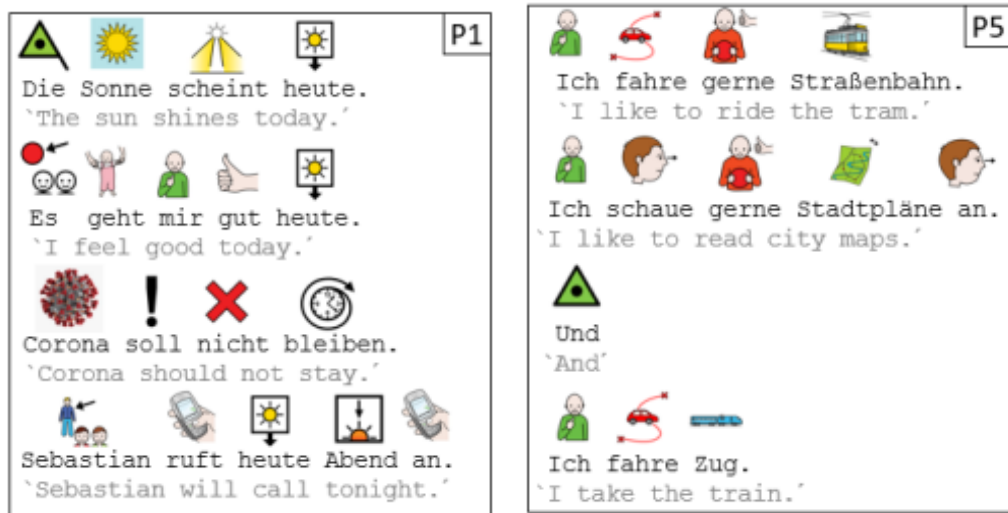
6.4.3 Results

In the following, we discuss the results of our case study with target users. In general, the evaluation corroborates the easy and intuitive interface design of *EasyTalk*. Only participants P6 and P8 are regular computer users; two participants stated that it was their first time using a laptop for text writing. Nevertheless, all participants successfully typed a series of at least three sentences with an average of four words with *EasyTalk* (see Figure 6.4 for the text typed in two sessions) – a remarkable result according to the leader of the writing workshop. Four participants spontaneously skipped the interactive example rehearsal and typed their own sentences without problems. Participant P8, who can write texts beyond the scope of LS in *MS Word*¹², stated that *EasyTalk* did not benefit her and opted out of the test after writing a four-word sentence. We exclude P8 from the following evaluation. Spontaneously, P5 judged: “*The headers help with concentration*” and “*The connectors between sentences are important. Sometimes there are longer sentences. You can do them piece by piece in this manner.*”. P2 stated:

¹⁰Several participants moved their heads or looked away from the monitor during the session. Nevertheless, we deliberately refrained from re-calibrating to avoid distractions and confusion. Compared to comparative eye tracking studies used, for example, for market analysis, the loss of data accuracy was acceptable because the analysis of viewing patterns from individual free writing sessions of the case study did not require exact metrics.

¹¹Information on the *Tobii Pro Lab* calibration process: connect.tobii.com/s/article/how-to-calibrate-and-validate-in-tobii-pro-lab

¹²*Microsoft Word*: www.microsoft.com/en/microsoft-365/word



(a) P1 chose to type the interviewer's example himself as first sentence. P1 skipped the choice of connectors for the entire text.

(b) P5 typed four sentences without rehearsing the interviewer's example and used an explicit connector once (*und* 'and').

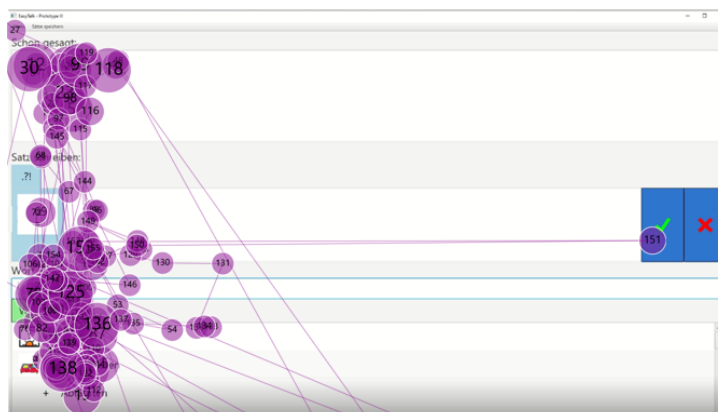
Figure 6.4: Two samples sessions of participants P1 and P5.

"It works great but I have to concentrate a bit here.". We attribute the overall positive result to improvements of the overall interface based on our previous three test sessions with different groups of substitute users described in Sections 6.1, 6.2 and 6.3. The tests in free-writing sessions with the target group confirm that the communication with the system is easy to learn due to intuitive dialogues all over the system.

The eye-tracking data supports this claim. We defined areas of interest (AOIs) (see Figure A.25 in the appendix for details) in the interface to be able to track task-accomplishment paths. In addition, we extracted times of interest (TOI) in form of the sentence-writing intervals and the sentence-connecting intervals from the session recordings for a more precise analysis. Gaze plots¹³ offer insights on viewing patterns by showing location, order, and time spent looking at locations on the stimulus. In combination with event logs of keystrokes and mouse clicks they reveal interaction strategies. Heat maps¹⁴ visualize dwell time and indicate the participants' visual attention and focus. All users focused on the dialogue ele-

¹³We use gaze plots exported from *Tobii Pro Lab*, v. 1.171. Each circle of a gaze plot marks the location of a fixation. The numbers show the order of fixations. The lines between the circles visualize the gaze trail. The circle sizes indicate the duration of the fixation – a large circle represents a long fixation, a small circle a short one.

¹⁴We use heat maps based on the absolute duration of fixations within a TOI exported from *Tobii Pro Lab*, v. 1.171. We use the standard color range from green to red – the warmer the color, the longer the fixation duration. Areas without fixations are rendered transparent.



(a) A gaze plot of a sentence-writing interval showing participant P3 looking at the sentences in the Text Panel.



(b) A heat map of a sentence-connecting interval showing participant P1 looking at the previous sentences.

Figure 6.5: Two samples of eye-tracking data showing participants looking at *EasyTalk*'s Text Panel during a text-writing interval (a) and a text-connecting interval (b). The eye-tracking data are layered over an empty screenshots of *EasyTalk* for visual orientation.

ments in the intended manner. With respect to effectiveness, we did not find traces of searching around for functionality. P2 and P5 who have visual impairments indicated, however, that a larger and/or highlighted cursor would be beneficial to them. (This is reflected in their eye-tracking data, which show signs that both participants were looking for the cursor.)

During the writing sessions, the read-aloud feature was positively mentioned and repeatedly used by our participants. The eye-tracking data documents the inspection of the Text Panel additionally. Figure 6.5 shows two examples: The gaze plot in Figure 6.5a reveals that P3 looked repeatedly at the finished sentences in the Text Panel while writing a new sentence. The heat map in Figure 6.5b

visualizes P1 looking at the text during a sentence-connection interval with a focus on the most recent sentence. The data support the claim that – for text-writing – it is beneficial to display more than the current sentence to users to provide context and memory support (cf. the AAC tools in Section 3.1 that commonly only show the current sentence).

According to the eye-tracking data, the participants focused mainly the Next-Word Panel while writing the current sentence. The Text Panel and the Sentence Panel were used to back up the flow of thoughts. Figure 6.6 shows three representative heat maps of sentence-writing intervals of different users.

All participants supplemented their sentences with modifiers (e.g., the cues *When* or *How* were spontaneously selected in the Next-Word Panel). As mentioned, P5 said that the wh-cues helped him concentrate. The eye-tracking data confirm that participants focused the wh-cues while writing. In detail, they exhibited different interaction strategies: The wh-cues were used to filter the word list for a certain component, to gain ideas on what more to add to a sentence, and to provide orientation and thought support in the current sentence. Figure 6.7 shows four example scenarios that indicate these different underlying uses: In (a), the participant presumably scanned the cues to overview the available sentence components before activating the *How*-cue to filter the word list. In (b), the participant viewed the already activated cue just before selecting the next word. In Figures (c) and (d) the cues were focused in free writing mode (i.e., selecting words from the suggestion list without filtering for a specific sentence component – cf. Section 5.4.2.6 before the next word was chosen. In both cases, the next word corresponded to the sentence component indicated by the focused cue (e.g., the cue *Wem* ‘Whom_{obj}’ was focused before the participant selected the word *Zug* ‘train’).

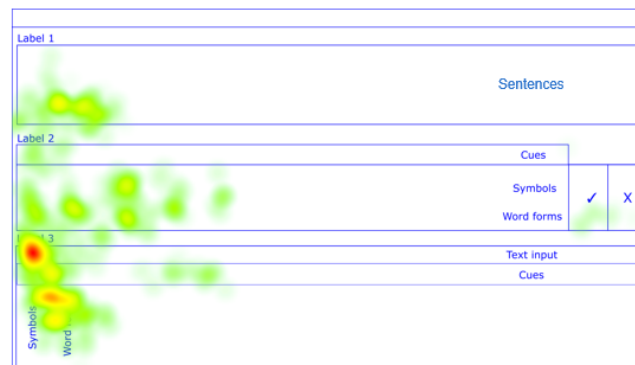
P3 and P7 spontaneously wrote a question. Both changed the sentence mode by clicking on the punctuation mark in the Sentence Panel without further instructions. Figure 6.8 shows how P3 started to write a question in three successive gaze plots.

The participants successfully performed the actions demonstrated in the system introduction – like clicking the green checkmark-button in the Sentence Panel to complete a sentence. Additionally, our observations reflect that *EasyTalk* is easy and intuitive to use for the target group beyond explicitly demonstrated features: Six participants completed the decision dialogue for complex verb constructions (cf. Section 5.4.2.3). Although we had not demonstrated this decision dialogue in the introduction, four participants typed sentences in present perfect tense, and two users selected a modal as finite verb followed by an infinitive (cf. the third example sentence in Figure 6.4a). Two participants spontaneously erased words in the Sentence Panel using the X-button – also not shown in the introduction.

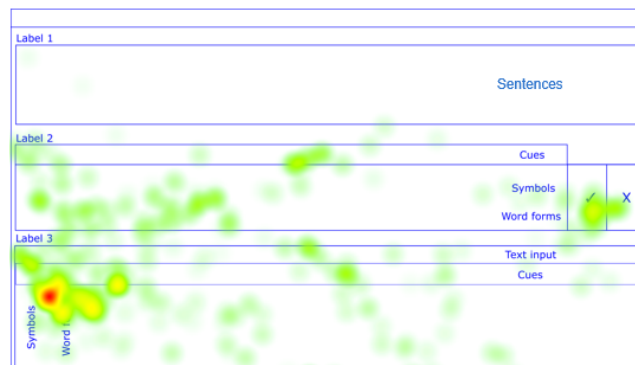
With respect to efficiency, P4 systematically selected the words as soon as they



(a) P3's 4th sentence-writing interval: P3 focused more on the text elements than on the symbols. P3 looked at the previous sentences in the Text Panel. The Next-Word Panel was the main focus.

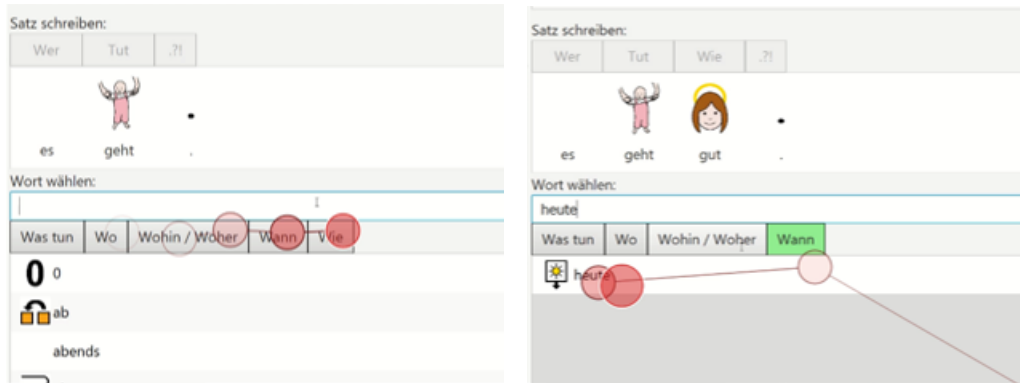


(b) P1's 4th sentence-writing interval: In the Text-Panel, P4 viewed the most recent sentence the longest. P4 focused text and symbol elements. The Next-Word Panel was the main focus.



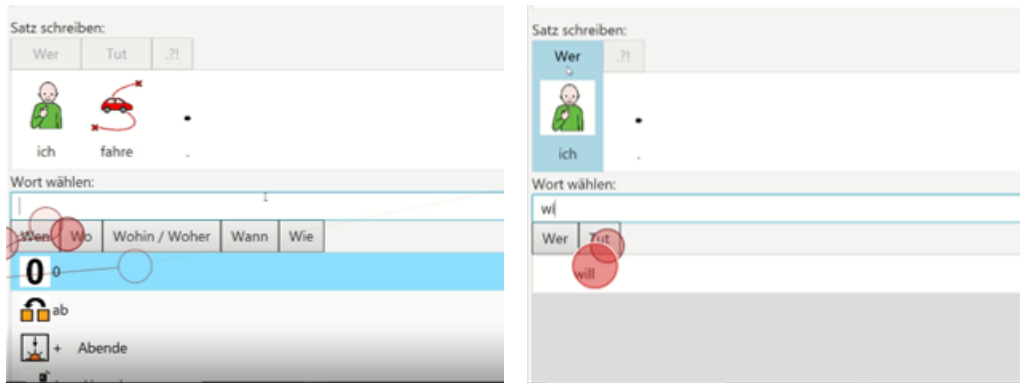
(c) P4's first sentence-writing interval: The empty Text-Panel shows no focus. In the Text Panel, wh-headers, symbols and text were looked at. The focus on the x-button confirms that P4 looked at the element to delete words. The main focus was on the Next-Word Panel.

Figure 6.6: Three heat maps of representative sentence-writing intervals of three different users. For orientation, the blue lines mark the *EasyTalk* interface elements.



(a) P1 read through the wh-cues from left to right before activating the cue *Wie* ‘how’ to filter the word list before he chose the modifier *gut* ‘good’

(b) P1 looked from the keyboard to the already active cue *Wann* ‘When’ to the time modifier *heute* ‘today’ in the suggestion list before selecting it.



(c) P5 focused the cue *Wen* ‘Whom_{obj}’ before adding the direct object *Zug* ‘train’ to his sentence. He did not activate the cue.

(d) P4 looked at the cue *Tut* ‘does’ before selecting the verb, although she did not use it to filter the suggestion list.

Figure 6.7: Four snapshots from the eye tracking recordings that show the gaze paths of different users focusing the wh-cues of the Text Panel in different contexts of use.

appeared in the completion list in favor of writing the words to the end. In contrast, P6 initially typed every word from start to finish. Later on, P6 selected the words from the completion list as soon as possible. P2 commented: “*Writing to the end is better.*” and judged the completion list as helpful to prevent spelling mistakes – a verdict we previously received in the tests with L2 learners. Nevertheless, we observed problems due to spelling or typing errors (e.g., accidentally entering a character several times, or misspelling a word). In return, the suggestion list remains unexpectedly empty. We already encountered this problem in the tests with beginner German-learners with low computer skills: If users look at the keyboard while typing, they first notice typing errors when they look at the screen. At

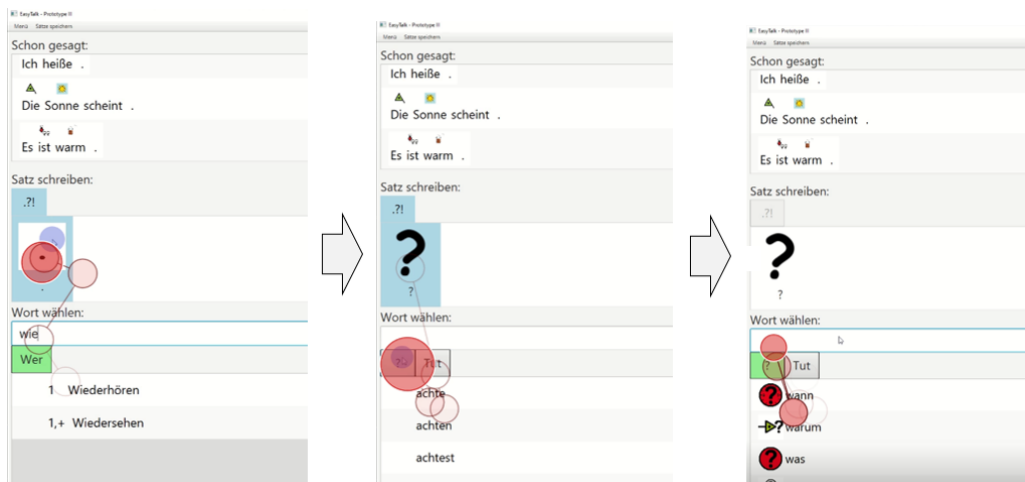


Figure 6.8: Three successive gaze plots showing how P3 started to write a question. First, P3 changed the sentence mode by clicking the punctuation mark. Then P3 selected the ?-cue to filter for wh-question words. Finally, P3 looked at the suggestion list and focused on the input-field before starting to type.

this point, however, the word list is already empty and does not indicate at what point the user has misspelled. Therefore, we plan to improve the word entering strategy in the next version of *EasyTalk* and provide improved spelling assistance.

The wh-cues displayed as headers at the top of the Sentence Panel indicate which sentence components have already been written and which words belong to which component. These headers, however, were not mentioned by any tester and were only looked at by one user according to our eye-tracking data. We suspect that this is due to their unobtrusive gray font. Changing their design would make the elements more visible.

To connect a sentence, all participants looked at the previous text in the Text Panel and read through the Connector Panel (see Figure 6.9 for an example gaze plot). Unfortunately, nobody felt inclined to add a connector systematically after reading through all/some options. The writing workshop leader pointed out that she was unsure whether the participants were proficient in sentence combining techniques – such as the use of connectors to create text coherence. Consequently, we intend to set up an active training mode in *EasyTalk* that teaches when and how to use text connectors (Reid et al., 2013). However, the eye-tracking data also unveiled shortcomings of the Connector Panel’s layout. Often, the second row of connector options was considerably less likely inspected. Moreover, the eye-tracking metrics of the AOIs revealed that the colon and the ‘choose other’-button were barely focused. In total, the colon received 5% and the ‘choose other’-button 3% of the participants’ dwell time when connecting sentences (see

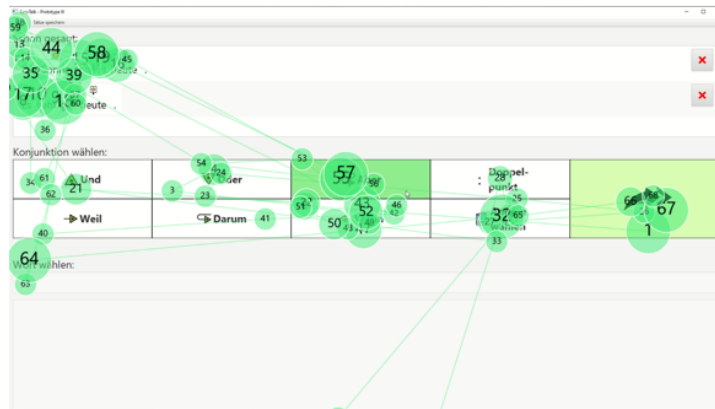


Figure 6.9: P1’s gaze plot while connecting sentences 2 and 3 using the Connector Panel. P1 looked at the previous text in the Text Panel and read through all connector options before operating the arrow button to skip the connector.

Appendix A.4.9 for the detailed numbers). Accordingly, we plan to offer the use of a shortened list of explicitly mentioned options in the system settings.

A potential stumbling point in the sentence-connecting interval is the display of the inactive Next-Wort Panel and its label (*Wort wählen* ‘Choose word’, see, e.g., Figure 6.9). All eye-tracking recordings show that participants focused on the label of this – at this point in time – functionless element (see, e.g., Figure fig:gazePlotP1Connecting). We plan to hide the element until it is activated by clicking the *Andere wählen/Choose other*-button. In any case, this particular element needs further investigation as it has not been used by any participant in our tests so far.

The participants preferred to use *EasyTalk* with the AAC symbols, although neither of them regularly communicates with AAC symbols – nor were they familiar with the ARASAAC symbol set. Based on the ratings from previous tests (cf. Sections 6.2 and 6.3), we suspect that the symbols give the system a friendly appeal. As previously expected by AAC experts (see Section 6.1), the ability to customize the symbols turned out to be important: Several participants expressed the wish to use the names of their loved ones associated with custom symbols (in this case, photos). Some AAC experts anticipated that the abstract symbols in the Connector Panel might distract or confuse users. Testing with target users, we could not confirm this concern. We plan to consider the effect of the connector symbols in a future in depth eye-tracking analysis of the Connector Panel.

For users with visual impairments, we found that *EasyTalk* needs to provide more interface customization options like a high contrast interface and a scalable mouse pointer. For users with motor impairments, *EasyTalk* is flexible enough as any adaptive input devices can be connected and *EasyTalk* can be used on touch

devices with on-screen keyboards.

While the vocabulary used in the tests was generally adequate, the participants expected to express themselves using familiar terms. If a particular word was not available, the writing process stalled, and the interviewer had to assist in finding a synonym (a rather complex process). The option to expand the dictionary on the fly would be beneficial – for example, to quickly add names of loved ones or special terms to the system without having to enter the system settings.

Finally, we address our observations on the scope of linguistic constructions. In the tests, the participants used *EasyTalk* with the linguistic scope of LS. However, we found that ELS (Harbusch and Steinmetz (2022), see Section 2.4) would be a better variant. ELS, for example, allows the use of simple past with modal verbs - a common construction, which test participants in this session also wanted to use. Two participants wanted to start their sentences with adverbs. (A common construction in German, for instance: *Jetzt frühstücke ich*. ‘Now, I will eat breakfast.’) For advanced users, the system could provide the flexibility to start the sentence with a different component than the subject. *EasyTalk* could offer this and similar features in the form of language options in the system settings to better suit the diverse language abilities and preferences of its target group.

Closing the session, all seven remaining participants expressed interest in using *EasyTalk* in their writing workshop in the future, positively indicating that *EasyTalk* is suitable for the target group and implies the joy of use. Because of the participants’ overall positive response, the leader of the writing workshop asked for a copy of *EasyTalk* for using it in future.

6.4.4 Changes to *EasyTalk* based on the test results

The positive results of the case study with our target group confirmed that the system supports our target users in writing and that *EasyTalk*’s interface and user guidance are simple and intuitive for users with low computer skills. When writing sentences, several users tried to use constructs beyond the scope of LS, such as using simple past for modal verbs. Therefore, we have expanded the linguistic scope of the system to ELS in the current version of *EasyTalk*, which supports these linguistic features. In addition, we discovered some points to further optimize the system for our target users in the future which we summarize in the following.

To provide more wealth of expression for advanced users, we plan to add more language constructs to be activated in the options menu, such as for example a more flexible sentence order that allows users to start a sentence with an adverb. In terms of accessibility, we want to provide options to increase the size of the font size and highlight the mouse pointer for users with visual impairments and plan to offer a high contrast mode. Since the wh-headers in the Sentence

Panel appeared to be a potential blind spot in the eye-tracking data, we plan to improve their design to make them more visually prominent to highlight more clearly which components of the sentence plan have already been filled. Moreover, we plan to add options to customize the word entering strategy in the next version of *EasyTalk* to provide more flexibility in terms of interaction patterns of different users (e.g., ignore trailing spaces, support selecting words using double click) and provide more profound spelling support. Moreover, we want to investigate whether the suggestions of the IT expert testers, such as displaying similarly spelled or similarly pronounced words as the user input in the suggestion list, are suitable for the target group or are more likely to confuse our users. Furthermore, we want to follow the suggestions of several IT-expert test participants who proposed to add vocabulary suggestions based not only on the spelling but entered via the microphone. We plan a sub-series of tests with users with functional speech that explore a speech recognition device (VUI). Finally, in line with suggestions by some IT-experts test subjects, we want to implement a mode with lemmas on disposition from a pre-selected topic or domain – similar to common AAC grid layouts (cf. the AAC systems presented in Section 3.1). When connecting the sentences, the participants chose the option to omit the selection of a connector. Only one user chose to add the connector *und* ‘and’ once. The leader of the writing workshop suspected that the users did not (yet) know how to systematically connect the sentences into a coherent text. Therefore, we plan to develop an active training mode in *EasyTalk* that teaches when and how to use text connectors (Reid et al., 2013). Additionally, the eye-tracking data revealed potential to improve the design of the Connector Panel. As the participants often did not pay attention to all connector options, we plan to provide a customizable design of the Connector Panel in the form of a flexible grid (e.g., displaying 4, 6 or 8 connectors), which can be assigned individual connector options. Finally, during sentence-connecting intervals, the Next-Word Panel should only be displayed after the user operated the button *Andere wählen* ‘Choose other’.

6.5 Summary of the evaluation results

In this section, we summarize the results of the individual test iterations and the resulting changes to *EasyTalk* during the HCD process to illustrate the iterative development process of the system and the progressive refinement of the user requirements (cf. Section 4.3). Overall, the system received positive feedback from all test groups. Nevertheless, each evaluation iteration provided important insights on how to further improve the system.

Figure 6.10 provides an overview of identified needs for improvement and the resulting changes to the system of each version.

EasyTalk Evolution

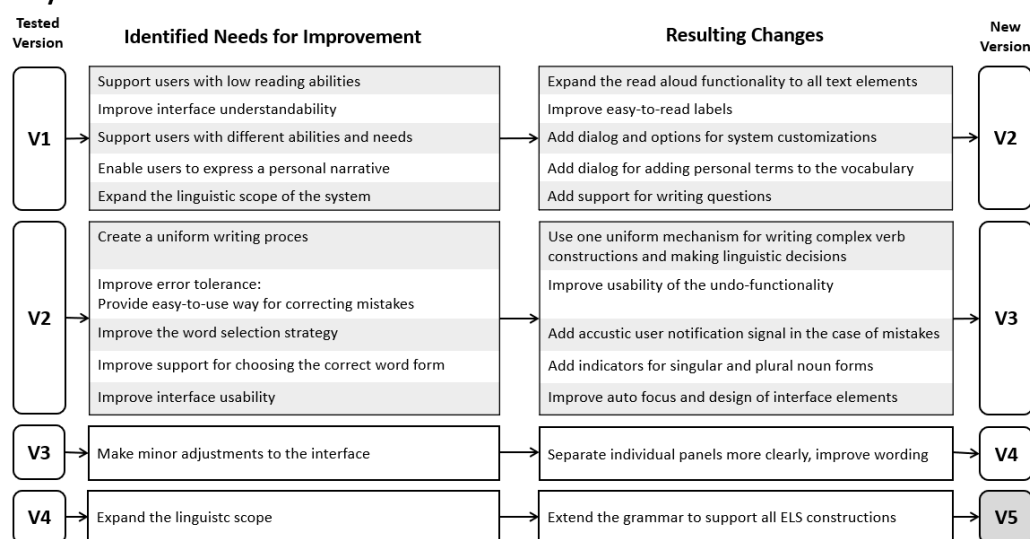


Figure 6.10: Overview of the system’s evolution over the course of the four test iterations conducted. From left to right: the tested version, the identified needs for system improvement to this version, the resulting changes to the system, and the resulting version of *EasyTalk*.

In the first evaluation iteration, the expert group agreed that the concept of *EasyTalk* was suitable for the target group. The system was generally perceived as easy to use and easy to understand. All experts appreciated the use of wh-question cues to communicate grammatical functions and decisions. Encouraged by the positive feedback, we continued with the original design concepts of *EasyTalk*. We implemented the linguistic scope of the system to the scope of LS – for example, by adding prepositional attributions and the possibility to formulate interrogative sentences. Based on the experts’ feedback on accessibility improvements, we added the option to expand the read-aloud function to all text elements in *EasyTalk* to the system settings. Additionally, we improved the understandability of the user interface by using easy-to-read labels for all interface elements. As the experts stressed the importance of being able to use personal terms, we added an easy-to-use dialog for adding personal terms to the dictionary of *EasyTalk*. In addition to being able to exchange the AAC symbols via the system’s file directory, we added the option to hide the AAC symbols to the system settings.

The second test iteration with beginner L2 learners with rudimentary computer skills confirmed that the system is suitable and easy to use for users with low computer skills. It successfully supported the low-literate users to write correct sentences by themselves. In the free-writing sessions, the L2 learners made significantly more use of the delete button than the participants of the expert group

in the cognitive walkthroughs conducted with pre-given sentences. In this context, we observed the need to improve the undo function of the system for correcting the current sentence. As a result, we changed the behavior of the delete button. (Before, the delete button removed the last entire sentence component. In the current *EasyTalk* version, it deletes the last word entered from the Sentence Panel.) Before testing with the low-literate L2 learners, we had expanded the scope of linguistic constructions to include the present perfect tense. In the tested version, *EasyTalk* automatically adapted the tense of a sentence to past tense, whenever the modifier ‘yesterday’ was used. However, we noticed that automatic changes of the system went unnoticed by the users. To create a more conscious and uniform writing process, we decided against the use of any automatic adaptations. Instead, we apply the mechanism established for modal verbs to all complex verb clusters – including combinations consisting of an auxiliary verb and a participle (e.g., *Ich habe gegessen* ‘I have eaten’). In the tests with the users with low-computer skills, we observed that typing mistakes might go unnoticed by users focusing on the keyboard, causing the completion list to render empty. To notify the user in time, we added the option to use an acoustic signal to direct the user’s attention back to the text input field once the list runs out of suggestions. To additionally support users in choosing the correct word forms in the case of nouns, we added the option to display an indicator for singular and plural next to noun forms in the suggestion list. Finally, we corrected some usability flaws we observed during the free-writing sessions, such as inconsistent focus functions of interactive interface elements. The improved element focus saves clicks and enhances user guidance throughout the writing process.

The third version of *EasyTalk* was tested by IT-expert L2 learners, who perceived *EasyTalk* as a good system that is easy to learn, very motivating, practical, friendly, pleasing, and very interesting. Due to the overall positive verdict, we made no significant changes to the system after testing with the IT expert group. Regardless, we have noted the suggestions made by the IT experts to further expand the system’s writing support in the future (cf. Section 8 for desirable future work).

The final evaluation iteration with members of the target group showed that *EasyTalk* also supports our target users well in writing. After only a short introduction, all participants were able to write several sentences themselves with the system. Even features that were not explicitly demonstrated – such as decision dialogues for complex verb clusters – were used without any problems. Eye-tracking data showed that users viewed the interface elements in an anticipated manner. With one exception, all participants voiced interest in using the system – indicating the system’s usefulness and joy of use. Of course, we still saw room for further development of our prototype system. When writing sentences, several users tried to use linguistic constructs beyond the scope of LS, such as the

use of simple past for modal verbs. Therefore, we expanded the linguistic scope of the system to comprise the full scope of ELS, which supports these linguistic features. Future work still pending includes, for example, improving the design of the Connector Panel, improving the word-entering strategy in the Next-Word Panel by exploring possibilities to compensate for spelling problems on the side of the users, and adding more options for system customizations in terms of accessibility (e.g., to accommodate visual or motor impairments).

Chapter 7

Results and discussion

This chapter considers all aspects of our system in relation to our research questions. We link the user requirements to the concepts we have developed and implemented in *EasyTalk* and relate them to the results of system evaluations obtained with different user groups.

First, we discuss the linguistic scope of *EasyTalk* and its appropriateness for the target audience (RQ1). Then, we consider how *EasyTalk* supports the user in writing at the personal skill level (RQ2) and whether this support is sufficient. Next, we address whether the profound use of wh-cues in *EasyTalk* successfully transforms concepts from text production theory into intuitive on-demand dialogues (RQ3). Finally, we address whether we succeeded in creating an accessible interface and supportive interaction patterns that do not overtax the user (RQ4).

7.1 The linguistic scope of the system

A successful writing aid must offer a linguistic scope adequate for the target group and support the use of personal terms (cf. RQ1, considering the user requirements refined in Section 4.3). In the following, we discuss *EasyTalk*'s base vocabulary, the use of personal terms, and the system's scope of linguistic constructions.

Base vocabulary *EasyTalk* uses the extensive *CELEX* lexicon, which covers over 51.000 base forms and contains many – in terms of LS – “difficult” words and technical terms. To provide a reasonable suggestion list, we restricted the test lexicon to the vocabulary of CEFR L2-level A1-A2, which is generally considered comparable to the vocabulary of LS texts (cf. Section 2.3). This choice was generally found to be appropriate for the users of the target audience participating in the free writing sessions (cf. Section 6.4). However, two participants wanted to use words not included in the restricted test lexicon used in the case study. In

these cases, the caretaker had to help to find a synonym or a different wording – a rather cognitively demanding and challenging process for the users, which interrupted the flow of writing. As previously emphasized by the expert group (cf. Section 6.1), this highlights the importance of being able to prepare custom vocabularies for the users. The AAC systems presented in Section 3.1 are often laboriously adapted to the user’s personal vocabulary. Text editors (cf. 3.2) are flexible here. In principle, the user can write anything. As a tradeoff, the correctness of the formulations is not guaranteed. We recommend offering both options for *EasyTalk* in the future: For one, the system should provide an easy-to-use way for to customize the base vocabulary for individual users. To use additional terms in an ongoing writing session, *EasyTalk* should offer just-in-time vocabulary – for example, by enabling access to the full scope of *CELEX* on demand or the possibility of adding free text elements within a sentence component.

Personal terms For the use of personal terms, *EasyTalk* offers a dialog for expanding the vocabulary. This dialog was used repeatedly in the case study with target users to add names of people they wanted to mention in their texts. The users appreciated the possibility of associating personal terms with custom symbols (e.g., a person with a photo). For AAC users, using familiar symbols is inextricably linked to their ability to express themselves. Therefore, *EasyTalk* allows users to adapt or exchange a set of symbols easily. In summary, the system provides good support for using personal terms, which can be extended in the manner described above.

Scope of linguistic constructions To appropriately define the linguistic scope of our writing system, we considered the rules of LS in terms of text generation (cf. Section 2.3) and recent research on the comprehensibility of the linguistic constructs licensed in LS for the target audience (cf. Section 2.4, in particular, see Bock (2019)). Based on a corpus study of authentic LS texts, written and spoken German by Harbusch and Steinmetz (2022), the grammatical scope of *EasyTalk* was set to ELS (cf. Section 2.4). The tests with the target group confirmed that the participants wanted to use ELS constructions beyond the core rules of LS, like negations with *nicht* ‘not’ or simple past tense for modal verbs. One participant in the case study wanted to start sentences with adverbs, e.g., ‘today’ or ‘hopefully’ – a typical pattern in spoken German not included in ELS. Following a reminder of the obligatory SVO structure of declarative sentences in *EasyTalk*, the participant started his sentences with the subject. However, further studies with the target audience should explore the extent to which the grammatical scope of *EasyTalk* (and ELS) could be further extended to allow more linguistic variety.

E(LS) does not license complex sentence clauses with commas. To link sen-

tences into a coherent text, *EasyTalk* promotes adding explicit discourse relations in the form of connectors in a separate line. We will detail on this in the upcoming Section addressing writing support at the text level.

Summary on the linguistic scope of the system In summary, we positively answered the question of an appropriate linguistic scope of a writing system for the user group by choosing ELS as the scope of grammatical constructions, a base vocabulary at CEFR L2-level A1-A2 expandable with personal words, and personal AAC symbols. Previous research has focused on the readability and comprehensibility of LS texts (cf. Section 2.3). In contrast, we have focused on constructions that the target audience likes to use to express themselves in (E)LS. Based on a corpus study (cf. Section 2.4), we theoretically assumed ELS to be an appropriate linguistic scope for our writing system. Our system evaluation showed that the target audience wants to use ELS constructions. Of course, we cannot generalize the results of our case study. The practical application of ELS in *EasyTalk* can, however, contribute to a better understanding of how the target audience wants to express themselves in writing by enabling observations and feedback on which linguistic constructions users tend to use, omit, or miss in writing. Therefore, we recommend an extensive study with a large number of target users using *EasyTalk* to create an authentic text base written by the target audience and gather profound feedback on the linguistic scope of the system.

7.2 Writing support at the personal proficiency level

In this Section, we discuss *EasyTalk*'s writing support at the user's personal skill level (cf. RQ2 and the respective user requirements defined in Section 4.3).

The analysis of the target group in Section 2.2 and the considerations regarding the context of use in Section 4.2 highlight the heterogeneity of our target group. Our user group cannot be assigned to a specific level of writing ability. Therefore, *EasyTalk* offers support at different levels of writing, based on *Alpha Levels* 1-4 for low-literate people (cf. Section 4.3; Grotlüschen and Riekmann (2011, 2012)). In more tangible terms, this means that *EasyTalk* must provide flexible support at the letter level, the word level, the sentence level, and the text level. At the same time, the system must compensate for a lack of writing skills on the side of the user, i.e., it is responsible to maintain the grammatical correctness of the content produced. To facilitate text understandability, *EasyTalk* must support the creation of text coherence and remind the user to add audience-design elements. In the following, we discuss our design concepts for these aspects (cf. Section 4.4) in their practical implementation in *EasyTalk* (cf. Section 5.4) in relation to

the results of their evaluation in free writing sessions low-literate L2 learners of German (cf. Section 7.3) and target users (cf. Section 6.4).

Letter- and word-level support On the letter and word level (cf. Section 5.4.1), *EasyTalk* supports the user with AAC symbols that facilitate word recognition, read-aloud functionality to remedy reading deficits, and spelling support in the form of word suggestions. All these elements were rated positively in the test sessions.

During the free writing sessions, the beginner German learners (cf. Section 6.2) and the target users (cf. Section 6.4) made repeated use of the read-aloud functionality and expressed positive opinions about it.

The AAC symbols were also positively received. The L2 learners stated that the symbols helped them to recognize words. The target users all preferred to use *EasyTalk* with symbols and export their written texts with symbols. It is an open question how users who communicate with AAC systems would judge the use of symbols in *EasyTalk*. To investigate this, tests with this specific user group would be needed. Such tests could also evaluate if AAC users accustomed to grid systems can successfully write texts with *EasyTalk*.

Providing correct word-form suggestions for the current sentential context was appreciated by all participants of our four test iterations. The L2 learner groups appreciated the automatic correct inflection. The expert group anticipated that the list could provide the users with a feeling of security and confidence. This assumption was confirmed by target users, who used the word-form suggestions as a spelling aid: One participant said, “Writing to the end is better.” He judged the completion list as helpful in preventing spelling mistakes. Like him, several participants used the word form suggestions to help them type words from beginning to end. However, we observed that the usability of the suggestion list could be improved for users with low computer skills and spelling problems, which we discuss in detail in Section 7.4.

Sentence-level support *Scaffolded writing* (Harbusch and Kempen, 2011) using a natural language paraphrase generator (Harbusch et al., 2007, 2014) enables *EasyTalk* to provide profound linguistic support on the sentence level minimizing the need for grammatical knowledge on the side of the user (cf. Section 5.4.2): In addition to suggesting correctly inflected word forms for any sentential context, the system ensures correct word order and the completeness of a sentence by ensuring that all sentence components are complete and all obligatory arguments are mentioned according to the verb-valency frame. Moreover, the system visualizes the sentence plan using wh-question cues. This reminds the user which sentence components have already been written and which ones can still be added – e.g., to

facilitate text understandability, which we address below in Section 7.2.

The evaluation with members of the target audience implies that *EasyTalk* successfully supports low-literate users with IDD and/or CCN to write at the sentence level: All target users successfully wrote a series of at least three sentences with an average of four words in their free writing sessions – according to the leader of the writing workshop, a very good result. The texts included modifiers, complex verb constructions, and questions. As mentioned above, the word form suggestions were positively perceived by all test participants. The word order support was particularly appreciated by L2 learners, whose native languages require a different word order – a relevant aspect for such target users who are not native German speakers. For non-native users, another phenomenon that occurred with some L2 learners could be relevant: The users can formulate statements in *EasyTalk* that are grammatically correct per se but are not used that way in German (e.g., *nach Zoo fahren* ‘to drive to zoo’ – cf. Sections 6.2 or 6.3). It would be necessary to check *EasyTalk*’s suggestions for their meaning in the context of the sentence to notify the user in such cases. For this purpose, the system would have to be extended by a content representation.

Text-level support On the text level, *EasyTalk* offers explicit discourse relations in the form of connectors to express the grammatical function of a sentence to create text coherence. This concept was rated positively in the expert interviews (cf. Section 6.1) and test sessions with L2 learners (cf. Sections 6.2 and 6.3). Some IT-expert L2 learners expressed the wish to additionally employ content representation providing feedback on the quality of their choice of connectors. In the case study with the target group (cf. Section 6.4), one participant expressed a positive opinion on the sentence-connecting feature: “The connectors between sentences are important. Sometimes there are longer sentences. This way, you can do them piece by piece.” However, he was the only participant in the case study who used this option. Since we did not ask why the users did not connect the sentences, we can only speculate. Possibly, the feature exceeded the skill level of the participants, who may have had yet to reach a writing level that included writing coherent texts. Another reason may be that we did not explicitly ask the case study participants to connect their sentences. (We intentionally chose not to influence the participants’ text writing to obtain an authentic impression of how they would use the system.) Finally, we identified options to improve the usability of the respective interface dialog, which we discuss in Section 7.4, which focuses on the system’s usability. We recommend performing additional tests with target users dedicated to the sentence-connecting process to obtain more insights on this aspect.

Facilitating text understandability To facilitate text understandability for the readership, *EasyTalk* reminds the user to add audience design elements to their sentences – e.g., time and place of an event. The system uses wh-cues to communicate which sentence modifiers can (still) be added to the sentence (cf. Section 5.4.2). In the case study with target users (cf. Section 6.4), all participants spontaneously supplemented their sentences with modifiers providing the reader with important information. The eye-tracking data reveal that users looked at the visualization of the sentence plan prior to adding this information (cf. Figure 6.7). The test results indicate that displaying possible extensions to the sentence encourages the user to add information. Additionally, participants voiced positive opinions on the design of the respective interface elements, which we discuss in Section 7.3 dedicated to transforming linguistic decisions into intuitive dialogues. While testing has shown that our target users use the features offered to add audience design elements, it would be interesting to know if users would have added this information to the same extent without the support offered. We recommend further testing to explore this aspect.

Summary on the writing support at the personal skill level Using *EasyTalk* for the first time, our target users wrote a series of complex sentences and expressed interest in using the system in the future. Testing showed that our target users utilized the support provided at different levels of the sentence-writing process and added audience design elements. Due to the characteristics of the target group (i.e., CCN, IDD, low literacy, and low computer skills), we did not receive detailed verbal feedback on the individual writing support features. Hence, we base our insights on our observations and the recorded eye-tracking data. Overall, we rate the system’s support at the user’s writing level as positive. However, the text-level support to combine sentences using connectors needs further consideration. Although it was rated positively in tests with domain experts and substitute users, the target group hardly used it. The reason for this needs to be explored in more detail. One aspect could be necessary improvements in its design, which the eye-tracking data suggests – or the need to teach the concept of connecting sentences into a coherent text.

We are pleased with the positive test results. However, we cannot generalize the results of a case study. Furthermore, we cannot say whether *EasyTalk* enabled users to write beyond their skill level, as our case study with the target users did not include an assessment of the writing skills of the individual participants. Gaining more precise insights into this aspect would require more extensive tests with a larger number of participants preceded by a classification of the writing abilities of the individual participants.

7.3 Transforming text-production theory-concepts into intuitive on-demand dialogues

To provide profound NLG support, *EasyTalk* uses a natural language paraphrase generator that guarantees the completeness of the sentence components and the sentence as a whole, ensures correct word order, and generates correct word forms for any sentential context (cf. Section 5.2). Therefore, *EasyTalk* builds up a derivation tree in scaffolded writing in a dialog with the user, which requires communicating linguistic decisions. For the target group, it is essential to avoid using linguistic terminology in this process. This section discusses how *EasyTalk* translates concepts from text-production theory into intuitive on-demand dialogues that guide the users through the writing process (RQ3, refined by the user requirements in Section 4.3).

The system relies on intuitive, easy-to-read, and easy-to-understand wording to accommodate the low literacy skills of the target users. The process of writing a text is divided into separate, easy-to-label steps to reduce its complexity, namely: adding the *next word* (1) to a *new sentence* (2) and *connecting the sentences* (3) into a *written text* (4) (cf. Section 5.3). Starting a new sentence, the user intuitively defines the sentence type (e.g., declaration or question) by selecting the appropriate punctuation mark. Each word the user adds to the sentence is assigned to a component in the internal sentence plan of its NLG component. The system uses the sentence plan in the user interface to visualize the current component, finished components, and components that can still be filled, providing the user with orientation in the sentence-writing process. A wh-question communicates the grammatical function of each sentence component to the user. The concept of wh-cues (cf. Section 4.4) is used in school to teach sentence components (e.g., ‘Who?’ asks for the subject of a sentence) and common in spoken dialogues to ask for information (e.g., ‘When?’ asks for the time of an event). To further assist the user in the writing process, the wh-labels in *EasyTalk* are interactive: When the user selects a wh-label, it is highlighted, and the system provides word suggestions according to its grammatical function, presenting the user with suitable, correctly inflected word forms for the respective sentence context. *EasyTalk* offers two writing modes (cf. Section 5.4.2.6): In the *default mode*, one wh-cue is always active. The user can deliberately change this cue to switch to another sentence component. In the *free-writing mode*, the system does not require an active wh-cue. The user can either first activate a cue or directly select a word form from the suggestion list. In the latter case, *EasyTalk* uses feature unification to assign the word form to a sentence component. If the unification result is ambiguous (e.g., if a word form can belong to a direct or indirect object), the system presents a decision dialog to the user, which communicates all possible options in the form

of the corresponding wh-cues. The mechanism of the decision dialog is also used for all other necessary linguistic decisions on the sentence level to create a uniform writing process. For example, after choosing an auxiliary verb, the system asks whether the user wants to continue the sentence writing in the present or present perfect tense.

In all four test iterations, the dialogues and the wh-cues were rated as simple, intuitive, easy to understand, and easy to use. Most importantly, our target users operated the dialogues without problems after only one example sentence introducing the system, which implies the intuitive comprehensibility of the dialogues for the target group. The eye-tracking data and session recordings show that the target users performed the individual writing steps as expected. Users who wanted to ask a question successfully switched sentence types by changing the punctuation mark. We rate the interactive wh-cues as particularly successful: One participant expressed that the wh-questions helped him concentrate. Our observations and the eye-tracking data indicate several use cases (cf. Figure 6.7):

- Selecting a cue to filter the suggestion list communicates a conscious choice of the next sentence component to be written.
- Looking back and forth between the suggestion list and the active cue indicates that participants used the cues as a reminder of the current component.
- Reading through the cues (with or without selecting one) before starting a new component indicates that the cues provide the user with an overview of the open components and can offer inspiration on how to continue the sentence.

Although we had not demonstrated the decision dialogues in the system introduction, participants successfully used them – for example, to write present perfect tense and in the context of modal verbs.

In conclusion, our design concepts and their application in *EasyTalk* successfully transform linguistic decisions into intuitive on-demand dialogues. Compared to the symbol- and text-based writing systems for the target group presented in Sections 3.1 and 3.2, the *EasyTalk* dialogues minimize the necessary grammatical knowledge on the part of the user. Even to write complex sentences consisting of several components, the user does not have to worry about spelling or choosing the correct word forms manually. Due to *EasyTalk*'s interactive user guidance, the user is guided through the writing process instead of being confronted with a blank page. As always, we cannot generalize the results of our case study. However, we are confident that the concepts would generally test positive with the target group. We base this conclusion on the positive results of all test iterations denoting that the chosen concepts are suitable for users with low reading, writing, computer, and German skills.

7.4 Usability of the system for the target group

In this section, we review the usability of *EasyTalk* for the target group to discuss whether we succeeded in creating an accessible interface with supportive interaction patterns that do not overtax the user (RQ4).

We obtained positive evaluation results in all test iterations on the usability of *EasyTalk*'s interface and interaction patterns. In the third test session, we performed guided usability tests with IT-expert L2 learners to evaluate the system's usability from an expert's point of view. In the UEQ questionnaire, *EasyTalk* scored good to very good ratings in all categories (cf. Table 6.4 and Figure 6.3) – a strong indicator ruling out any significant deficiencies in usability. However, we cannot draw any direct conclusions about the system's usability for the target group based on the ratings of IT expert users. Therefore, we were pleased that the subsequent case study with target users confirmed the positive results: Despite their low literacy and computer skills (two participants stated that they were writing on a PC for the first time), all participants could write sentences with *EasyTalk* after a short introduction in the form of one demo sentence performed by the interviewer. We attribute these positive results to the overall HCD process used to develop the system (cf. Section 4), which allowed us to consider various viewpoints of different stakeholders and diverse user groups.

In the following, we evaluate the usability of *EasyTalk* based on the test results of the case study with the target group in relation to the seven interaction principles of the usability-norm *ISO 9241-110*¹: suitability for the user's tasks, self-descriptiveness, conformity with user expectations, learnability, controllability, error robustness, and user engagement.

1. Suitability for the user's tasks The evaluation confirmed that *EasyTalk* provides target users with appropriate functionality and writing support. All test participants successfully wrote a series of sentences. To minimize the actions needed to write a sentence, users can select word suggestions from a filterable list, which reduces the need for typing. Optionally, *EasyTalk* offers a *free writing mode* which uses feature unification to assign words to their respective sentence components automatically. Only in case of ambiguities the user needs to make an active choice in a decision dialog. Activating a wh-cue becomes an optional action in this case – reducing the required user actions to a minimum.

2. Self-descriptiveness In *EasyTalk* the division of the text-writing process into four panels (the Next-Word-Panel, the Sentence Panel, the Connector Panel, and

¹ISO 9241-110:2020 – *Ergonomics of human-system interaction - Part 110: Interaction principles* www.iso.org/standard/75258.html

the Text Panel – see Section 5.3 for their introduction) establishes a clear structure and straightforwardly communicates the system’s functionality. All test participants used the system without asking for help. Our observations and the eye-tracking data do not show any signs of confusion or searching around for functionality. The eye-tracking data show that participants used the panels according to their intended purpose (cf. Figure 6.6): When selecting words, the users focused on the Next-Word Panel. The Sentence Panel was used to read through the current sentence. The Text Panel was used to back up to flow of thoughts after completing a sentence (see, e.g., Figure 6.5a) or before starting a new one (see, e.g., Figure 6.5b). Although the users read through the connectors (see, e.g., Figure 6.9), all but one participant chose to skip the choice of a connector throughout their entire writing session. As we did not ask the users for feedback on the dialog, we can only speculate why this was the case. The eye-tracking data indicates that the interface and interaction design of the Connector Panel could be improved: For one, the interactivity of the Connector Panel could be emphasized. Additionally, not all connectors were viewed equally – for example, the second row of connectors was looked at considerably less. One option to improve the dialog could be to reduce the number of connectors on display for beginner users. We recommend further design and evaluation iterations specifically for this Panel.

3. Conformity with user expectations *EasyTalk* provides consistent support throughout the text-writing process. As mentioned, the evaluation results showed that *EasyTalk* successfully guided the users through the writing process. We did not observe confusion or searching for functionality on the users’ side. Nevertheless, we identified a minor detail for improvement: Some users tried to select word forms in the Next-Word Panel using a double instead of a single click. This behavior caused the system to add two words in a row, and the users had to delete the last involuntary chosen word form from the sentence. Therefore, we propose to add the choice option between using single or double click to select words in the system settings.

4. Learnability The case study showed that the system is easy and quick to learn for the target users. The system’s functionality was explained in the test sessions using a single example sentence. This sentence was either written by the interviewer or directly by the user under the guidance of the interviewer. After this brief introduction, all users were able to write in *EasyTalk* themselves. The participants successfully used elements that the interviewer had not presented in the demo, such as operating decision dialogues.

5. Controllability All participants were able to control the system successfully.

Two users with visual impairments expressed their wish to highlight or enlarge the mouse cursor for better visibility. In addition, we helped participants with motor impairments to control the standard laptop mouse. These users could operate *EasyTalk* with input devices adapted to their personal conditions. Depending on the needs and abilities of the individual users, *EasyTalk* would need to offer additional options for customization to be accessible for a large spectrum of target users and employ more control options, such as eye control or scanning. An in-depth consideration of these aspects was not the focus of our research and would have exceeded our scope. However, we recommend offering more options to control the system in later versions.

Focusing on the writing support of the system, we identified potential for improving the word selection strategy in the Next-Word Panel for users with low spelling and keyboard typing skills: In the free writing sessions, we observed that the flow of the writing process is interrupted when the suggestion list remains empty due to spelling or typing errors. We observed this repeatedly for users who focus heavily on the keyboard due to a lack of keyboard typing skills. They first notice their mistakes once they look back up at the screen. At this point, the suggestion is already empty. The users have to consider where they made a mistake and correct their input. *EasyTalk* offers an optional acoustic signal to direct the user's attention back to the screen once the suggestion list becomes empty. In addition, making the Next-Word Panel more tolerant towards spelling or typing mistakes by offering similarly written or pronounced word forms could improve the usability of the dialog. However, this could also irritate our target users, as their input might not be a direct match for the word form suggestions on display. Therefore, we recommend this as an optional feature.

6. Error robustness In *EasyTalk* provides easy, straightforward ways to correct errors: Finished sentences can be deleted from the Text Panel by clicking the X-button displayed at the right edge of each sentence row. Writing mistakes in the current sentence can be undone by operating the X-button in the Sentence Panel, which deletes the last word added to the current sentence. Typing mistakes in the Next-Word Panel can be corrected by the use of the *back*-button on the keyboard. Our target users corrected their mistakes during the free text-writing sessions without problems.

7. User engagement Participants showed interest in using *EasyTalk* in all test sessions, indicating that the system is inviting, motivating, and enjoyable to use. The expert group expressed various ideas to use the system with the target group, such as collaborative writing of LS text and weekly logs of residents of facilities for assisted living or teaching writing skills in a school context (cf. Section 6.1).

The L2 learners (cf. Sections 6.2 and 6.3) were interested in using the *EasyTalk* to practice German. They particularly appreciated that *EasyTalk* offered them freedom of expression, in contrast to German learning systems based on fill-the-gap exercises. Most importantly, all but one of the participants in the target group (cf. Section 6.4) stated after their writing sessions that they would like to continue using the system and would like to employ it in their writing workshop soon. Based on their positive opinions, the head of the writing workshop asked for a version of the system.

7.5 Summary

We have presented *EasyTalk* as a viable answer to our research questions: Offering the linguistic scope of ELS and a personally expandable base vocabulary augmented with AAC symbols, *EasyTalk* provides our target group with appropriate linguistic means to express themselves in writing (RQ1, cf. Section 7.1). *EasyTalk* supports users at different levels of writing in creating correct and coherent texts (RQ2, cf. Section 7.2). The system reminds users to add important audience design elements to their texts to facilitate text comprehensibility for the readers. The system guides the user through the writing process in a successful way, using intuitive dialogues that communicate linguistic information in an easy-to-understand way (RQ3, cf. Section 7.3). Overall, the interface and the interaction patterns demonstrated suitability for target users (RQ4, cf. Section 7.4). Still, we took note of points for usability improvements in the form of optional customizations – for example, increasing the error tolerance in the word selection strategy or redesigning the Connector Panel to motivate users to utilize it more to link their sentences.

Throughout the iterative system development process (cf. Section 4.5), different versions of our system were evaluated by different user groups with positive results (cf. Section 6.5). Testing the matured system in a case study with target users showed that *EasyTalk* successfully supported the participants in their writing who expressed interest in continuing to use the system as a writing aid. Generalizing the results of our case study requires further testing with a large number of users representing a wide range of needs and abilities of this particularly heterogeneous target group. To gain insight into whether *EasyTalk* helps users write beyond their current skill level, we recommend classifying the participants' writing abilities in future evaluations. The following Section presents ideas for further options to develop the system and future research based on *EasyTalk* for the benefit of the target group.

Chapter 8

Future Research

The *EasyTalk* prototype proved to be a suitable writing aid for our target users. This section presents possible further developments of the system and desirable future research.

The results presented and discussed in Chapter 7 mentioned two main aspects limiting a generalization of our evaluation results. One aspect is the small number of target users participating in our case study. While *discount testing* with eight users confirmed that *EasyTalk* can provide users with fruitful writing support, testing with a large number of participants representing the spectrum of needs and abilities of our very heterogeneous target group would add insights on how to make the system accessible for a wide range of users. To further examine the writing support at the individual stages of text-writing, we recommend classifying the participants' writing abilities in future tests with the system. This would also allow insights on the following interesting research questions:

- Can *EasyTalk* help users to write beyond their current skill level?
- Which linguistic constructs do users of *EasyTalk* prefer to use in their texts?
- Is there a need to expand the scope of ELS?

In the scope of this work, *EasyTalk* has matured into an advanced prototype. In order to make the system fit for real-world everyday usage, further extensions beyond the scope of our research are needed, such as:

- additional options for system customization in terms of accessibility and barrier-free control options (e.g., eye control, scanning, or AAC grid layouts),
- providing quick and easy ways to personalize the vocabulary,

- explore the option to use on-the-fly vocabulary to enable users to use terms without having to add them to the vocabulary first,
- text sharing functions (e.g., via email or messenger).

In addition to preparing *EasyTalk* for practical use, we would like to implement the concept of *EasyText* presented in Section 3.4 to explore the system's capability to teach text production to its users in the form of a digital writing workshop. Moreover, we would like to add an active teaching unit for connecting sentences to the system.

Finally, we recommend developing an app version of the system for smartphones and tablets to offer more flexibility of use in line with recent advancements in (assistive) technology and – most importantly – to introduce the system to a larger audience to assist them in writing and participating in written discourse.

Chapter 9

Conclusion

Leichte Sprache is a simplified variety of German developed as an instrument for inclusion aiming to overcome language barriers. It is used to provide barrier-free texts for a broad spectrum of people, including low-literate individuals with learning difficulties, intellectual or developmental disabilities and/or complex communication needs. So far, LS is mainly provided for its target group by authors proficient in standard German. It is, however, rarely written by its target readers. One reason may be the need for more technical support in the form of an assistive writing system. Modern author support tools for LS include, for example, automatic text simplification or rule-based validation of syntactical text simplicity. However, these systems target literate writers proficient in standard German. To the best of our knowledge, there is no recent assistive writing aid using profound linguistic processing by computer designed to support low literate users with IDD and/or CCN in writing.

This work set out to enable this user group to participate in written discourse themselves by creating a writing system based on natural language processing whose linguistic support and ergonomic software design meet their specific needs. Pursuing this goal raised several research questions: What linguistic constructions might LS readers like to use in a writing tool? How can natural language processing support our target users to write correct, understandable, coherent text at their personal level of proficiency? Can we transform concepts from text-production theory into intuitive on-demand dialogues? What does it mean to design an accessible interface and supportive interaction patterns that do not overtax the user?

In response to these research questions, we developed *EasyTalk*, an assistive writing system that considers the target group's specific requirements. Based on observations from a corpus study into authentic LS text, spoken and written German (Harbusch and Steinmetz, 2022) and insights from the LeiSA study (Bock, 2019), which analyzed the ease of comprehensibility of linguistic constructions with members of the LS audience, *EasyTalk* supports the scope of ELS, an ex-

tended variant of LS. The system strives for correct and understandable text both during sentence production and sentence combination. To minimize the writing skills needed, *EasyTalk* uses natural language processing to a large extent and provides interactive user guidance that breaks down writing into an intuitive step-by-step process. The system communicates linguistic decisions using easy-to-understand wh-question cues (Who, When, . . .). As the system targets a very heterogeneous user group, it supports users in writing at their individual level of proficiency by offering support at the letter, word, sentence, and text level. On the letter and word level, *EasyTalk* offers a personalized vocabulary supplemented with customizable communication symbols. All words and commands in *EasyTalk* can be read aloud on demand. Within a sentence, the system maintains syntactic correctness by building up a derivation tree in so-called scaffolded writing in an easy-to-use dialog with the user based on wh-question cues to provide a context-sensitively filtered list of word forms by using a natural language paraphrase generator. In addition, the wh-cues remind users to add text elements that enhance text comprehensibility in terms of audience design (e.g., time and place of an event). Between the sentences, *EasyTalk* prompts the user to add explicit coherence specifications inspired by Rhetorical Relation Theory to connect sentences – thus, expressing the intended communicative goal of the sentence and creating text coherence. *EasyTalk*'s clearly structured user interface and consistent user interactions throughout the writing process suit users with low literacy skills and low computer skills and take into account conditions such as intellectual difficulties, memory, or cognitive flexibility limitations.

The development of *EasyTalk* followed the principles of human-centered design to tailor the system to the target group's needs. We incorporated the perspectives of all stakeholders (i.e., users, their helpers, and potential readers of the texts) into the development process. We matured the system in iterative development cycles, combined with targeted evaluations of specific aspects conducted in case studies with expert groups from the fields of IDD, CCN, LS, and IT, as well as second language learners of German. In all test sessions, we received positive feedback. In a final case study, members of the target audience tested the system in free writing sessions. The system guided the users through the writing process in a successful way. Despite their low literacy and computer skills, all participants successfully typed a series of sentences. The evaluation generally corroborates the easy and intuitive design of *EasyTalk*. Half of the participants spontaneously skipped the interactive example rehearsal and directly typed their own sentences without problems. Even features that were not explicitly demonstrated – such as decision dialogues for writing complex verb clusters – were used without any problems. In addition to our observations, we used eye tracking to obtain objective insights into the target users' interaction with the system. The eye-tracking data showed that users viewed the interface elements in an anticipated manner.

The wh-question cues provide supportive orientation and help to focus and were used to add audience design elements to the user texts. However, eye-tracking data hints that the offer of connectors in the sentence-combining interface element should be improved. In addition, we took note of points to improve accessibility for specific user needs in the form of personal system customizations, such as optimizing the system's interface for users with visual impairments. With one exception, all participants voiced interest in using the system – indicating the system's usefulness and joy of use. The study confirmed that adults with IDD and/or CCN who have low reading, writing, and computer skills could instantaneously type complete and correct sentences in ELS using *EasyTalk*. Nevertheless, the generalisability of case study results is – of course – limited. This work showed that it is possible to support low literate users with IDD and/or CCN in writing by combining NLG techniques with simple and intuitive user guidance. In addition to offering our target audience a writing support system, our insights contribute to understanding how NLP techniques can be employed to benefit the target audience. We hope to inspire future research in this area, as the development of NLP writing aids can make an important contribution to inclusion by promoting active participation in written discourse by the target group.

Desirable future work includes longitudinal studies with a larger group of test participants. For these studies, we recommend assessing the individual users' writing skills. This will lead to new insights on whether our writing support system can not only assist users in writing but additionally help them write beyond their skill level and improve their writing skills. More broadly, research is also needed to determine in collaboration with the target group whether the linguistic scope of ELS for LS writing systems is sufficient or whether users demand more freedom in expression, such as using a more flexible word order than subject-verb-object-adjunct order. Additionally, we plan to implement the usability improvements of the system mentioned above – i.e., making over the sentence-combining functionality and enhancing it with an active teaching unit that helps users purposefully choose connectors to create text coherence. Moreover, we want to offer more options for system customization to meet the accessibility needs of individual users. Finally, a natural progression of this work is developing an app version of *EasyTalk* to make the system available to a broader group of users, thus encouraging the production of LS text by the community itself.

Appendix A

Original Test Data

This appendix includes the session notes of all test sessions conducted during the iterative system evaluations (cf. Chapter 4, Section 4.5 and Chapter 6). Section A.1 contains the test data of the interviews with domain experts, Section A.2 the notes of the sessions with L2 German learners with low literacy and computer skills, Section A.3 the notes of the IT-expert L2 learners, and Section A.4 the notes of the case study conducted with members of the target group.

A.1 Interviews with domain experts

In this Section, we provide the detailed session notes for the interview and think-aloud probes conducted during the first evaluation cycle. Note that the original notes were taken in German and then translated into English for their presentation in this work. For the evaluation of the expert interviews, see Section 6.1.

A.1.1 LS reader

RQ1: What was easy and what was difficult during typing?

- The introduction was too short. It would be good to have a more detailed introduction to the program. Showing something only once was too little. For example, a video tutorial would be very good.
- If you have basic computer skills, the system is easy to use.
- The grammar support is very good. It is very good because you can practice with it and make fewer mistakes when writing.
- It is very important that the symbols match the words. Otherwise, it is confusing.

- It [the system] is very easy to use. You don not have to think much about it.
- The system needs more words. When words are missing, it is confusing, and you have to think a lot to find another word to use.

RQ2: How do you evaluate the AAC-support features?

- Images are very important in LS. The symbols in the system also help.
- The symbols are very good. But everyone should be able to choose their own symbols.
- The symbols for the conjunctions [in the Connector Panel] are not understandable.
- If symbols are not understandable, they are confusing and should be exchanged or left out.

RQ3: Do you think the system is useful? Please elaborate on your choice!

- The system is very useful, and I like that an LS writing system is being made.
- I normally use MS Word to write. But only because I do not have a better choice. If I had a better writing tool, I would use that.
- Other people can benefit even more from the system. For example, pupils in the third or fourth grade who are still practicing writing.
- I like that the system makes writing in LS and, in general, easier.
- The system is great for writing and practicing building sentences.

RQ4: Would you like to use the system? Please elaborate on your choice!

- When the system is ready, I want to use it.

Notes from the think-aloud probes:

- On viewing the user interface the participant says: The word *Satzhistorie* 'sentence history' is too difficult. It needs to be replaced.
- The participant notes that it would be good if the symbols could be easily turned off and on.

- The participant says that the dictionary should be more clearly separated from the Sentence Panel.
- The participant voice that he would appreciate color codes for the different word types. Preferably similar to the system that he knows from school classes. There, the verbs were colored in red, the nouns in black, and the adjectives in blue. After a moment, the participant adds that it would be even better if each user could choose their own color coding system.
- The participant criticizes that the system does not support writing questions.

A.1.2 LS writer

RQ1: What was easy and what was difficult during typing?

- The symbols associated with the conjunctions are not common. They need to be exchangeable.
- The UI should be reworked to be more intuitive.
- The font size should be bigger.
- The placement of the buttons for 'OK', 'Continue', and 'Delete' is not understandable.

RQ2: How do you evaluate the AAC-support features?

- The symbols associated with the conjunctions are not common. They need to be exchangeable.
- It's OK to show the conjunctions between the sentences in a separate way. This way, each sentence has SVO order.
- The use of symbols is not adequate for all users. Users should be able to choose if they want to use the system with or without symbols.

RQ3: Do you think the system is useful? Please elaborate on your choice!

- The system is too complicated for people with severe [cognitive] disabilities. People with learning difficulties are a suitable target group.

RQ4: Would you like to use the system? Please elaborate on your choice!

- The system could be used to write LS texts together with clients.

Notes from the think-aloud probes:

- The symbols associated with the conjunctions are confusing.
- The offered sentence structure should be adaptable for different target groups.
- The system should support adding special and personal terms.
- The system should offer different, predefined vocabularies of varying sizes for different users.
- The system should offer different settings like difficulty levels: easy, medium, and high. This would require less configuration by the caretaker to customize the system for the user.

A.1.3 AAC expert

RQ1: What was easy and what was difficult during typing?

- It is good that the system enforces the sentence structure of LS.

RQ2: How do you evaluate the AAC-support features?

- The combination of symbols and words is very good.
- The read-aloud function is very good.
- Not only sentences but all words need to be able to be read aloud.

RQ3: Do you think the system is useful? Please elaborate on your choice!

- The dictionary of the system can help clients with writing, who know single letters but cannot form words themselves.

RQ4: Would you like to use the system? Please elaborate on your choice!

- Yes. For example, to create texts for pamphlets or weekly logs of residents of facilities for assisted living.
- To use the system, it needs a customizable, extendable dictionary that supports adding personal words.
- The symbols need to be exchangeable.

Notes from the think-aloud probes:

- The combination of words and images reminds of *Lesestränge* ‘reading chains’ from literacy training. These are, for example, used in children’s books with symbols for language acquisition.

- The system must support writing sentences like *Ina und Jana essen Eis*. ‘Ina and Jana eat ice cream.’ or *Ina isst mit Jana Eis*. ‘Ina eats ice cream with Jana.’
- In sign language, tenses and punctuation marks are always chosen at the beginning of a sentence.
- The colon should be offered in the Connector Panel.

A.1.4 Domain expert

RQ1: What was easy and what was difficult during typing?

- The label *Satzhistorie* ‘sentence history’ is a difficult word and should be exchanged.
- It is difficult to understand that a sentence is written component by component. What is a sentence component? When does a sentence component end? Which components are there?
- Having to skip unused sentence components may confuse the user.

RQ2: How do you evaluate the AAC-support features?

- The reading function is good, but it should be able to read aloud all text elements.

RQ3: Do you think the system is useful? Please elaborate on your choice!

- It is useful.

RQ4: Would you like to use the system? Please elaborate on your choice!

- There is no context for me to use it. But for other people it is useful.

Notes from the think-aloud probes:

- It should be possible to delete single sentence components one by one instead of deleting the whole sentence at once.
- It should be possible to delete by pressing the ‘delete’ key on the keyboard and using the ‘delete’ button offered by the system.
- Using adverbs may confuse some users. The possibility of using them should be optional.

- The dictionary is too small.
- The Sentence Panel is too big.
- Visually, the dictionary needs to be more clearly separated from the Sentence Panel.
- The first word in the dictionary should be highlighted more clearly as selected.
- The system should offer to enter ‘free text’ elements and personal terms spontaneously.

A.2 L2 learners with low computer skills

Here, we present the notes from the test sessions with the L2 learners of German with low computer skills and low literacy skills in the German language – P1, P2, and P3 – evaluated in Section 6.2. The original notes were taken in German and were translated into English for their presentation in this work.

A.2.1 Participant P1

Profile: P1 is a 28-year-old male. His mother tongue is Amharic. His German skills are at CEFR level A1-A2. His computer skills are low.

Q1: Was the system easy or hard to use for you?

- It is easy, but it needs a better introduction, like a video.

Q2: Was the sentence construction process easy or hard to understand for you?

- It is OK.

Q3: Do you feel supported in increasing your grammatical skills in German by using the system?

- Yes, as a beginner.
- It is good to learn articles, conjugation, and word order.

Q4: What do you think about the symbols?

- They are for kids, right?
- At the top, with the sentences, there should be no images. Only for the sentences and in the word list.

Additional observations from the test session in chronological order:

- **O1:** P1 tries to start the first sentence with a verb.
- **O2:** Several times, P1 presses enter twice in a row causing *EasyTalk* to immediately choose two words in a row without P1 being able to read both words.
- **O3:** P1 writes: *Ich mache die Wäschen*. ‘I do the laundries.’
Note: while the sentence is grammatically correct, it is semantically falsy. The article *die* is valid for singular and plural forms. We assume that P1 did not notice that the singular word form was required here or did not know the correct word form.
- **O4:** After selecting a word from the Next-Word-Panel, the system does not automatically refocus the text-input field.
- **O5:** P1 wants to start a declaration with a verb. He asks why he cannot write *Kaufen Buch*. ‘Buy book.’. The interviewer responds that the sentence needs to start with a subject, like a person or another actor. P1 nods. Then, P1 writes: *Ich kaufe* ‘I buy’. Then, P1 pauses for a moment before asking for the correct article for *Buch* ‘book’. P1 says it is good that one cannot write wrong grammar. But he says it is not easy to find the correct article immediately. The interviewer asks what he thinks about that. P1 responds that he thinks that it is OK. This way, he has to think to find the correct article.
- **O7:** P1 praises the system’s automatic verb conjugation. During the session, he repeatedly reads the conjugated verb forms out loud.
- **O8:** After choosing a conjunction, *EasyTalk* does not automatically focus the text-input field in the Next-Word Panel as it should.
- **O9:** P1 wants to write *Ich habe Zeit*. ‘I have time.’. Instead of starting with a Z, however, he types am S as the first letter. He reads through the word list, trying to find the word in vain. Then, the interviewer offers support with spelling.
- **O10:** Writing sentences with modal verbs, P1 uses the word order required by *EasyTalk*. E.g., he writes *Ich will lesen das Buch*. ‘I want to read the book.’ which *EasyTalk* automatically corrects to *Ich will das Buch lesen*. ‘I want to read the book.’ (In German, the infinitive verb is placed at the end of the sentence after the object.) The interviewer asks P1 if this is

the word order of his mother tongue (Amharic). P1 nods. P1 tries two more sentences with modal verbs. Now he wants to use the German word order immediately (writing the object before the second verb was defined.). This is not supported by the system. The interviewer notices that P1 is confused and explains the word order used for complex verb constructions by *EasyTalk*. P1 nods and does not comment further.

- **O11:** Except for two sentences, P1 began all sentences with personal pronouns. The interviewer asks if he would prefer the pronouns to be at the top of the word list. P1 says: “Yes, and then the articles.
- **O12:** The interviewer notices that P1 is confused by the questions ‘Whom_ *acc*?’ and ‘Whom_ *dat*’. The interviewer asks if he knows the difference. P1 answers that he knows but cannot always tell which one to use.
- **O13:** In one sentence, P1 uses the adverb *gestern* ‘yesterday’. *EasyTalk* automatically puts the sentence into the present perfect tense. *EasyTalk* notes that this is useful for learning. However, he would prefer to be able to write directly in the present perfect tense.
- **O14:** The interviewer notes, that P1 did neither use nor ask about the colon.
- **O15:** P1 requests to be able to delete a word form or sentence component without having to start the sentence over.

A.2.2 Participant P2

Profile: P2 is a 37-year-old male. His mother tongues are Cotocoli and French. His German skills are at CEFR level A1. His computer skills are low.

Q1: Was the system easy or hard to use for you?

- Yes, so easy

Q2: Was the sentence construction process easy or hard to understand for you?

- It feels like a game.

Q3: Do you feel supported in increasing your grammatical skills in German by using the system?

- It is like a learning system. Like training for the driver’s license on the computer.
- It is good that you cannot make mistakes.

- That you cannot make a mistake it is not good. You need to make the mistake to learn. Then the program needs to tell you the mistake so you learn.

Q4: What do you think about the symbols?

- It's good to see images with words.

Additional observations from the test session in chronological order:

- **O1:** The session starts with an example sentence written by the interviewer: *Du antwortest der Frau.* 'You answer the woman'. Then, the interviewer prompts P2 to try writing a sentence. P2 asks *Was für ein Satz?* 'What type of sentence?'. The interviewer answers in English: "I give you the book". P2 starts writing the sentence in German. After writing *Ich gebe dir das* 'I give you the', P2 clicks the arrow button and finishes the sentence.
- **O2:** P2 needs support from the interviewer to write the sentences. The interviewer reads the wh-question cues to P2 to guide him through the sentence. This way, P2 writes one component after the other.
- **O3:** P2 types *gibe*. The suggestion list does not show any words. The interviewer asks which word P2 is looking for. P2 answers that, again, he wants to write *Ich gebe dir das Buch.* 'Ich gebe dir das Buch.' The interviewer helps him with the spelling and points out to look at the images to find the word in the list. P2 finds the word and says: "Ah, the images are a good help."
- **O4:** P2 wants to write a question. The interviewer says that this feature is still in development. P2 nods.
- **O5:** About 15 minutes into the session, P2 asks where to download the system to use it for practicing his German.
- **O6:** P2 praises the read-aloud function, which he used several times.
- **O7:** P2 does neither use nor comment on the colon.
- **O8:** Several times, the word list is empty due to P2 having entered the wrong article for the noun that he wants to use as the object.
- **O9:** The interviewer shows P2 the automatic transformation of the sentence from present to perfect present tense when entering the word *gestern* 'yesterday'. The interviewer asks P2 what he thinks about the feature. P2 says: "Oh, a, egal" 'Oh, uh, does not matter.'

- **O10:** P2 read aloud all conjunctions in the Connector Panel. He says he only knows *und* ‘and’ and *oder* ‘or’.

A.2.3 Participant P3

Profile: P3 is a 24-year-old male. His mother tongue is Tigrinya. His German skills are at CEFR level A2. His computer skills are rudimentary.

Q1: Was the system easy or hard to use for you?

- Average. With practice, it is easy.
- The voice (from the read-aloud function) is a little bit fast if you cannot understand (German) well.

Q2: Was the sentence construction process easy or hard to understand for you?

- I think it is a great idea like this. If you cannot speak German well, you should use this method.
- It is good that it is always correct. But it is not so important at the beginning when speaking German.

Q3: Do you feel supported in increasing your grammatical skills in German by using the system?

- Yes. You can be very creative.

Q4: What do you think about the symbols?

- You can guess what the word means. Of course, the images are good for that.
- The biggest advantage (of the system) is: The word and the image are (shown) together. If you cannot read, you can know anyway what the meaning is.
- Images should be everywhere in the system.

Additional observations from the test session in chronological order:

- **O1:** The session starts with the example sentence *Ich antworte dir* ‘I answer you’ proposed by P3. P3 then asks if the interviewer could assist him with the next sentence as well. He says that he doesn’t understand yet, how the system works. As the second sentence, he proposes *Du sagst mir das Wort.* ‘You tell me the word.’. After writing the second sentence together with the interviewer, P3 says that he now understands how the system works.

- **O2:** After writing two sample sentences together with the interviewer, P3 takes some time to answer different questions about the system, like “What is this button for?”, “What happens if I click here?”, “How can I choose a word?”. Then he says: *OK, ich habe verstanden.* ‘OK, I understand now.’
- **O3:** P3 only types the first letters of a word before selecting the corresponding word form from the word list. The interviewer asks P3 what he is looking at more in the suggestion list – the words or the images. P3 answers: “Both.”
- **O4:** P3 is confused by the “Play-Button” used to transition to the next sentence component. He asks why he cannot immediately type the next word without having to operate the play button.
- **O5:** The interviewer asks P3 to try to write a more complex sentence, like *Der Hund auf dem Auto schläft.* ‘The dog on the car sleeps.’ P3 asks if this is the correct word order. He would prefer *Der Hund schläft auf dem Auto.* ‘The dog sleeps on the car.’
- **O6:** P3 says: *Konjugation ist gut. Ist sehr gut zum Lernen und einfach zu verstehen* ‘Conjugation is good. It’s very good for learning and easy to understand.’
- **O7:** P3 says: *Bild und Wort ist immer gut.* ‘Image and word is always good.’
- **O8:** The interviewer shows by example, that *EasyTalk* automatically puts sentences into the present perfect tense, if the word ‘yesterday’ is added to the sentence. P3 says that it is good for learning but that he would not have noticed by himself.

A.3 L2 learners with expert computer skills

Here, we present observations and feedback from the test sessions with the L2 learners of German with expert computer skills and varying literacy skills in the German language – P1-P10 – evaluated in Section 6.3. All interviews were conducted in English. The notes were taken from the video and audio recordings taken during the remote testing sessions using *Microsoft Teams*¹.

¹*Microsoft Teams*: www.microsoft.com/microsoft-teams

A.3.1 Participant P1

Profile P1 is male and 31-35 years old. His mother tongue is Romanian. He is a professional software developer and learned German by teaching himself. He also used the app *DuoLingo* to practice German. His German skills are at CEFR level ~B1.

Observations and feedback

- P1 seems to be familiar with conjunctions and chooses them without second guessing.
- P1 comments on the system: “It’s really nice.”
- Writing the third sentence, the tester tries to enter two words at the same time. He then realizes that he needs to enter each word separately and comments on this with, “Oh, just one at a time.”
- When writing *zieht an* ‘puts on’, the tester comments “Ah, but this is different – but it’s fine right?”. Then, P1 chooses *zieht an* ‘puts on’ together with the particle. He then asks, “Why is it two words?”. The interviewer explains, “It’s a separated word.” P1 says: “Aaah, right.”
- Choosing the third connector, P1 says to the interviewer: “I’d like you to comment if I am choosing the right connectors, so maybe I can improve my German.”
- P1 comments after finishing the fifth sentence: “Ah, interesting. I am not exactly sure why I choose ‘ski’ after ‘fahren’”.
After the test, the interviewer explains that the second verb needs to be chosen first in order to have the correct verb valency. (‘will’ is connected to ‘fahren’ which is connected to ‘ski’.) The interviewer asks what P1 thinks about this mechanism. P1 voices that he cannot imagine the mechanism working differently since the user must choose between an action and an object.
- P1 proposes to let the user make mistakes and then highlight mistakes to create a better learning effect.
- After the session, P1 comments that the system was very enjoyable and that he likes how the system works.
- P1 says, except for ‘sentence 5’ everything was very easy to understand. Only the complex verb clusters were confusing to him, since he did not know the mechanism before using it in the sentence.

- P1 says that he would like to use the system more in order to give more detailed and better feedback.
- P1 says the system is very supportive due to the auto-complete function and the images.
- The images appear very friendly to P1.
- P1 says the UI is not leading edge.
- P1 comments on the system: “It kind of held my hand while typing and writing, this was very motivating.”
- P1 comments on the understandability of the system: “It was pretty clear, but slightly confusing for the one part.” (referring to complex verb clusters)
- P1 proposes to have a deeper introduction and maybe tutorial videos or tooltips on how to use the system.
- P1 comments on the UI of the system: “It definitely was not cluttered at all. You don’t want to put in too much stuff so it doesn’t get confusing.”
- The interviewer asks P1: “Would you recommend the system to German learners at beginner level?”
P1: “I don’t think so. Not because the program is bad or anything, but because there’s stuff as *DuoLingo* or *Google Translate*, which is much easier, I think. Because here you have to know what you want to say and you have to have at least a start and then you build it up. And also here you cannot make mistakes. You build it up and it’s always a more or less good sentence. So I think you learn more if you can make mistakes and you have to correct it. It’s just like an auto-complete. That’s all you need to remember to succeed. If you’re learning for real, you may make a mistake on the last letter, and you have to learn all of them in the end.”

A.3.2 Participant P2

Profile P2 is male and 25-30 years old. His mother tongue is Arabic. He is a professional software developer and learned German by taking German classes and living in Germany. His German skills are at CEFR level ~B1.

Observations and feedback

- P2 asks: “If I did something wrong [when connecting the sentences], will I give me hints or tell me?”
- P2 comments: “It would be perfect to have two modes – one where you write correctly, and one that tells you your mistakes, and you have to correct them.”
- “Comparing it (the system) to *DuoLingo*, it would be nice to also have levels. So, for example, as a beginner, I can focus on one thing, maybe first getting started and being corrected and then try myself but be shown my mistakes to be aware of them.”
- The interviewer asks: “Do you think the sentence connection display motivates you to choose connectors”?
P2: “I’d say the symbols are sometimes more confusing than helping me. I’d say the choice of symbols for the connectors can be enhanced. Also, it would be perfect to have a tooltip or explanation for the connectors to show you, what their meaning is -- e.g., ‘aber’ is contradiction. It would be good to have the grammatical function. Even if I don’t understand the word or symbol, I can learn the function of the connectors.”
- The interviewer asks: “Would you recommend it (the system) to people who start learning German?”
P2: “Yes, for sure. It’s a bit like having a picture dictionary but through a computer program and to help you write. It’s also helping with the photos. You immediately understand what a word means from the picture, which is nice.”
- P2: “It’s a good thing, that the system is no predicting which words to show me as it is not boring me then. I can write what I want. The system should not predict my choices too much for learning.”
- P2: “It would be much more better for learning as an app on the smart-phone.”
- P2: “It feels like a fun learning app, but not like a game. It’s good. If I have a motive to learn something about the language, I would use it.”

A.3.3 Participant P3

Profile P3 is female and 35-40 years old. Her mother tongue is Spanish. She is a professional (IT) solution consultant and learned German living in Germany. Her German skills are at CEFR level ~A2.

Observations and feedback

- P3 repeatedly makes typing errors when selecting words from the completion list, which cause the completion list to be empty. At first, this throws her off and, she asks: “Hm, is it doing something or what’s going on?” The interviewer hints at her spelling mistake, and she says: “Aaah.”. After this incident, she looks for spelling mistakes when the completion list is unexpectedly empty and then corrects her mistakes herself by looking at the vocabulary given for the exercise.
- P3 voices that the symbols help her to recognize the words.
- P3 thinks about which connector to choose but is unsure if her choice is correct.
- P3 voices that she chooses the “arrows” to in the Connector Panel to continue without a connector because she feels like a new action begins between the third and the fourth sentence.
- P3 appreciates that the verb and its separated prefix, *zieht an* ‘puts on’, are presented together and that the object is then moved in between the verb and its particle.
- P3 says: “I really like the icons, though. They are cute.”
- In the fifth sentence, the interviewer explains the decision dialog (“Wen” vs “Was tun”) to P3 as she pauses when it occurs. She says “Ah, ok, I choose an action because it’s a movement.”
- Being asked about the reversed word order (second verb before object), P3 says it does not seem unnatural to her, as Spanish has the same word order.
- P3: “I think this can definitely help people learn.”
- P3: “I think it was easy to learn.”
- P3: “I think it was funny.” (as in “fun to use”)
- P3: “I think it can be useful to people.”

- P3: “I think after an example sentence it is easy to use.”
- P3: “I think it was inventive. I have not seen any tool alike. I have not used too many learning apps, but I did not see one like this.”
- P3: “I think the icons help to see what to construct as a sentence.”
- P3: “I think complicated or easy depends. For me it was fairly easy, but I think it depends on the skills of the user.”
- P3: “I think it can help to improve my German, as it reminds me of the grammar and the word order in a German sentence.”
- P3: “It’s organized, you have an order in your sentence and it helps you build a story.”
- P3: “The connection panel helps you to connect or to separate your sentence.”
- The interviewer asks: “Would you recommend it/like to use it?”
P3: “Yes, I think so. The icons help a lot. It helps you visually to see what you are writing. Also, the questions in the labels help you to learn how to write a sentence in a correct order. Also, it helps you to learn the grammar without having to know the Dative and Accusative right away.”

A.3.4 Participant P4

Profile P4 is male and 41-45 years old. His mother tongue is Swedish. He is a professional (IT) business consultant and learned German in school classes for six years and by living in Germany for six months. His German skills are at CEFR level A2-B1. P4 spoke German and English during the test session.

Observations and feedback

- After the example sentence has been demonstrated by the interviewer, P4 asks: “In the Wörterbuch you have written ‘Wem’, ‘Wen’, ‘Wo’, ‘Wann’, ‘Wie’ – is that helping how to create the sentence?”
The interviewer clarifies: “It’s for helping to create the sentence.”
P4: “Ah, Ok, so ‘wann’ would be time-related things, I see.”
- P4 uses the wh-labels to filter the dictionary while constructing the sentence.
- P4 tries to expand the dictionary. “Can I not expand the Wörterbuch? Es würde schön, wenn ich mehrere Alternative sehen könnte.”

- P4: “Es schneit. Ähhh, Punkt.” P4 hesitates at the end of the first sentence as he searches for the ‘Punkt’ (punctuation mark) to end the sentence. The interviewer explains: “When you’re done, click the checkmark.” P4 comments: “Ahh, ich sehe. Und dann kommt der Punkt automatisch und dann ich würde weitermachen.”
- P4 comments after finishing sentence 2: “Ah, hier gibt’s ein Punkte. Ja, genau.”
- P4 takes a moment to read through all conjunctions before choosing ‘und’ to connect the second and the third sentence.
- Writing the third sentence, P4 comments, “Ah, Ok, das ist gut.” as the system moves the object between the verb *zieht* ‘put’ and the particle *an* ‘on’. He adds *seine* ‘his’ and *Jacke* ‘jacket’ and then says: “Ohh, wunderbar, er hat an die Ende geschuckt geworden, das ist gut.”
- P4 comments while constructing the third sentence: “Er zieht seine ‚Jacke‘ nicht ‚wen‘ – aber ‚was‘”
- Writing the fourth sentence, P4 is first confused why *fahren* ‘drive’ has to be selected before *Ski* ‘ski’. Nevertheless he chooses ‘Was tun’ instead of ‘Wen’ correctly in the decision dialog.
- P4 comments: “Ah, es ist nicht viele Ski, es ist eine Ski.” on the indication of the number through the 1,+ symbols next to the nouns in the suggestion list of the Next-Word Panel.
- Referring to the construction of complex verb clusters, P4 asks: “Why do I have to choose here? Why can the sentence constructor not let me choose as I go and then build the sentence in the right Ordnung?”
- P4: “The ‘wen’, ‘wo’, ‘wohin’-suggestions at the bottom, I think those could be expanded a bit to give suggestions – for example, after writing ‘er will ski’ it could give suggestions like ‘fahren’ or ‘kaufen’ or other words that fit the context.”
- P4: “Creative – I would like to write my own sentences. It’s good that I can write what I want.”
- P4: “I think the structure was good with the sentences at the top, the construction in the middle and the Wörterbuch at the bottom.”
- P4: “I didn’t quite understand the suggestions, so that part was a bit unpredictable.”

- P4: “It was fairly easy.”
- P4: “The graphical style was a bit on the plain side, I would say, but you shouldn’t judge it on the surface of things.”
- P4: “I think this could be valuable for a person learning German.”
- The interviewer asks: “Would you recommend it to someone learning the language?”
P4: “Yes, but I would have expected there to be some more suggestions on how to complete a sentence vocabulary wise.”
- P4: “I didn’t know how big the vocabulary was.”
- P4: “Language is also about formulating your thought. Helping the users to get to the right expressions would be valuable. Maybe I wouldn’t know the word Ski – I would know there’s something to put on my feet, but I don’t know the word for it. Maybe if I start typing winter, I could get some suggestions in that direction. Both nouns as well as verbs and directions.”
- The interviewer asks: “Any thoughts on the Connector Panel?”
P4: “If there was a story and one thing led to another and there was a causal connection between those things, this would be very valuable.”
- P4: “I think your test should have included more free sentences, like write a story of what happened yesterday.”
- P4: “I think there could be different vocabularies for different themes. Like ‘daily life in my house’ and then maybe you get suggestions for words and expressions of that topic.”

A.3.5 Participant P5

Profile P5 is male and 41-45 years old. His mother tongue is Swedish. He is a professional (IT) solution architect and learned German by teaching himself. His German skills are at CEFR level A1-A2.

Observations and feedback

- P5 speaks several languages (Swedish, English, French, Spanish, German) and teaches French, English, and Spanish on occasion.
- P5: “Do I click once or twice? Ah, ok, once.”

- P5: “Ah, so it clears out the Wörterbuch when I change the filters. Ok, that makes sense.”
- P5: “Do I always choose the word by typing a part of it and then clicking or can I select the first word directly by pressing Enter? That would speed it up even more.”
- Writing the fourth sentence, P5 voices when choosing the verb: “I am thinking about one thing here, when I write ‘zieht’ I get ‘zieht an’ together, right? That’s interesting, because then I get the full word with the particle. ‘er zieht an’ und dann nehme ich ‘seine Jacke’.”
- P5: “If I didn’t know that ‘ziehen’ is spelled with a ‘z’ would I get suggestions? I mean based on the sound of the letters I wrote. That could be useful if I know some words but cannot spell them yet. Like if I wrote ‘ts’ would it suggest ‘z’ because it sounds ‘ts’? So could it help you out by recommending words that start with a similar sound?”
- On sentence 4, P5 comments: “Ah, it moves the object in between the verb and the particle, so it understands that. That’s a nice support.”
- On sentence 4, P5 further comments: “Ah, so I could put an ‘und’ between objects? Like ‘seine Jacke und seine Mütze’ Ah, I see. That’s good. So I can write more complex objects also. “ (while thinking out loud the P5 types ‘und’ into the completion list and comments, that he found the ‘und’.
- P5 changed the fifth sentence from “Er will Ski fahren” to “Er will nach Zoo fahren.” The system is not able spot the mistake of writing “nach Zoo” instead of “in den Zoo”, so P5 thinks his sentence is correct.
- P5: “Ah, no it asks for a decision. Hm, ‘wen’ oder ‘was tun’. Ah, like ,Ich will eine Pizza. Ah, in Swedish we would add a little ‘haben’. So well yes, in a way you always have to add the second verb directly after the first one. So this teaches you the German word order.”
- P5 types ‘faren’ instead of ‘fahren’ and then comments: “Ah, it needs to have the h, there’s no correction or suggestion. Ok.”
- Interviewer: “Would you allow typos?”
P5: “Maybe it would be interesting to have two columns, one with words you’re heading straight at and one with ‘ah, maybe you meant this?’ Especially if you have the cases if one letter is almost not pronounced or the case we had before with ‘ts’ and ‘z’”

- P5: “So in that case, if I would choose ‘fahren’, do I have to go with ‘Ski’ or could I also choose ‘nach München’?”
 I: “Yes, you could also choose a direction or add a time, if you like.” P5: “And that’s what I choose here, right?” (Tester refers to the wh-labels above the dictionary)
 I: “You have two options, you can either select a filter or you can type ahead and the system will recognize the component. If there are several, it will ask you to make a choice.”
 P5: “Ah, so here I could go ‘nach’” (Types in the filter input of the completion list) “And then it would say ‘what type of *nach* do you want to go for’?”
- P5: “Ah, ok right. The funny thing is, I would type in the order ‘Er will fahren nach München’ and the system would put it to ‘Er will nach München fahren.’ The funny thing is that the word order I would enter, is the normal Swedish word order. So the system translates Swedish words in Swedish word order into German words with German word order. That’s useful to learn the correct word order.”
- P5: “So now I select *nach* then I have different choices. So, ‘Wohin/woher’ would be ‘Where to, right’? So I could go in a place to a play. For instance, in South America to Peru. Ok, got it.” (As the system presents ‘Wie’, ‘Wann’, ‘Wo’ and ‘Wohin/Woher’ in the decision dialog)
- “So then I could go this. OK, good.” (The tester selects ‘Zoo’).
- Interviewer: “Do you think it’s confusing to choose the second verb before the first one?”
 P5: “One way would be to say ‘Ok, you’re typing ahead, but you’re still missing something. It’s lacking something. We’ll remind you later, but we let you go first. Hm, I’m not sure. If you’re writing something in your native language you could forget something. Both Swedish and German have that reverse word order in some cases. In German you put the infinitive at the end. In Swedish we switch the word order for negations, for example. Or in a subordinate clause, for example. But it’s the sort of mistake you expect immigrants to make, because in many languages the word order is the other way round. So if you hear someone speak with an accent but in that word order, you go ‘ah, you probably learned the language for a while’ because you already picked up on that exception.”
- P5: “Sometimes the native language of someone doesn’t have definitive articles. And when they learn how to use them, they put them in the strangest

places.”

Interviewer: “Interesting point. This could actually be an obstacle for choosing words in this system, because if you chose the article in the wrong gender, you won’t be presented the noun you are looking for in the next step.”

P5: “I see, yes, that’s an interesting point to cover.”

- P5: “It sort of forces you to write in structure, but at the same time you can write what you want.”

A.3.6 Participant P6

Profile P6 is male and 45-50 years old. His mother tongue is Swedish. He is a professional software developer and learned German in school for 3 years and from occasionally talking in German in his work environment. His German skills are at CEFR level ~A2. During the test session P6 spoke English and German.

Observations and feedback

- P6: “Ah, there’s no exclamation mark. That’s missing.”
- P6: “So it’s kind of a flow chart, so I would connect [sentence 1+2] with a ‘und’”.
- Writing the third sentence, p6 comments: “So *sich* is kind of an object I guess.”
The interviewer adds: “Yes, above the dictionary it indicates the object with ‘wen’”
P6: “Ah, OK. I missed that. Ahh, ‘Wen’ natürlich.”
- P6: “It’s very nice to have to icons next to the words.”
- “Ah, ‘zieht an’” P6 comments when choosing the verb for the fourth sentence. Continuing, he comments further when he adds the object: “Ah, the *an* is moving. That’s really cool. I like that. ‘Anziehen’ I guess is the word. So this is really useful.”
- Connecting the fourth and the fifth sentence, P6 reads through all connectors and says: “Ah, hm, 2 Punkte, I guess.” And chooses the colon to connect his sentences.
- P6 chooses “Was tun” in the decision dialog in the fifth sentence. Then he wants to continue with ‘ski’ and the interviewer clarifies that he needs to choose the second verb first. P6 comments: “Ah, so I choose that I want to go and then what I want to go with. Yes. Makes sense.”

- The interviewer says: “You mentioned, that you think it’s nice that it moves the particle the right way. With the 2 verbs in sentence 5 it’s the same way – how did you feel about that?”
P6: “I think it was OK, but I think I was thinking of ‘ski fahren’ as one concept or maybe even one word written together.”
Interviewer: “Would you like to have different suggestions after choosing ‘fahren’? Like ‘Auto fahren’, ‘Ski fahren’?”
P6: “Yes, at least that’s how I feel.”
- P6: “I think it was very understandable. I got thrown off by the ‘fahren’, but then it was fine. I assume when you’ve done a few short stories like these, it would be ok.”
- P6: “Can you write anything you want?”
Interviewer: “Yes, well within the grammar rules of Easy-to-Read German.”
P6: “Nice.”
- P6: “Yes, I think it is very efficient. When you’ve done it a few times, it will be.”
- P6: “It’s very nice and friendly with the images.”
- Interviewer: “Denkst du, Schülern würde das System helfen zu lernen?”
P6: “Ja, ich denke besonders die Konzept mit die trennbaren Verben und die Positionen im Satz.”
- Interviewer: „Would you like to add anything?”
P6: “The exclamation mark, for example.”
- Interviewer: “How did you feel about choosing the connectors?”
P6: “It made me think. I guess I spent a lot of energy on it.”
- Interviewer: “Would you recommend it (the system) for someone learning the language?”
P6: “Yes, it’s nice. From a configurator standpoint of view – does it create correct sentence at all time or does it only help you to avoid the most common mistakes?”
Interviewer: “It’s designed to always ensure grammatical correctness.”
P6: “Ah, that’s cool.”
Interviewer: “Do you think that’s a benefit or would you prefer if it hinted mistakes for correction?”
P6: “No, actually I think that’s really useful. If I want to make a sentence, but I don’t know how to do it. Or if I want to use a Swedish structure, that’s

not valid in German, then I have to think about it and how to continue. Also it gives you hints with the ‘wen’, ‘wann’, ‘wo’. I think that’s really nice. I think it’s really cool that it creates correct sentences.”

- Interviewer: “You mentioned it feels like a flow chart – how come?”
P6: “Maybe because it makes you put the sentences into a logical order.”

A.3.7 Participant P7

Profile P7 is male and 30-35 years old. His mother tongue is Swedish. He is a professional IT project manager and learned German by studying in Germany for one semester and by playing around with the app *DuoLingo*. His German skills are at CEFR level A2-B1.

Observations and feedback

- P7 chose to write his own sentences after the example sentence ‘Es schneit’:
Peter schaut aus dem Fenster:
Es schneit.
Und
Alles ist sehr kalt.
Er zieht seine Jacke an.
Aber
Die Jacke ist klein.
Er zieht seine Jacke abends an.
- P7: “Do I click this one now?” (after choosing the first verb) Interviewer: “The checkmark you click at the end of a sentence.” P7: “Ah, right, ok. So I just type.”
- Writing “*Alles ist*” ‘everything is’ evokes the decision dialog. P7 reads the decisions and goes for *Gegenwart* ‘present tense’, while commenting his choice with “present tense”.
- After writing *ist* ‘is’ P7 voices: “Ah, and then I get to choose between those words, right?” (Commenting on the wh-questions)
Interviewer: “Exactly. You can use them to filter the dictionary. If you don’t apply a filter, the system will automatically categorize the word form.”
P7: “Hmmm. OK. So then if I start writing something. Ah, nice.” P7 comments as he writes *sehr* ‘very’ and it is recognized as a ‘wie’-component.

- Referring to the icon of *sehr* ‘very’ P7 asks: “Ah, what does this mean, the icon? Is it to describe the word?”
Interviewer: “Exactly.”
- P7 on reading through the suggestion list: “What is the difference between ‘kalt’ and ‘kalte’”.
Interviewer: “It’s a different inflection – for example: ‘mir ist kalt’ or ‘ich will kalte Süßigkeiten essen’.”
P7: “Ah, Ok ok.”
- “Can I go to the next sentence directly clicking this one?” P7 asks hovering over the arrow buttons with the mouse.
Interviewer: “Yes.” P7: “OK.”
- T: “Ah, it proposes ‘zieht an’ together. But if I want to write ‘zieht seine Jacke an’ does it move in between?”
Interviewer: “Try it.”
P7, on the object moving between the verb and its particle: “Ah, nice. That’s good.”
- Observation: P7 doesn’t hesitate to choose conjunctions. It seems like he already has the story he wants to write in his mind and specifically goes for the connectors he wants to choose.
- P7 invokes several decision dialogues. He navigates them without hesitation.
- P7 wants to write ‘Die Jacke ist zu klein’. However, ‘zu klein’ is not covered by the grammar of *EasyTalk*. The interviewer clarifies this to P7. P7 then goes for ‘Die Jacke ist klein’.
- P7: “I think it was quite fun.”
- P7: “I think the usability was good. The problem for me was just the [wh-]questions. It would be good to have some help text for those. Or you know, when you start a software for the first time and you get explanations for the components. That would be good. There could be an explanation not only for the structure, but for the steps.”
- Interviewer: “As a second language learner, would you like the explanation in Swedish or English? What would be your preference?”
P7: “Swedish, in this case. English would be OK, but the native language is of course easier.”

- P7: “It would also be good to have some tooltips on the grammatical function of the connectors, like ‘aber’ is a contradiction, etc.”
- P7: “The graphical interface could be improved. It makes it appear very usual.”
- P7: “The sizing of the buttons, font and images could also be adapted, depending on what’s more important. For example, the arrow button in the panel of the connectors is bigger, but not more important than the others.”
- P7: “I would recommend to use it. I played around with *DuoLingo*. Which is also good, but is missing one aspect as it does show you mistakes, but no correction. I think this has the power to give hints about mistakes and why the grammar would not be correct in a case. A combination of that would be awesome.”

A.3.8 Participant P8

Profile P8 is male and 45-50 years old. His mother tongue is Swedish. He is a professional IT solution architect and learned German in school for five years. His German skills are at CEFR level ~A2.

Observations and feedback

- P8 asks if Easy-To-Read German is the same as Simplified German.
- Watching the interviewer writing the sample sentence, P8 asks: “The small green boxes are indicating the grammar context, right?”
- P8 says that he thinks the wh-questions communicating the grammar functions are probably easier to use for native German speakers than for second language learners, especially “Wen” or “Wem”.
- P8 would like to see more words in the completion list.
- P8 suggests a tag cloud for vocabulary choices.
- Writing the third sentence, P8 asks how the wh-labels above the dictionary can be used. The interviewer explains that they can be used to filter the completion list by grammatical function. P8 says that he thinks this feature is very helpful.

- Writing “er zieht an” P8 asks: “So, if I don’t know, that ‘zieht an’ is spelled with a z? What would happen – for example, if I start with an ‘s’?” The interviewer explains that then it would not appear in the completion list. P8 says: “Ah, it could also filter by the sound-likeness because it could assume that the user may be misspelling it.”
- Adding ‘seine Jacke’ to the sentence, P8 says: “Ah, Jacke would be only one.” referring to the number indicator (1,+) next to the symbol in the completion list.
- P8: “So in this case, the grammar function for ‘Jacke’ and ‘Jacken’ would be the same, right?”
- P8, on using the Connector Panel: “Hm, should I join this? Yes, I think I want to join it with the next sentence. But I’m not sure which connector to choose – ‘weil’ is a cause, and ‘darum’ – hm. I will just go for ‘und’.”
- P8: “I think tooltips explaining the grammar function would be helpful for second language learners.”
- P8: “Can I select directly by pressing ENTER?”
Interviewer: ‘You have to go down once with the arrow keys, then yes.’
P8: “Ah, OK, choosing directly could speed it up.”
- Regarding the complex verb cluster in the fifth sentence, P8 says: “The problem I will have, of course, is that it will not be complete until I fill out all the verbs. However, the language process is not in the German order. If I would think it my way, or in English, it would be ‘I will go skiing.’ So then it would work, but it doesn’t fit the German word order.”

A.3.9 Participant P9

Profile P9 is female and 25-30 years old. Her mother tongue is Mandarin. She is a professional product modeler for configure price quote knowledge bases and learned German during a student exchange for eight months and then by living and studying in Germany. Her German skills are at CEFR level ~B2. During the test session, she mostly speaks German.

Observations and feedback

- P9 asks how second language users that are not familiar with choosing the correct determiner for a word will be able to navigate the vocabulary (because if they start with the wrong article, e.g., ‘die’ (Gender: female) they will not find the word they expected (e.g., ‘Problem’ (Gender: neuter)).

- P9 wanted to type 'zieht' without the prefix 'an'. Then, she says: "Ah, das Programm fügt 'an' automatisch dazu. Ahhh".
- „Ah das ist eine Jacke und mehrere Jacke. Ah, Ok." The tester notes choosing between 'Jacke' and 'Jacken'.
- P9: "Es ist hilfreich, wenn man nicht so gut schreiben kann, dass ,an' und ,aus' und so automatisch mit dem Verb kommt."
- P9 wanted to write "skifahren" as one word, resulting in an empty completion list.
- Writing complex verb clusters, P9 is not irritated by the fact that the second verb has to be chosen before an object.
- P9 asks if the symbols can be exchanged/adapted for the user.
- P9 asks if the user can choose a second mode, that will point out mistakes to correct them instead of always writing correctly.
- P9 says that she thinks for children the symbols are very useful. For adults she isn't sure if they will be put off by the symbols.
- P9 asks: "Wie ist es mit dem der/die/das?"

A.3.10 Participant P10

Profile P10 is male and 31-35 years old. His mother tongue is Swedish. He is a professional IT business consultant and learned German by living in Germany for six months. His German skills are at CEFR level ~A1.

Observations and feedback

- P10 writes his own text:
Peter schaut aus dem Fenster:
Und
Es hat gesehen:
Es schneit abends.
Er freut sich.
Und
Er will seinen Freund anrufen.
- P10 has problems choosing the vocabulary, as he can only speak a bit of German and does not have a lot of spelling skills. This often results in an empty completion list. The interviewer assists with spelling.

- P10 wants to write “dass”, which is not offered in *EasyTalk*.
- P10 appreciates that he can write whatever he wants and *EasyTalk* has no limitations for grammar support. He compares *EasyTalk* to *DuoLingo*: in *EasyTalk* the user can write freely, while in *DuoLingo* given sentences are practiced.
- P10 likes the symbols and thinks that they are very helpful
- P10 would like to have more assistance for the vocabulary, e.g., by offering word sets for different domains.
- P10 says he would recommend the system to second language learners, as he thinks it’s motivating the user to write correctly and to write coherent texts, instead of uttering only single phrases.
- P10 uses the images to interpret the meaning of the words in the completion list.
- P10 would like to see more words in the completion list.
- P10 appreciates that the system moves words to their correct spot (e.g., ‘er will anrufen seinen Freund’ is transformed to ‘er will seinen Freund anrufen’) and suggests a sentence order by providing the wh-labels.
- P10 uses the wh-labels to filter the completion list and to find vocabulary suggestions (‘Es schneit’ – “Wann” – ‘Es schneit abends.’)
- P10 navigates the user decision dialogues without problems. He chooses ‘Vergangenheit’ to filter for participles to write ‘hat gesehen’.
- P10 would like to have tooltips providing translations and explanations regarding the grammatical function for the sentence connectors and for the wh-labels.
- P10 says the program is annoying him because it makes him want to be better at German. He clarifies that it is ‘annoying in a good way’.

A.4 Participants with IDD and/or CCN

In this section, we present observations and feedback and eye-tracking data from the test sessions with members of the target group – evaluated in Section 6.4. All sessions were conducted in German. The notes translated into English for their presentation in this work. At the end of the section, we present some additional eye-tracking data (Section A.4.9).

A.4.1 Participant P1

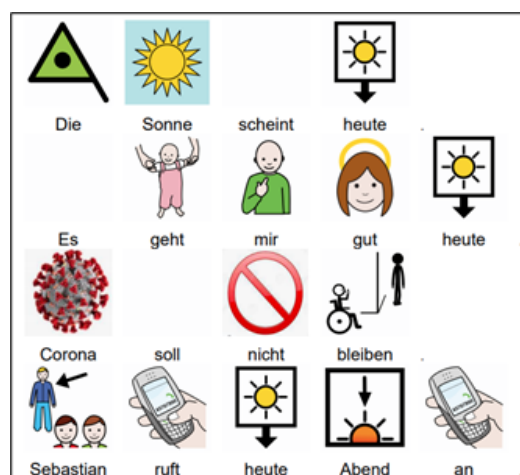


Figure A.1: Sentences written by P1 in the test session. Translation: S1: The sun shines today. S2: I am doing good today. S3: Corona should not stay. S4: Sebastian will call me tonight.

Profile P1 is male, 20-25 years old, and has autism spectrum disorder. His mother tongue is German. P1 has spontaneous decision-making problems. Together with his caretaker, P1 thought up his sentences prior to the writing session. His personal caretaker was present during the test session. P1 autonomously types on the keyboard, using his right index finger only. He looks for each letter individually but appears to know where the letters are located on the keyboard. He operates the mouse by himself. P1 agreed to the use of eye tracking.

Observations

- The interviewer writes the example sentence together with P1: *Die Sonne scheint heute*. ‘The sun shines today.’
- P1 wants to write as the first sentence *Mir geht es gut*. ‘I am good.’ However, inversion (in this case, starting the sentence with the dative object instead of the subject) is not supported by *EasyTalk*. The caretaker advises P1 to write *Es geht mir gut* ‘I am good.’ instead.
- On his first try, P1 types the entire sentence into the text input field in the Next-Word Panel. The interviewer reminds P1 that each word must be entered individually. P1 does not comment but enters word by word from now on.

- P1 uses the wh-cues before selecting the time modifier *heute* ‘today’.
- Without prior notice, P1’s supervisor repeatedly used wh-questions as a concentration aid for P1 when formulating sentences to remind P1 of what part of the sentence he was currently writing.
- P1 wrote each word from start to finish in the text input field in the Next-Word Panel. The interviewer pointed out that P1 can select the words from the suggestion list without typing them to the end. P1 responded that he prefers his method of “*fertig schreiben*” ‘writing to the end’.
- P1 occasionally received help with spelling from his caretaker and the interviewer.
- P1 wanted to type “*Hoffentlich ist Corona bald vorbei*. ‘Hopefully, Corona is over soon.’ Because *EasyTalk* does not support starting a sentence with an adverb, P1 had to rephrase his sentence to start with the subject.
- Several times, P1 added a trailing white space after entering a word in the Next-Word Panel. When finishing typing, P1 looked up from the keyboard to the laptop screen. The suggestion list was empty now. This confused P1. The interviewer assisted P1 and removed the trailing white space.
- P1 wanted to use his brother’s name in his text. However, the name had to be added to the dictionary first.
- Three times, P1 wanted to start his sentence with an adverb (*heute* ‘today’, *nachher* ‘later’, *hoffentlich* ‘hopefully’). Because *EasyTalk* does not support this word order, the caretaker reminded P1 to start his sentences with the subject.
- Several times, P1 wanted to use words that were not included in the vocabulary, like *aufhören* ‘to come to an end’.

Eye-tracking data Figure A.2 shows P1’s eye-tracking calibration values (7.10° Accuracy, 3.12° Precision (S), 0.80° Precision (RMS)). The calibration image indicates that P1 has a tendency to look downwards, away from the screen and onto the keyboard – as we noted in the observations above. This is also reflected in some of the gaze plots in Figure A.3 and Figure A.4 that show gaze paths running out the bottom edge of the image.

Figure A.3 visualizes the eye-tracking data of the sentence-connection intervals C1-C3. The gaze plots (a), (c), and (e) show that P1 focuses on the finished sentences in the Text Panel before reading through the connector options. Hereby

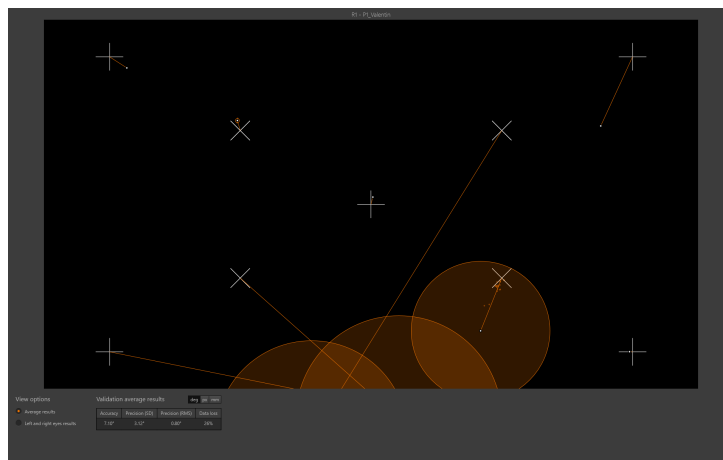


Figure A.2: P1's eye-tracking calibration-data exported from *Tobii Pro Lab*.

P1 focuses most on the sentence written last (the bottom sentence in the list), as the focus moves one line down the list with the increasing text length in C1, C2, and C3. The heat maps (b), (d), and (f) in Figure A.3 indicate that P1 focuses the top row of sentence connectors more than the second row of connectors. P1 focused the text label *Wort wählen* 'Choose word' in the inactive Next-Word-Panel at least once in each sentence connecting interval. The *skip connector*-button is the last element in focus of each sentence-connection interval.

Figure A.4 visualizes the eye-tracking data of the sentence-writing intervals S1-S4. The gaze plots (a), (c), (e), and (g) show repeated focus of the finished sentences in the text panel. The most recent sentence is focused upon most, which is also evident in the heat maps (b), (d), (f), (h). The *finish sentence*-button is focused last in the sentence-writing intervals. The Sentence Panel shows repeated focus on the words added to the sentence. The text input field and the topmost word in the suggestion list of the Next-Word Panel were focused upon for the longest time. In each sentence-writing interval, P1 focused the wh-cues. This is most evident in gaze plots (c) and (g) and heat maps (d) and (h).

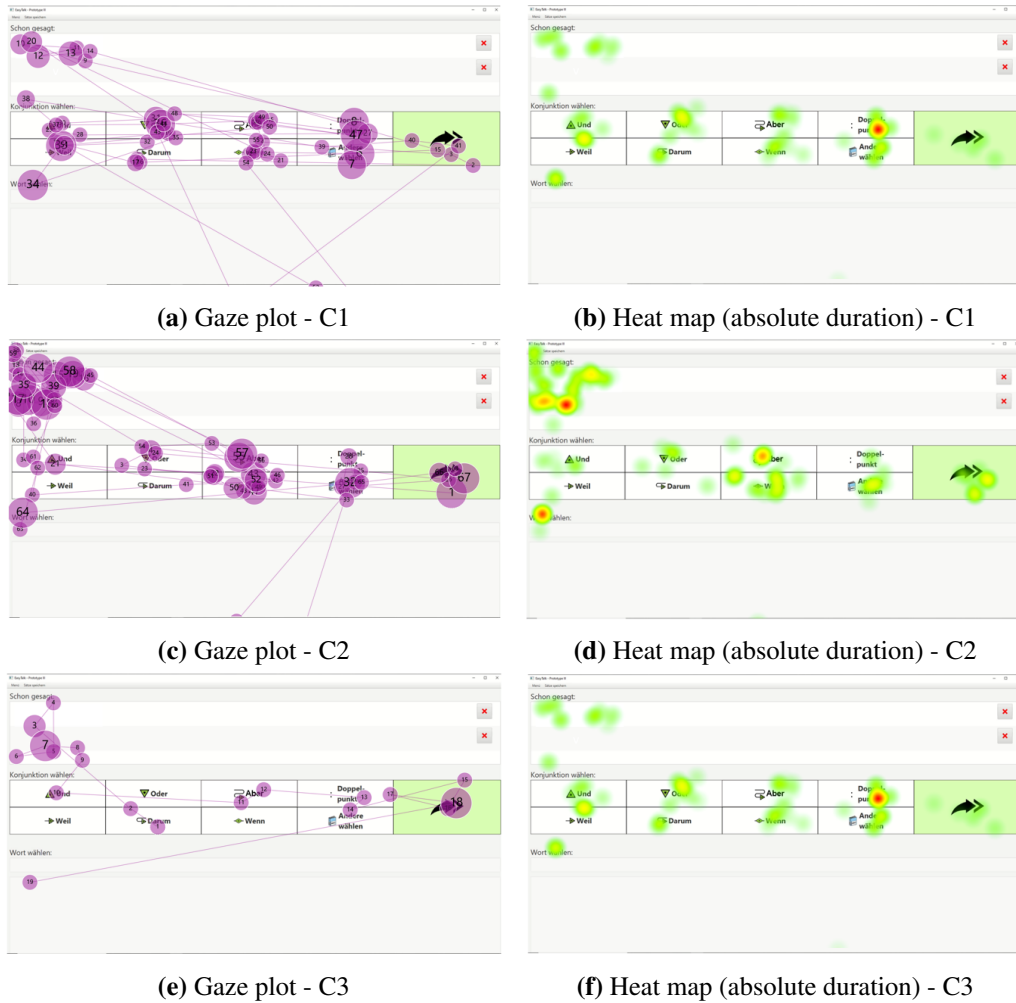
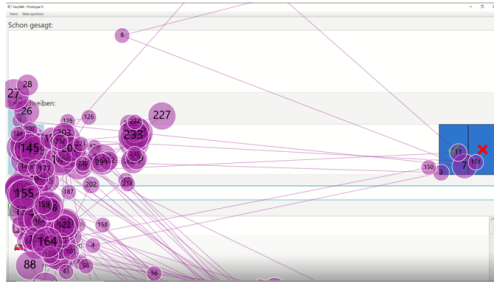


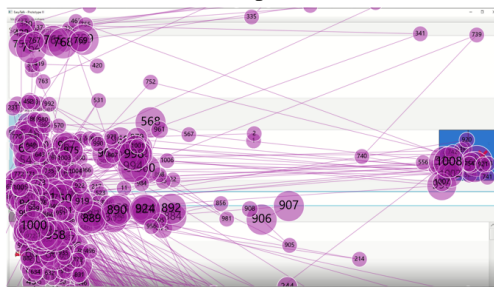
Figure A.3: The gaze plots (left column) and heat maps (right column) of participant P1's three sentence-connecting intervals C1, C2 and C3.



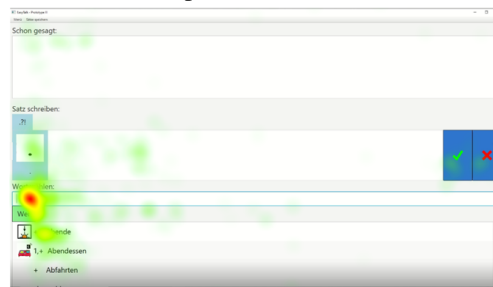
(a) Gaze plot - S1



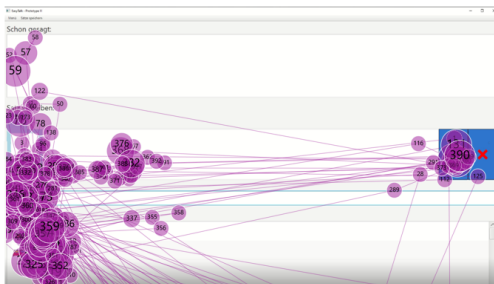
(b) Heat map (absolute duration) - S1



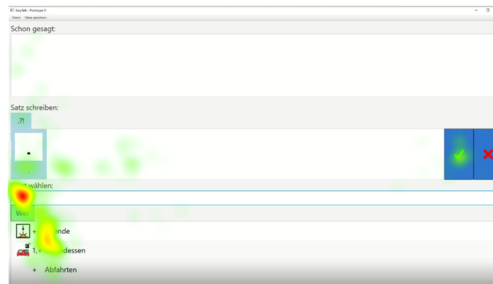
(c) Gaze plot - S2



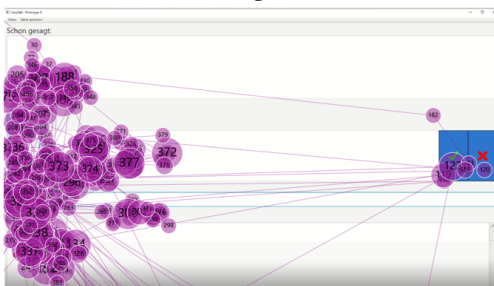
(d) Heat map (absolute duration) - S2



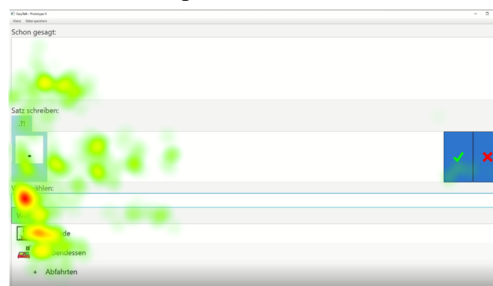
(e) Gaze plot - S3



(f) Heat map (absolute duration) - S3



(g) Gaze plot - S4



(h) Heat map (absolute duration) - S4

Figure A.4: The gaze plots (left column) and heat maps (right column) of participant P1's four sentence-writing intervals S1-S4.

A.4.2 Participant P2

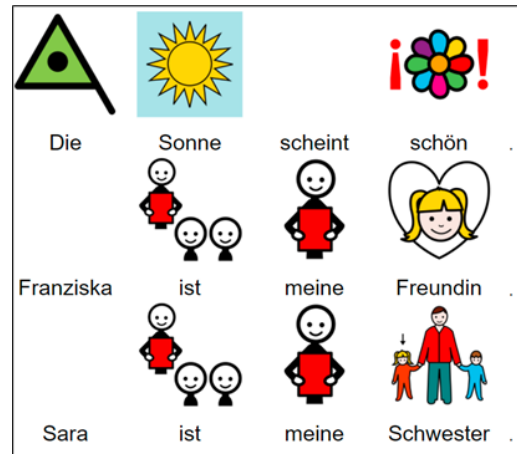


Figure A.5: Sentences written by P2 in the test session. Translation: S1: The sun shines nicely. S2: Franziska is my friend. S3: Sara is my sister.

Profile P2 is male, 20-25 years old, and has autism spectrum disorder and visual impairments. He is a native German speaker. During the test session, his caretaker was present but kept in the background. P2 phrased his sentences individually after the leader of the writing workshop helped him think of a topic to write about. P2 used both index fingers to type. He looked for the letters one by one. He moved and operated the mouse on his own. Due to his visual impairments, he sometimes had trouble spotting the mouse cursor. P2 agrees to the use of the eye tracker.

Observations

- P2 phrased his sentences on his own. Occasionally, the interviewer helped him with spelling.
- P2 voices that a bigger font would be of advantage for him.
- P2 says that a bigger mouse cursor *mit mehr Farbe* ‘with more color’ would be better for him.
- P2 typed all words to the end. When the interviewer pointed out to him that he could select the words from the suggestion list of the Next-Word Panel without typing them to the end, P2 answered: “*Mehr schreiben ist besser. Zu Ende schreiben ist besser.*” ‘Writing more is better. Writing to the end is better.’

- One time during the test session, the leader of the writing workshop asked P2: “*Klappts?*” ‘Is it going well?’. P2 answered: “*Ja super, aber ich muss mich hier bissl konzentrieren!*” ‘Yes, great, but I have to concentrate a bit here!’
- At the end of the session, the interviewer asked P2 whether to save his text with or without the symbols. P2 answered: “*Na mit Bildern natürlich!*” ‘With images, of course.’
- The interviewer asked P2 whether he felt like the symbols helped him or distracted him. P2 answered: “*Eigentlich egal. Weder noch.*” ‘Doesn’t matter, really. Neither nor.’
- P2 wanted to use the names of his sister and his friend in the text. Both names had to be added to the vocabulary first.
- P2 uses the read-aloud function several times.
- P2 takes his time to write his sentences.
- P2 voices that the system was easy to use. He added, that it would be beneficial to use it “*Zusammen*” ‘together’ (with a caretaker or teacher) in the beginning and then said: “*Später kann mans dann auch allein probieren.*” ‘Later on, you can try on your own.’
- The leader of the writing workshop asked P2 when he’d like to use the system again. P2 answered: “*Direkt am Donnerstag*” ‘Direct next tuesday.’ – meaning in the next writing workshop session two days later (the tests were conducted on a Tuesday).
- The interviewer asked P2 if he’d prefer to use *EasyTalk* without symbols P2 answered: “*Nein mit Bildern. Aber eigene Bilder sind besser. Mit Fotos und so.*” ‘No, with images. But using your own images is better. With fotos and such.’
- P2 said that he likes the wh-cues.
- P2 used the suggestion list in the Next-Word panel as spelling aid. He looked back and forth between the keyboard and the laptop screen and typed all words to the end. After making a typing error, P2 reacted autonomously and deleted the faulty characters on his own.
- P2 said that he has very limited experience with writing at the PC. When writing, he looked back and forth between the keyboard and the laptop screen.

- P2 wanted to write a sentence in the simple past tense, starting with *Ich war* 'I was'. However, *EasyTalk* did not yet support the use of simple past tense. Instead of writing *Ich bin gewesen* 'I have been', P1 discarded the sentence. He did not elaborate on what he had wanted to write.

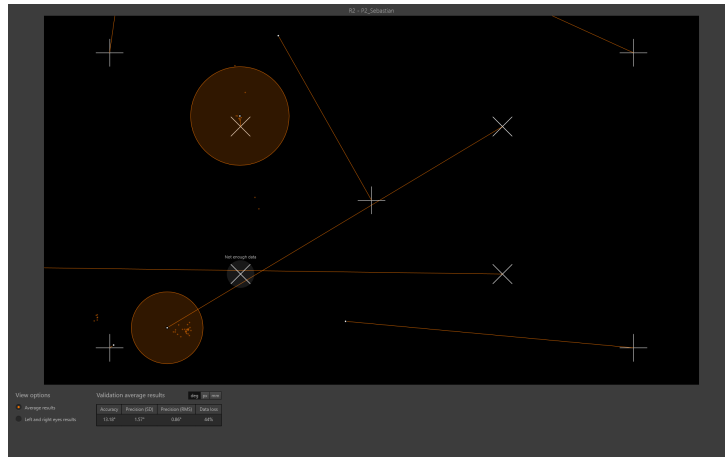


Figure A.6: P2's eye-tracking calibration-data exported from *Tobii Pro Lab*.

Eye-tracking data Figure A.6 shows P2's eye-tracking calibration values (13.18° Accuracy, 1.57° Precision (S), 0.86° Precision (RMS), and data loss of 44 %).

Figure A.7 visualizes the eye-tracking data of the sentence-connection intervals C1 and C2. Figure A.8 visualizes the eye-tracking data of the sentence-writing intervals S1-S4. The heat maps and gaze plots show large inaccuracies – supposedly due to P2 repeatedly looking away from the screen, P2 scanning the screen in search of the mouse cursor, and additional eye-tracking challenges due to P2's thick glasses and crossed eyes. The limited insights we can derive from the Figures are: P2 focused looked at the sentences in the Text Panel. His main focus was the text input field above the suggestion list of the Next-Word Panel. P2 focused the wh-cues.

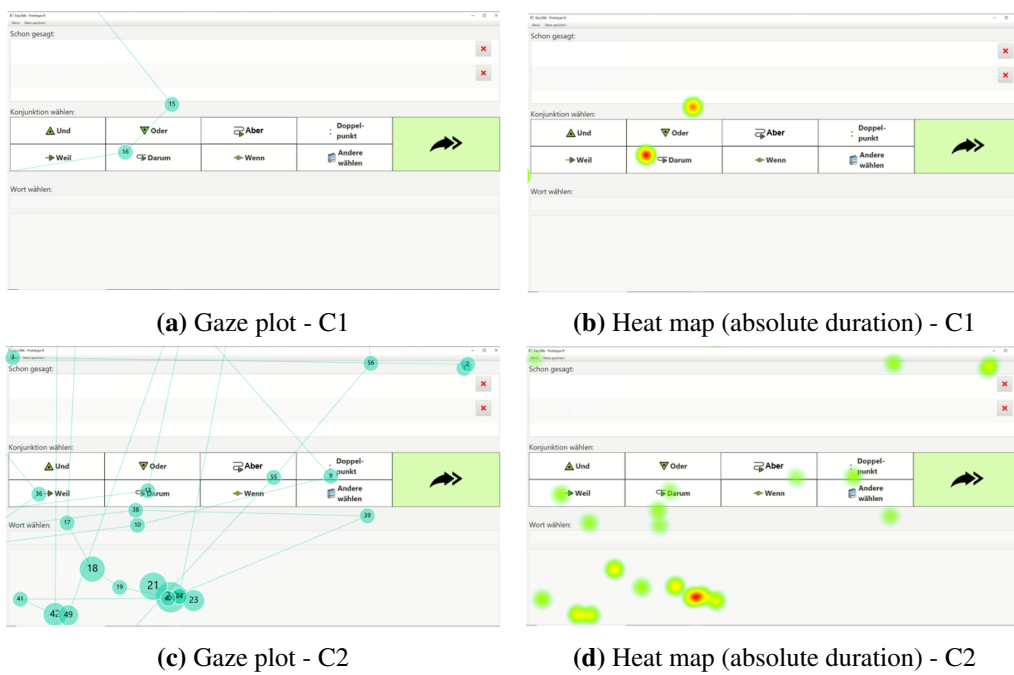


Figure A.7: The gaze plots (left column) and heat maps (right column) participant P2's two sentence-connecting intervals C1 and C2.

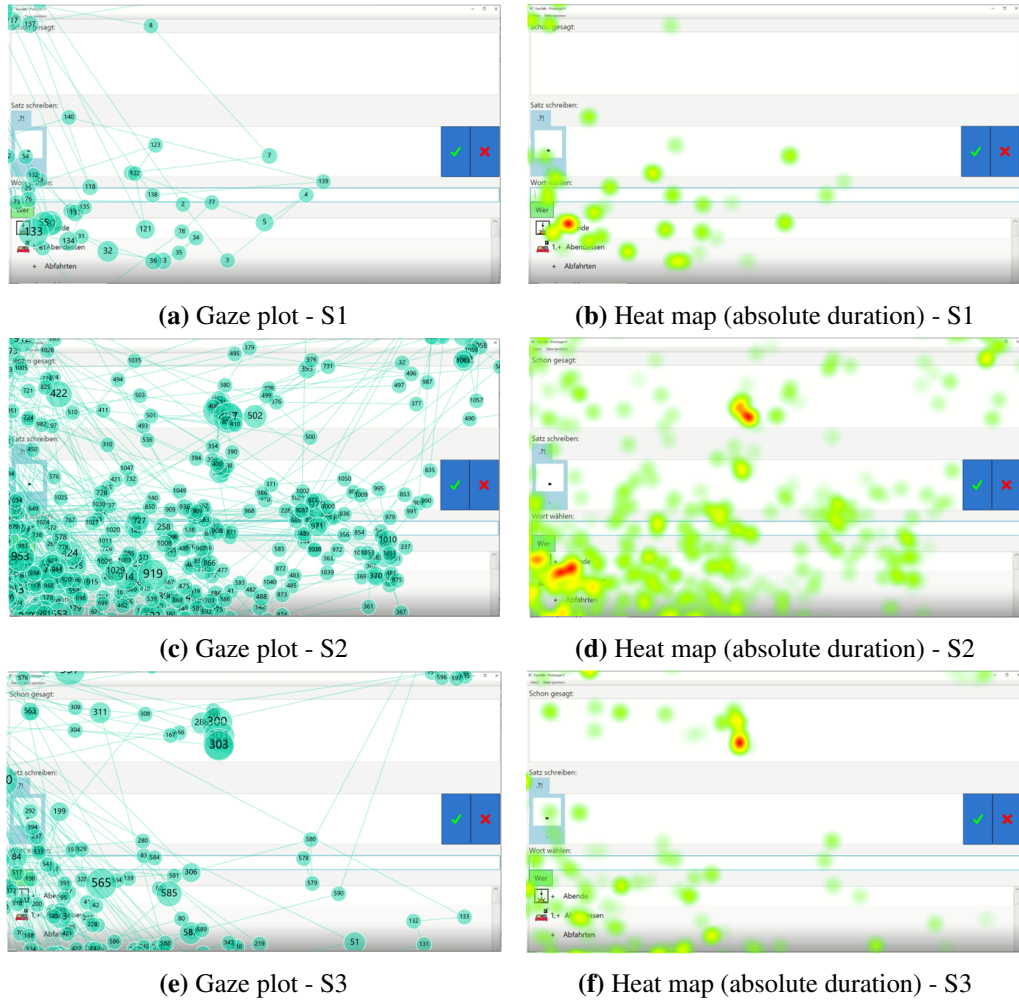


Figure A.8: The gaze plots (left column) and heat maps (right column) participant P2's three sentence-writing intervals S1, S2, and S3.

A.4.3 Participant P3

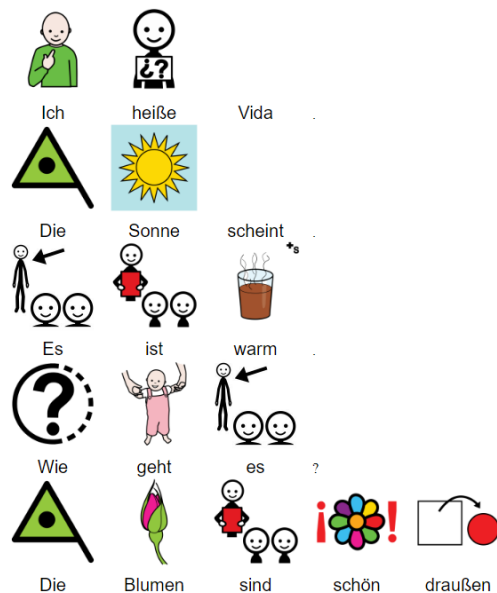


Figure A.9: The sentences P3 wrote in the test session. Translations: S1: ‘My name is Vida.’ S2: ‘The sun shines.’ S3: ‘It is warm.’ S4: ‘How are you?’ S5: ‘The flowers outside are beautiful.’

Profile P3 is a female, 18-20 years old, and German is not her mother tongue. She has severe hearing impairments and does not have functional speech. P3 read, write and understand simple short German sentences. During her test session, her caretaker was present. P3 communicated with him using hand gestures. The interviewer communicated the test setup in short, written sentences. The system introduction was done without words by use of three example sentence in *EasyTalk* (“*Die Sonne scheint. Ich heiße Ina. Wir schreiben zusammen.*” ‘The sun is shining. My name is Ina. We write together’). P3 indicated by using thumps-up/down gestures and nodding/shaking her head, whether everything was clear or if the interviewer had to repeat an action. P3 types independently, using several fingers on the keyboard. She operates the mouse on her own. P3 agrees to use the eye tracker.

Observations

- P3 spontaneously wrote a question.

- P3 wrote as first sentence “*Wie geht es?*” ‘How are you?’ – The interviewer assumed, that P3 intended to use *EasyTalk* for direct communication.
- P3 did not type words to the end. She selected the words from the suggestion list as soon as they appeared.

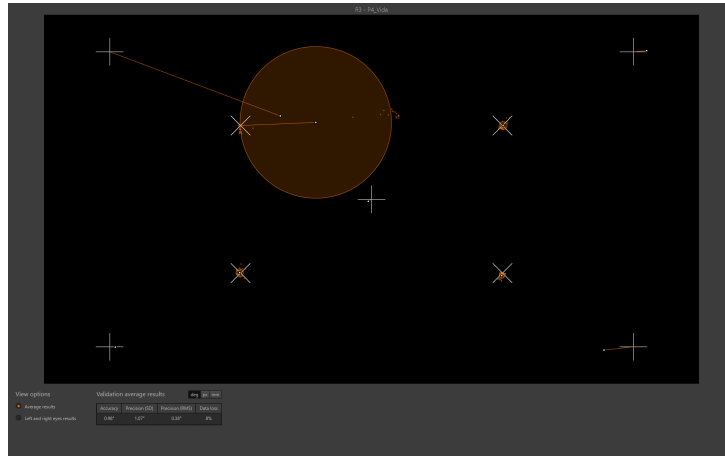
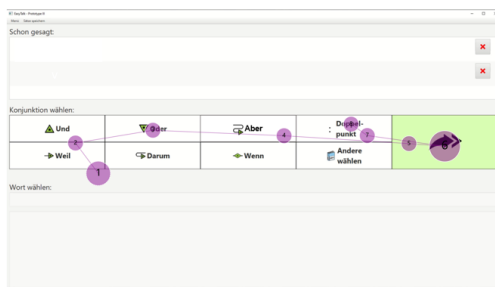


Figure A.10: P3’s eye-tracking calibration-data exported from *Tobii Pro Lab*.

Eye-tracking data Figure A.10 shows P3’s eye-tracking calibration values (0.98° Accuracy, 1.07° Precision (S), 0.38° Precision (RMS)).

Figure A.11 visualizes the eye-tracking data of the sentence-connection intervals C1-C5. The gaze plots and heat maps show that P3 looked at the connectors and focused the top row of connectors more (see, (a), (g), and (f)) . During the sentence-connection intervals, P3 focused less on the sentences in the text Panel than during the sentence-writing intervals.

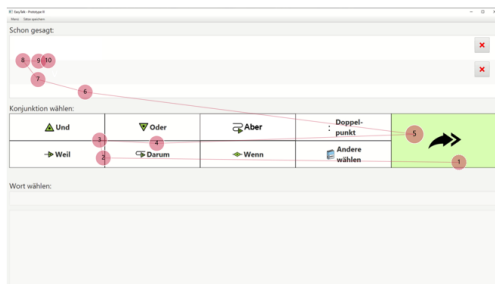
Figures A.12 and A.13 visualize the eye-tracking data of the sentence-writing intervals S1-S6. The gaze plots (a) and (c) in Figure A.12 show no focus on the Text Panel. In A.12 (e), and A.13 (a), (c), and (e), P3 focused the sentences in the Text Panel. At the end of each sentence-connection interval, P3 focused the *finish sentence*-button. The heat maps show that P3 focused the text input field of the Next-Word Panel and the topmost item of the word list the longest. The wh-cues were also viewed. The heat maps indicate a longer focus on the text elements than on the image symbols of the Sentence Panel and the Next-Word Panel.



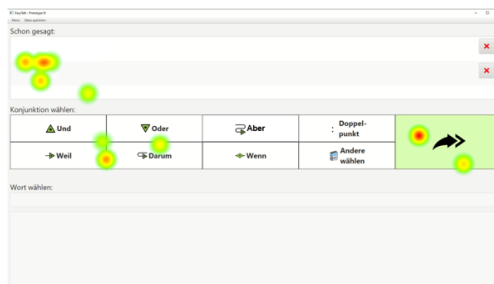
(a) Gaze plot - C1



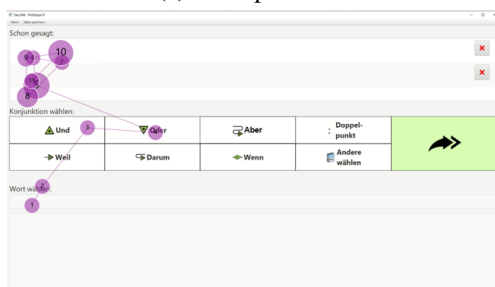
(b) Heat map (absolute duration) - C1



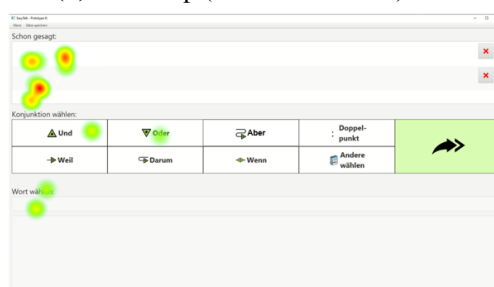
(c) Gaze plot - C2



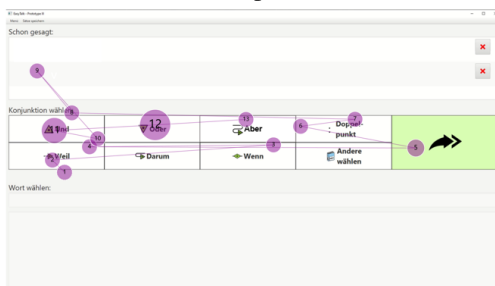
(d) Heat map (absolute duration) - C2



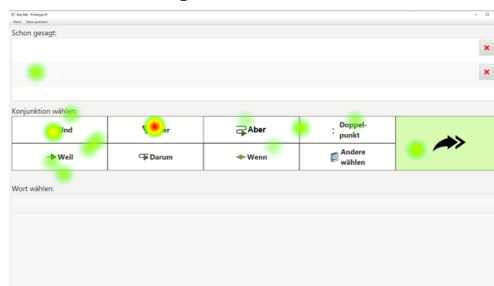
(e) Gaze plot - C3



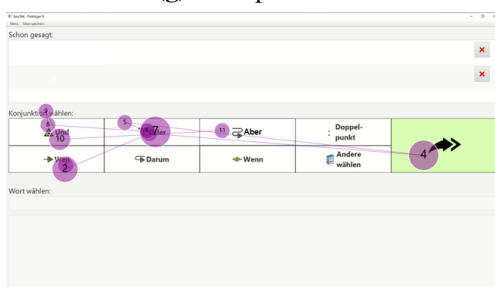
(f) Heat map (absolute duration) - C3



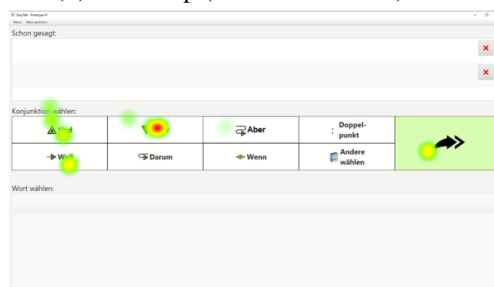
(g) Gaze plot - C4



(h) Heat map (absolute duration) - C4



(i) Gaze plot - C5



(j) Heat map (absolute duration) - C5

Figure A.11: The gaze plots (left column) and heat maps (right column) for the five sentence-connecting intervals C1-C5 of participant P3.

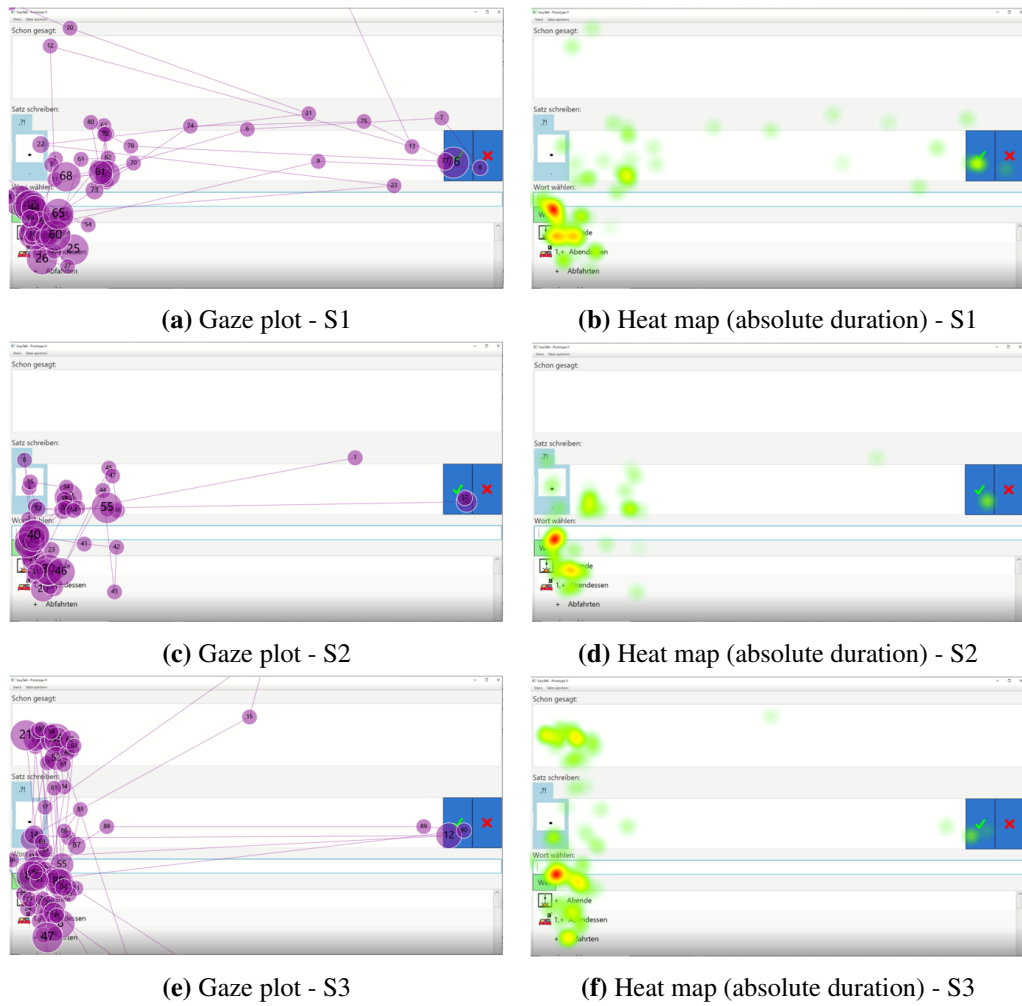


Figure A.12: The gaze plots (left column) and heat maps (right column) of participant P3's six sentence-writing intervals S1-S3.

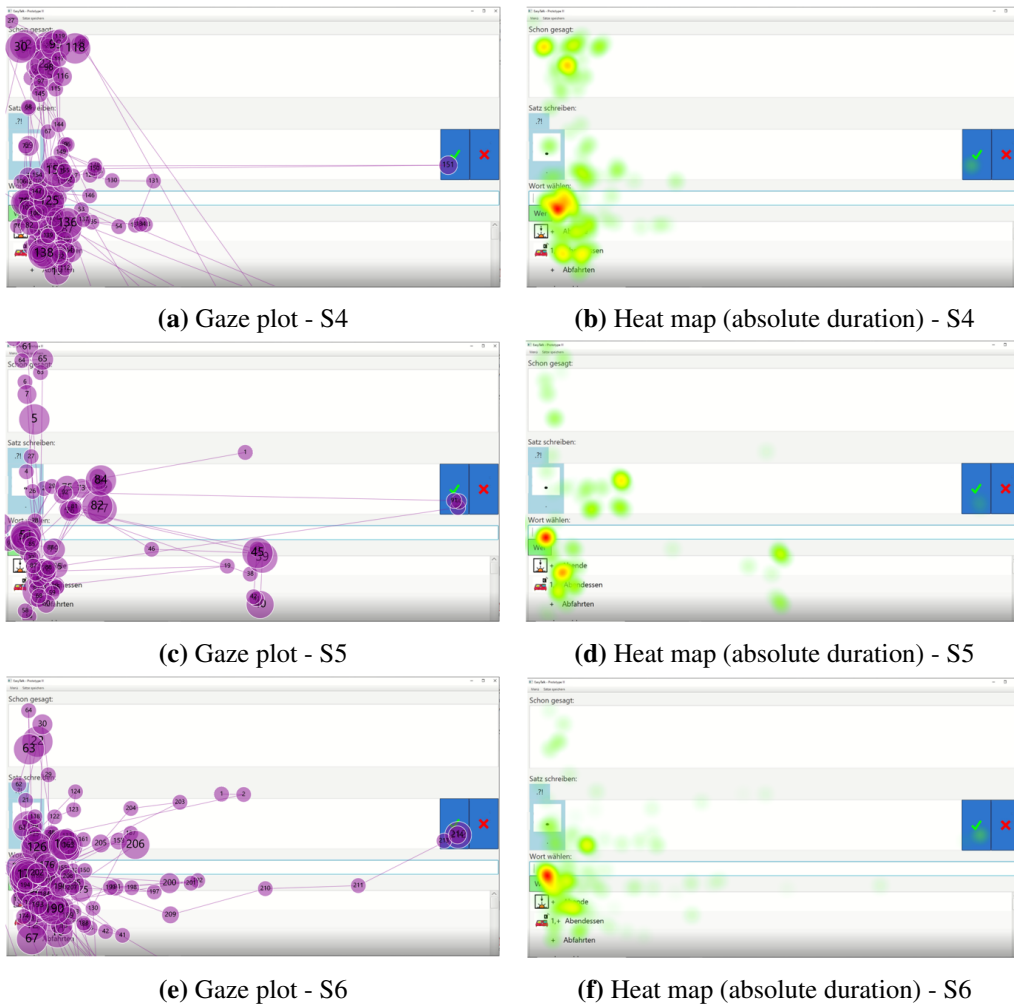


Figure A.13: The gaze plots (left column) and heat maps (right column) of participant P3's six sentence-writing intervals S5-S6.

A.4.4 Participant P4

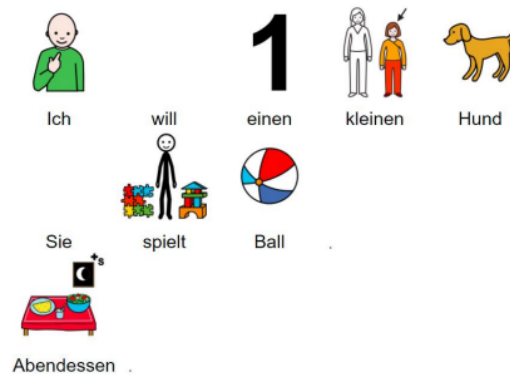


Figure A.14: The sentences P4 wrote in the test session. Translation: S1: I want a small dog. S2: She plays ball. S3: Supper.

Profile P4 is female, 20-25 years old, and has intellectual disabilities. P4 took part in the tests without a caretaker. She phrased her sentences on her own. Occasionally, the interviewer helped her with spelling. P4 typed on her own using her left index finger only. When typing, she looked onto the keyboard. She said she was using a laptop to write for the first time. The interviewer helped P4 operate the mouse. P4 agreed to use the eye tracker.

Observations

- P4 wrote her own sentence without practicing the sample sentence with the interviewer.
- P4 operated the decision dialog for modal verbs without difficulties.
- P4 typed autonomously.
- At first, P6 typed the words from start to finish. Then, she selected the words from the suggestion list when they appeared. She said that she liked this feature.
- P4 wanted to undo a word. For a moment, she looked for the delete button.
- P4 used double click to select a word. This resulted in faulty clicks. We note that using single or double click should be options in the system settings.
- Entering trailing white spaces after a word resulted in an unexpectedly empty suggestion list. The interviewer told P4 to remove the white spaces.

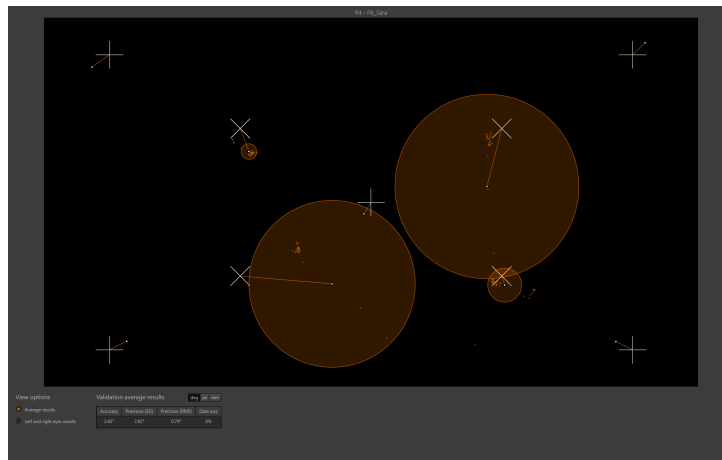
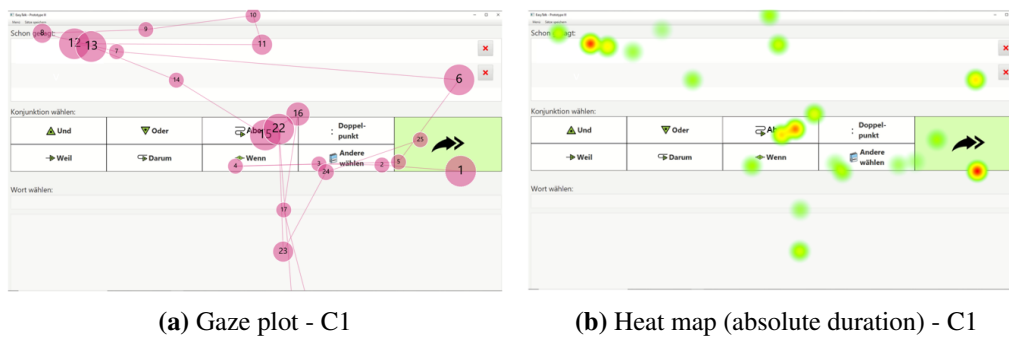


Figure A.15: P4's eye-tracking calibration-data exported from *Tobii Pro Lab*.



(a) Gaze plot - C1

(b) Heat map (absolute duration) - C1

Figure A.16: The gaze plot (left column) and heat map (right column) of P4's sentence-connecting interval C1.

Eye-tracking data Figure A.15 shows P4's eye-tracking calibration values (2.65° Accuracy, 2.92° Precision (S), 0.79° Precision (RMS)).

Figure A.16 visualizes the eye-tracking data of the sentence-connection interval C1. The gaze plots indicate that P4 looked at the sentence in the Sentence Panel, then briefly at the connectors before focusing the *Skip Connector*-button. However, looking at a singular, short connection interval we cannot assume a pattern here.

Figure A.17 visualizes the eye-tracking data of the sentence-writing intervals S1 and S2. The gaze plots (a), (c), and the heat maps (b) and (d) show no significant focus on the sentences in the Text Panel. During the session, P4 operated the *delete last word*-button repeatedly. This is reflected in the gaze plots. On finishing a sentence, P4 focused the check-mark-button. P4 looked at the text and symbols in the Sentence Panels. She viewed the wh-cues and focused the topmost item of the suggestion list the longest. Her eye-tracking data shows many outliers

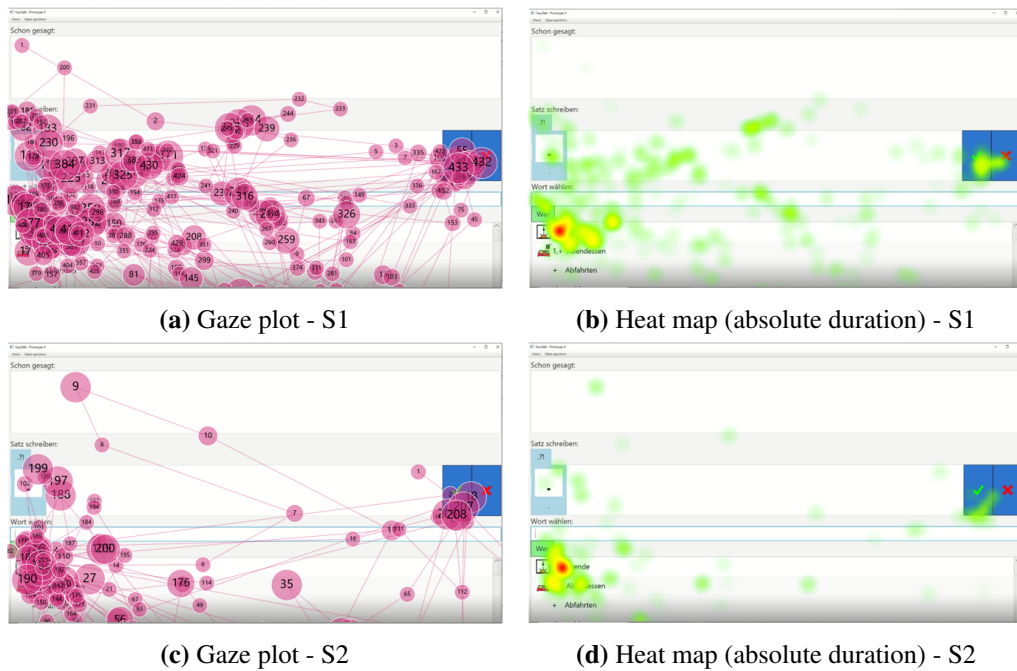


Figure A.17: The gaze plots (left column) and heat maps (right column) of participant P4's two sentence-writing intervals S1 and S2.

towards the bottom edge of the screen – presumably due to her looking down onto the keyboard when typing.

A.4.5 Participant P5

Profile P5 is 20-25 years old, male, and has intellectual disabilities. He phrased his sentences on his own. He operated the mouse and keyboard on his own. P5 agreed to use the eye tracker.

Observations

- Occasionally, the interviewer helped P5 spell.
- P5 looked at the keyboard most of the time. Therefore, he recognized typing mistakes when he looked up at the laptop screen. This resulted in an empty suggestion list at times. P5 corrected his typing mistakes by himself.
- P5 said that he'd prefer a bigger mouse cursor.



Figure A.18: The sentences P5 wrote in the test session. Translation: S1: I like to ride the tram. S2: I like to look at city maps. C1: And. S3: I ride the train.

- P5 double clicked to select words from the suggestion list. As *EasyTalk* expects single clicks, this resulted in faulty inputs. P5 deleted the unwanted words by himself.
- P5 said that the suggestion list helped the user: “*Man muss nicht ganz allein schreiben*” ‘One does not have to type completely alone.’
- P5 said that entering word by word helped him write.
- Referring to the wh-cues, P5 said : “*Die Überschriften helfen bei der Konzentration*” ‘The headers help with concentration.’
- P5 said: „*Die Verbinder zwischen den Sätzen sind wichtig. Manchmal gibt’s ja längere Sätze, dann kann man die Stück für Stück machen.*“ ‘The connectors between sentences are important. Sometimes there are longer sentences. This way, you can do them piece by piece’.
- P5 browsed the suggestion list to see what words it contains. He said that it is good to get ideas.
- P5 was pleasantly surprised to find symbols for trams and city maps.

Eye-tracking data Figure A.19 shows P5’s eye-tracking calibration values (7.10° Accuracy, 3.12° Precision (S), 0.80° Precision (RMS)).

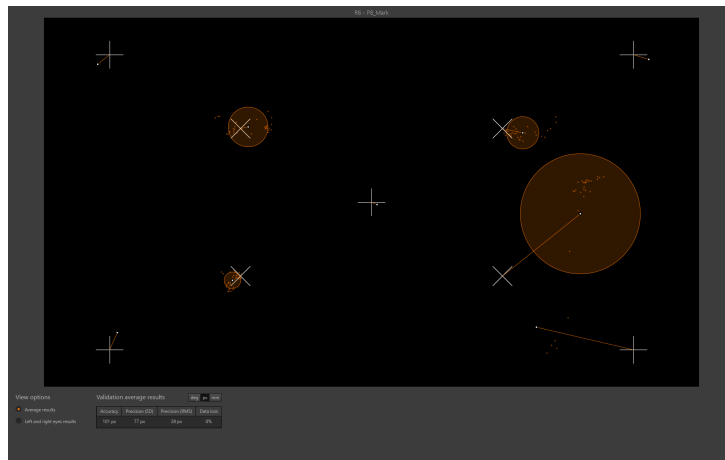
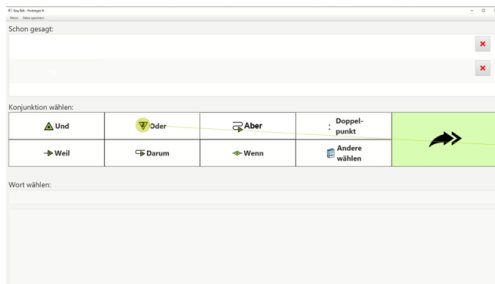


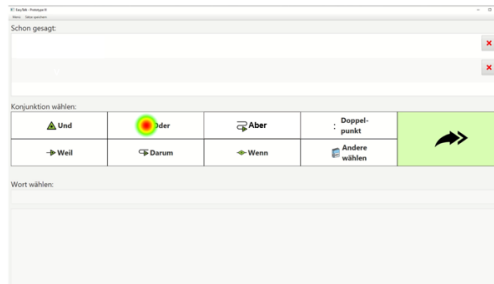
Figure A.19: P5's eye-tracking calibration-data exported from *Tobii Pro Lab*.

Figure A.20 visualizes the eye-tracking data of the sentence-connection intervals C1-C3. The first intervals, (gaze plots (a), (c), and heat maps (b),(d)) show no significant focus in the Sentence Connector Panel. In the third interval, C3, P5 focused the 6 leftmost elements of the Connector Panel before selecting the conjunction *und* 'and'. Heat map (f) indicates that the top row of connectors was focused longer. During the connection intervals, P5 did not look at the Text Panel.

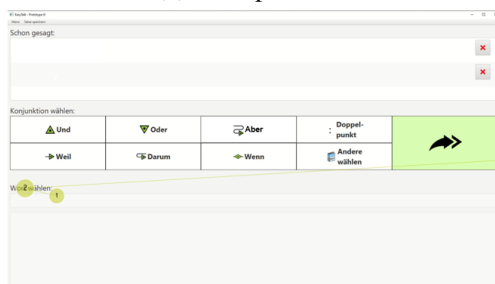
Figure A.21 visualizes the eye-tracking data of the sentence-writing intervals S1-S4. The eye-tracking data shows inaccuracies due to the P5's visual impairments. Nevertheless, the data reflects P5's statement that the wh-cues helped him focus: P5 repeatedly looked at the wh-elements in the Next-Word Panel. The majority P5's focus was on the two topmost elements of the word list and the wh-cues of the Next-Word Panel. The gaze plots and heat maps show that P5 looked at the previously written sentences in the Text Panel during the sentence-writing intervals. The eye-tracking data also show that P5 targeted the X-button to delete words and the *check mark*-button to finish a sentence. Due to the inaccuracies of the data, we cannot say to what extent P5 used the Sentence Panel element as thought support when writing.



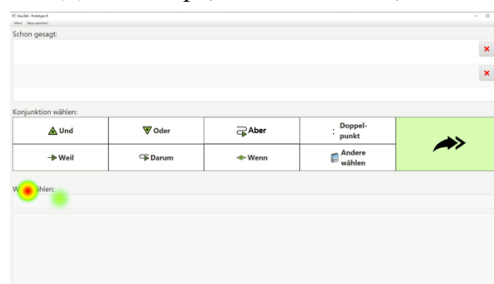
(a) Gaze plot - C1



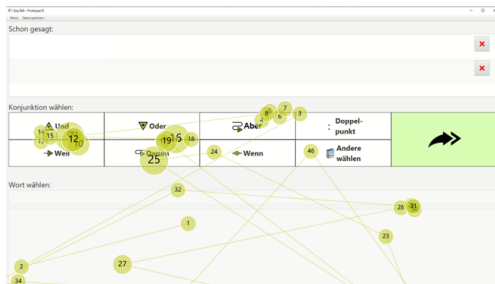
(b) Heat map (absolute duration) - C1



(c) Gaze plot - C2



(d) Heat map (absolute duration) - C2



(e) Gaze plot - C3



(f) Heat map (absolute duration) - C3

Figure A.20: The gaze plots (left column) and heat maps (right column) of participant P5's three sentence-connecting intervals C1, C2, and C2.

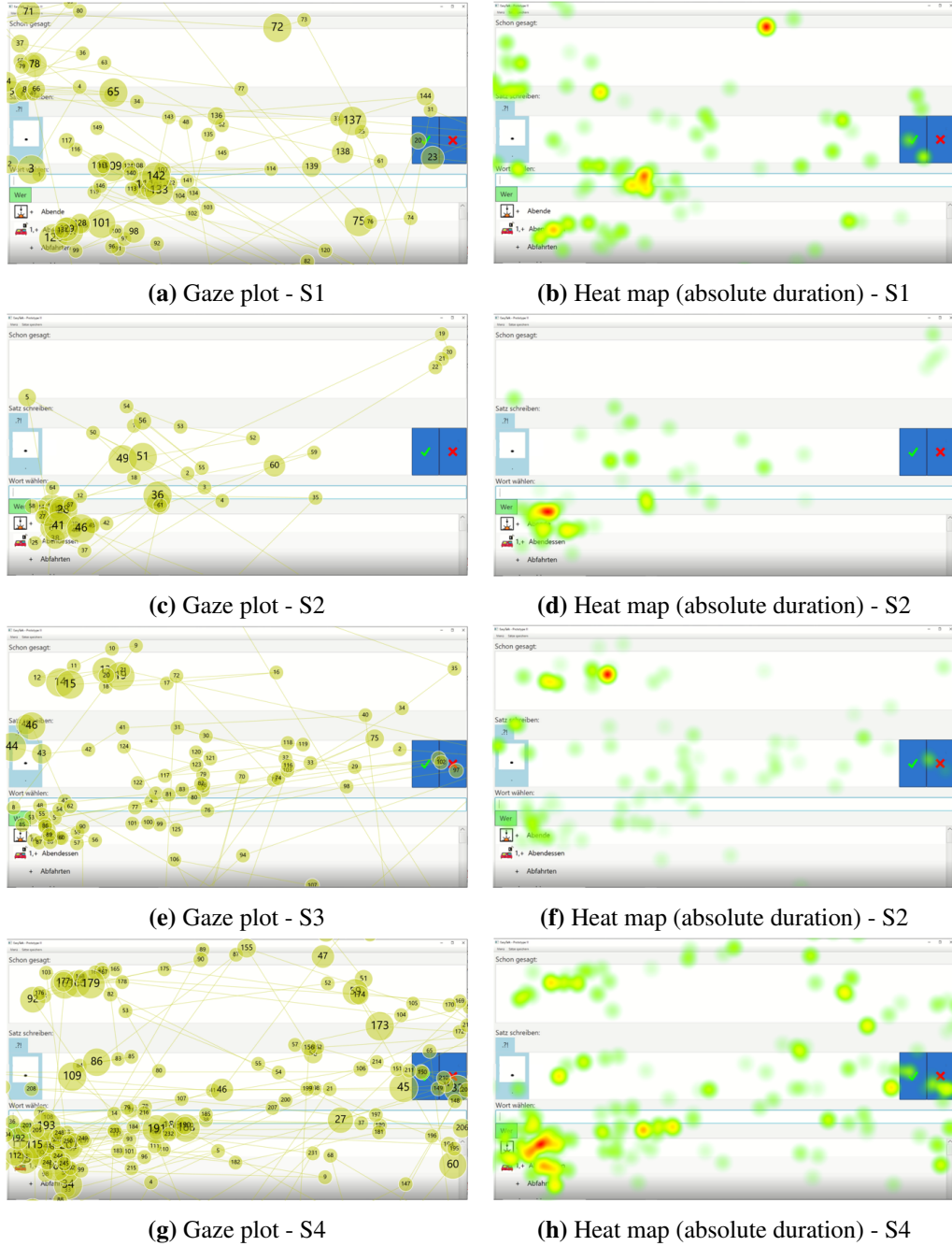


Figure A.21: The gaze plots (left column) and heat maps (right column) of participant P5's four sentence-writing intervals S1-S4.

A.4.6 Participant P6

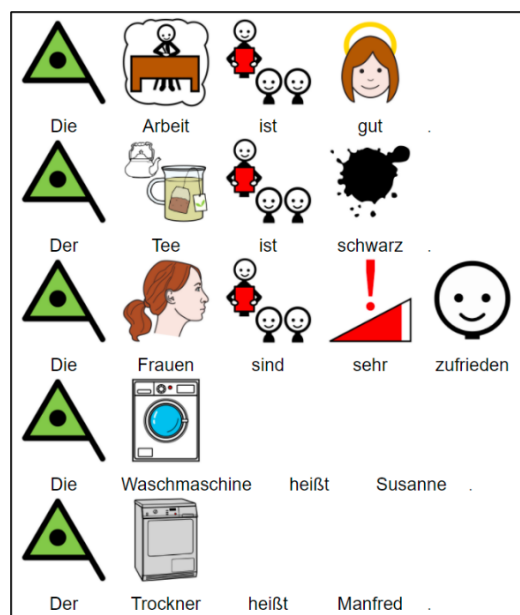


Figure A.22: The sentences P6 wrote in the test session. Translation: S1: The work is good. S2: The tea is black. S3: The women are very pleased. S4: The washing machine is called Susanne. S5: The tumble dryer is called Manfred.

Profile P6 is male, 20-25 years old, and a native German speaker. He has cognitive impairments. During his test session, the writing workshop leader was present and helped P3 find topics to write about. P6 has basic computer skills. In his free time, he regularly uses the PC to search for images on topics that interest him. He typed independently. If he made a typing error, he corrected his mistakes independently. P6 preferred not to use the eye tracker.

Observations

- P6 wrote independently. Only a few times, the interviewer assisted with spelling.
- The interviewer noticed that P6 did not make any use of the wh-cues and explained their functionality a second time. P6 did not comment nor use the wh-cues thereafter.
- P6 wrote about his workday prior to the test session. He wanted to use the simple past tense, writing *Die Arbeit war gut*. ‘Work was good.’. Because

EasyTalk did not yet support the use of simple past tense, P6 opted to write *Die Arbeit ist gut*. ‘Work is good.’ instead of using present perfect tense (*Die Arbeit ist gut gewesen*. ‘Work has been good.’)

- P6 types all words to the end.
- P6 wanted to write about his caretakers at work and use their names, which would have needed to be added to the vocabulary first. Instead of adding the names to the vocabulary, P6 opted to write *Die Frauen* ‘the women’. The names of the washing machine and the tumble dryer – an inside joke between P6 and his caretakers – were added to the vocabulary by the interviewer.
- P6 appreciated the symbols. In particular, there were even ‘uncommon’ symbols, such as the one showing a tumble dryer.
- Due P6’s slight motor impairments, the interviewer assisted P6 in moving the mouse occasionally.
- P3 used the read-aloud function several times to read aloud sentences repeatedly in a row.

A.4.7 Participant P7

Profile P7 is female, 20-25 years old, and has intellectual disabilities and motor impairments. She used a motorized wheelchair that she operated with a joystick using the heel of her right hand. She took part in the test session without a caretaker. P7 typed on her own but received help by the interviewer when she accidentally activated the touch pad of the laptop. Moreover, the interviewer helped P7 to move the mouse. P7 then clicked the mouse. P7 opted not to use the eye tracker.

Observations

- At first, P7 said she just wanted to write one sentence using *EasyTalk*. In the end, she wrote 4 more sentences.
- P7 typed all words to the end.
- P7 said that she preferred to use *EasyTalk* with symbols.
- P7 said that she likes the symbols. She noted that she’d like to add her own images to the system, such as a photo of her mother and her dog.

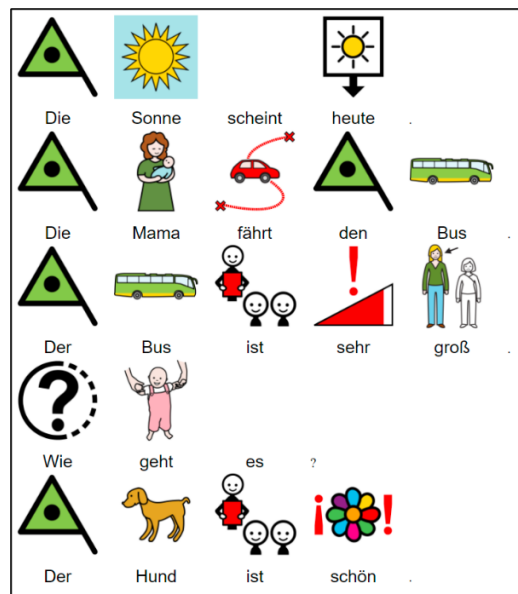


Figure A.23: The sentences P7 wrote in the test session. Translation: S1: The sun shines today. S2: The mom drives the bus. S3: The bus is very big. S4: How are you? S5: The dog is beautiful.

- P7 said that she would like to use *EasyTalk* again to practice writing.
- We observed that using a customized mouse would be beneficial for P7.
- P7 spontaneously wrote a question. She set the sentence mode on her own without difficulties.
- P7 used the read-aloud function of the Text Panel several times.

A.4.8 Participant P8

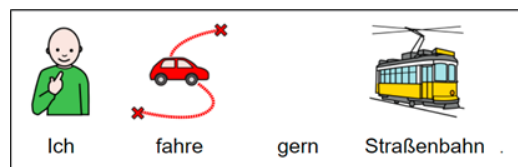


Figure A.24: The sentence P8 wrote before she opted out of the test session. Translation: S1: I like to ride the tram.

Profile P8 is female, 20-25 years old, and has learning difficulties. She takes part in the test session without a caretaker. Regularly, she uses *MS Word* to write. After writing one sentence with *EasyTalk*, P8 stated that she would not benefit from using *EasyTalk* and decided to end the test session.

Observations

- P8 wanted to start writing without the interviewer showing a sample sentence.
- P8 wanted to enter the complete sentence into the Text Panel. The interviewer explained that she had to enter word by word.
- P8 wrote one sentence, then she said that she would not benefit from using *EasyTalk*. She said that using *EasyTalk* slowed her down and that she could write faster in *MS Word*.

A.4.9 Additional eye-tracking metrics

This section contains the participants' calibration data (Figure A.1), an overview of the areas of interest defined to analyze the eye-tracking recordings (Figure A.25), and additional eye-tracking metrics (Tables A.2, A.3, and A.4). The visit durations depicted in the tables provide insights into how long the participants viewed a specific AOI. A visit is defined as the elapsed time between the first gaze point recorded in the target AOI and the last gaze point in the sequence of gaze points before leaving the AOI. (For details see the *Tobii Pro Lab* documentation on eye-tracking metrics at connect.tobii.com/s/article/understanding-tobii-pro-lab-eye-tracking-metrics) The *accumulated total visit duration* describes the visit duration for all visits of all participants of a target AOI during a predefined time of interest – here, the set of all sentence-connecting intervals or sentence-writing intervals, respectively.

Table A.1: The eye-tracking calibration values of participants P1-P5 exported from *Tobii Pro Lab*. Validation Precision: Standard Deviation. Precision: Root Mean Square

	P1	P2	P3	P4	P5
Validation Accuracy $^{\circ}$	7.10	13.08	0.98	2.65	1.65
Validation Precision $^{\circ}$	3.12	1.57	1.07	2.92	1.26
Precision $^{\circ}$	0.80	0.86	0.38	0.79	0.79
Data loss %	26	44	8	0	0
Gaze samples %	76	54	37	73	35

Table A.2: Total visit duration (seconds) of the sum of sentence-writing intervals per panel and participant P1-P5. A sentence-writing-interval starts with the activation of the Sentence Panel and ends with finishing the sentence by clicking the checkmark-button.

	Text-Panel	Sentence-Panel	Next-Word-Panel	All panels
P1	29.24	128.03	350.07	537.04
P2	22.18	9.89	62.44	94.51
P3	15.74	25.45	109.99	151.18
P4	0.72	40.10	101.64	142.46
P5	22.68	31.34	116.85	142.46
Total (s)	90.57	234.83	740.99	1.066.39
Total (%)	8.49	22.02	69.49	100%

Table A.3: Total visit duration (seconds) of the sum of sentence-connecting intervals per panel and participant. A sentence-connecting-interval starts with the activation of the Connector Panel and ends as soon as a connector is chosen or the skip-button is activated.

	Text Panel	Connector Panel	Next-Word Panel	All panels
P1	14.87	43.28	2.58	60.73
P2	1.61	1.46	4.08	7.15
P3	0.23	7.38	0.0	7.61
P4	3.83	4.85	0.50	9.18
P5	0.0	6.51	3.50	10.01
Total (s)	20.54	63.50	20.66	104.7
Total (%)	21.69	67.05	11.26	100%

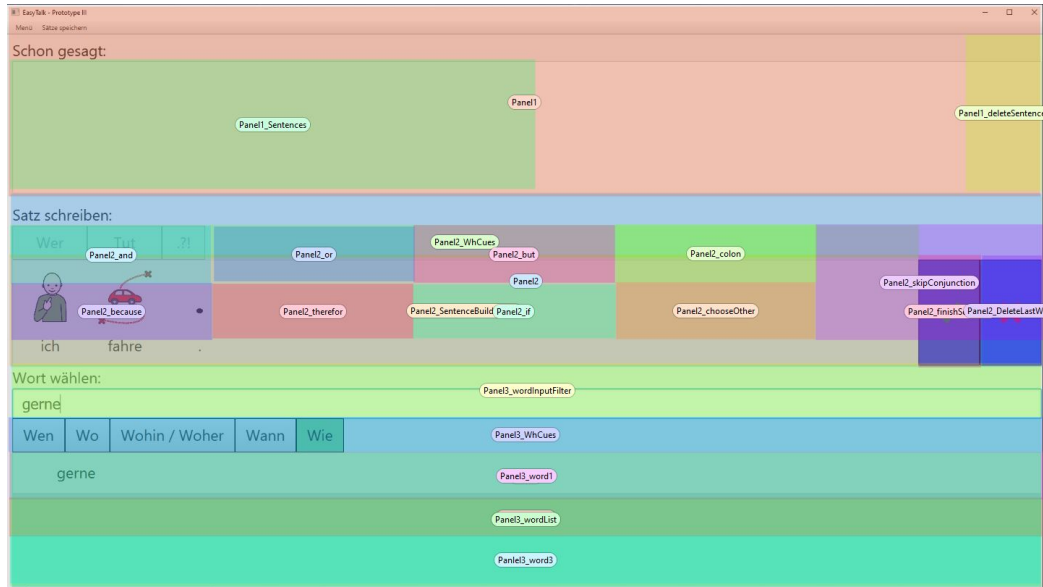


Figure A.25: The areas of interest defined for eye tracking.

Table A.4: Total visit durations (seconds) of the sum of sentence-connecting intervals per participant P1-P5 in the AOIs of the Connector Panel.

Connector/ Participant	P1	P2	P3	P4	P5	Total (s)	Total (%)
And	0	0	1.21	0	1.24	2.46	18.59
Or	0	0	2.17	0	0.07	2.23	16.83
But	0	0.13	0.14	1.40	0.84	2.52	19.03
:	0	0.20	0.42	0.00	0.09	0.71	5.40
Because	0	0.19	0.84	0	0.92	1.95	14.76
Therefor	0	0.13	0	0.00	1.08	1.21	9.17
If	0	0	0.09	0.25	0.24	0.59	4.52
Choose other	0	0	0	0.35	0.06	0.41	3.14
Skip	0	0	0.77	0.37	0	1.13	8.54
Total (s)	0	0.66	5.64	2.36	4.58	13.25	100.00

Appendix B

Additions

B.1 Data Availability Statement

The raw data supporting the conclusions of this work will be provided by the authors on request, without undue reservation.

B.2 Ethics Statement

Ethical review and approval were not required for the study on human participants in accordance with the local legislation and institutional requirements. The participants – or their legal guardians – provided their written informed consent to participate in this study.

B.3 Curriculum Vitae

	PERSONAL DATA
Name	Ina Steinmetz
Date of birth	February 14, 1989
Place of birth	Stuttgart
Contact	contact@inasteinmetz.de
	PROFESSIONAL EXPERIENCE
2020 – present	Senior Solution Consultant <i>Tacton Systems GmbH</i> , Karlsruhe, Germany Full stack development and technical project leadership, CPQ, 3D visualization and visual configuration
10/2016 - 2020	Solution Consultant <i>Tacton Systems GmbH</i> , Karlsruhe, Germany Full stack development, CPQ, 3D visualization and visual configuration
01 / 2015 – 09 / 2016	IT Consultant <i>EXXETA GmbH</i> , Karlsruhe, Germany Full stack development and user experience design, web and mobile solutions
	EDUCATION
10 / 2015 – present	External Phd student: Computational Linguistics <i>University of Koblenz-Landau</i> , Koblenz, Germany
Thesis	Working Title: ‘Interactive Generation of <i>Leichte Sprache</i> (Easy German) Deploying Customizable Symbols for AAC Users’
Doctorate Supervisor	Prof. Dr. Karin Harbusch
10 / 2012 – 11 / 2014	Master of Science Computational Visualistics <i>University of Koblenz-Landau</i> , Koblenz, Germany
Thesis	‘Developing a Symbol-Based AAC-System’
Overall Mark	1,5
10 / 2009 – 11 / 2012	Bachelor of Science Computational Visualistics <i>University of Koblenz-Landau</i> , Germany
Thesis	‘Detection of Articulating Objects in Range Image Sequences’
Overall Mark	2,2
09/ 2008 – 09/ 2009	Gemeinnütziges Bildungsjahr (Voluntary Educational Year) <i>Citizen Service</i> , Stuttgart, Germany Direct intern of the district representative of Stuttgart Weilimdorf, local citizen service, project and event management
1999 – 2008	General qualification for university entrance <i>Herzog-Christoph Gymnasium Beilstein</i> , Beilstein, Germany

Bibliography

- AppNotize UG. Snapshot of letmetalk on an iphone. <https://apps.apple.com/de/app/letmetalk-gratis-aac-talker/id919990138>, 2022. [Online] Accessed 11-January-2022.
- ARASAAC. Symbol set and resources for Augmentive and Alternative Communication (AAC). arasaac.org, 2022. [Online] Accessed 06-December-2021.
- Rúbia Eliza de Oliveira Schultz Ascari, Roberto Pereira, and Luciano Silva. Mobile interaction for augmentative and alternative communication: a systematic mapping. *Journal on Interactive Systems*, 9(2), Aug. 2018. doi: 10.5753/jis.2018.704. URL <https://sol.sbc.org.br/journals/index.php/jis/article/view/704>.
- Allan Bell. Language Style as Audience Design. *Language in Society*, 13(2): 145–204, 1984. doi: 10.1017/S004740450001037X.
- Joachim Bingel. *Personalized and Adaptive Text Simplification*. PhD thesis, Department of Computer Science, Faculty of Science, University of Copenhagen, Copenhagen, Denmark, 2018.
- BITV2.0. Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz (Barrierefreie-Informationstechnik-Verordnung - BITV 2.0). Barrierefreie-Informationstechnik-Verordnung vom 12. September 2011 (BGBl. I S. 1843), die zuletzt durch Artikel 1 der Verordnung vom 21. Mai 2019 (BGBl. I S. 738) geändert worden ist http://www.gesetze-im-internet.de/bitv_2_0/BJNR184300011.html, 2011.
- BMAS. Leichte sprach. ein ratgeber. abdruck der "regeln für leichte sprache" des netzwerks leichte sprache e.v. <https://www.bmas.de/SharedDocs/Downloads/DE/Publikationen/a752-ratgeber-leichte-sprache.pdf;jsessionid=4E0675208F94BC69CA49FE6DE0173AB0.delivery1->

- replication?__blob=publicationFile&v=3, 2013. Accessed 29-March-2022.
- Bettina M. Bock. Das Passiv- und Negationsverbot ‘Leichter Sprache’ auf dem Prüfstand – Empirische Ergebnisse aus Verstehenstest und Korpusuntersuchung. *Sprachreport*, 33 (1):20–28, 2017. URL https://ids-pub.bsz-bw.de/frontdoor/deliver/index/docId/6109/file/Bock_Passiv_und_Negationsverbot_2017_1.pdf.
- Bettina M. Bock. „Leichte Sprache“ – Kein Regelwerk: Sprachwissenschaftliche Ergebnisse und Praxisempfehlungen aus dem LeiSA-Projekt. Frank & Timme, Berlin, Germany, 2019. URL <https://ul.qucosa.de/api/qucosa%3A31959/attachment/ATT-0/>.
- Agnieszka Bojko. Eye tracking in user experience testing: How to make the most of it. *Proceedings of the 14th Annual Conference of the Usability Professionals’ Association (UPA)*. Montréal, Canada, 01 2005.
- Silke Brandt, Elena Lieven, and Michael Tomasello. German children’s use of word order and case marking to interpret simple and complex sentences: Testing differences between constructions and lexical items. *Language Learning and Development*, 12(2):156–182, 2016. doi: 10.1080/15475441.2015.1052448. URL <https://doi.org/10.1080/15475441.2015.1052448>.
- Joan Bruno. Gateway for snap - USER MANUAL - child language and literacy page set. <https://www.gatewaytolanguageandlearning.com/resources/ewExternalFiles/Gateway%20Child%20Language%20and%20Literacy%20Manual.pdf>, 2020. Accessed 11-January-2022.
- Klaus Buddeberg, Gregor Dutz, Anke Grotlüschen, Lisanne Heilmann, and Christopher Stammer. Low literacy in germany: Results from the second german literacy survey. *European Journal for Research on the Education and Learning of Adults*, 11(1):127–143, Feb. 2020. doi: 10.3384/rela.2000-7426.rela9147. URL <https://rela.ep.liu.se/article/view/3869>.
- Katja Burgherr. *LEICHTE SPRACHE IM ASYLWESEN: Eine praktische Untersuchung der Leichten Sprache in der Zielgruppe «Asylsuchende»*. PhD thesis, University of Geneva, 2021.
- James I Charlton. Nothing about us without us. In *Nothing About Us Without Us*. University of California Press, 1998.

- Robert Dale. Gpt-3: What's it good for? *Natural Language Engineering*, 27(1): 113–118, 2021. doi: 10.1017/S1351324920000601.
- Robert Dale and Jette Viethen. The automated writing assistance landscape in 2021. *Natural Language Engineering*, 27(4):511–518, 2021. doi: 10.1017/S1351324921000164.
- Christian Dejffal and Antonia Horst. Übersetzungen und Künstliche Intelligenz in der Öffentlichen Verwaltung. *Berichte des NEGZ*, 17, 2021. doi: 10.30418/2626-6032.2021.17.
- Patrick W. Demasco and Kathleen F. McCoy. Generating text from compressed input: An intelligent interface for people with severe motor impairments. *Communications of the ACM*, 1992. doi: <https://doi.org/10.1145/129875.129881>.
- Sarah Ebling, Alessia Battisti, Marek Kostrzewa, Dominik Pfütze, Annette Rios, Andreas Säuberli, and Nicolas Spring. Automatic text simplification for german. *Frontiers in Communication*, 7, 2022. ISSN 2297-900X. doi: 10.3389/fcomm.2022.706718. URL <https://www.frontiersin.org/article/10.3389/fcomm.2022.706718>.
- Elad Elram. Snapshot of symbotalk on an ipad. <https://www.symbotalk.com/>, 2017. [Online] Accessed 11-January-2022.
- Elad Elram. Foto of gateway on an ipad. <https://rehamedia.de/wp-content/uploads/2019/12/Vibe-mit-Gateway.jpg>, 2022. [Online] Accessed 11-January-2022.
- Susan Fager, Karen Hux, David Beukelman, and Renee Karantounis. Augmentative and alternative communication use and acceptance by adults with traumatic brain injury. *Augmentative and Alternative Communication*, 22(1):37–47, 2006. doi: 10.1080/07434610500243990.
- Walia Farzana, Farhana Sarker, Tom Chau, and Khondaker A. Mamun. Technological evolvment in aac modalities to foster communications of verbally challenged asd children: A systematic review. *IEEE Access*, pages 1–1, 2021. doi: 10.1109/ACCESS.2021.3055195.
- Federal Ministry of Labour and Social Affairs. Short version of the national action plan. <https://www.bmas.de/SharedDocs/Downloads/DE/Publikationen/a740-short-en.pdf>, 2012. [Online] Accessed 29-April-2022.

- Luciano Floridi and Massimo Chiriatti. Gpt-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30(4):681–694, 2020. doi: 10.1007/s11023-020-09548-1.
- Vincent Fortuin, Romann Weber, Sasha Schriber, Diana Wotruba, and Markus Gross. Inspireme: learning sequence models for stories. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, April 2018. doi: <https://ojs.aaai.org/index.php/AAAI/article/view/11407>.
- Gateway to Language and Learning. Gateway page sets. <https://www.gatewaytolanguageandlearning.com/product-information/gateway-for-mind-express.html>, 2022. [Online] Accessed 11-January-2022.
- Albert Gatt and Emiel Krahmer. Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61:65–170, 2018.
- Donald H. Graves and Donald M. Murray. Revision: In the writer’s workshop and in the classroom. *Journal of Education*, 162(2):38–56, 1980. doi: 10.1177/002205748016200205.
- Anke Grotlüschen and Klaus Buddeberg. *LEO 2018 - Leben mit geringer Literalität Bielefeld*, wbv Media, ISBN: 978-3-7639-6071-2. 03 2020. ISBN 978-3-7639-6071-2.
- Anke Grotlüschen and Wibke Riekman. leo.-level-one studie. literalität von erwachsenen auf den unteren kompetenzniveaus. *Magazin erwachsenbildung. at*, (10), 2011. doi: <https://doi.org/10.25656/01:7515>.
- Anke Grotlüschen and Wibke Riekman. Funktionaler analphabetismus in deutschland. *Ergebnisse der ersten leo, Level-One Studie. Münster ua*, 2012.
- Leon Gulikers, Gilbert Rattnik, and Richard Piepenbrock. German Linguistic Guide of the CELEX lexical database. In *Linguistic Data Consortium*. Philadelphia, MA, USA, 1995.
- Silke Gutermuth. *Leichte Sprache für alle?: eine zielgruppenorientierte Rezeptionsstudie zu Leichter und Einfacher Sprache*, volume 5. Frank & Timme GmbH, 2020.
- Silvia Hansen-Schirra and Christiane Maaß. *Easy Language Research: Text and User Perspectives*. Frank & Timme, Berlin, Germany, 2020. doi: <http://dx.doi.org/10.26530/20.500.12657/42088>.

- Karin Harbusch and Gerard Kempen. A quantitative model of word order and movement in English, Dutch and German complement constructions. In *Proceedings of the 19th international conference on Computational Linguistics – Volume 1 (COLING '02)*, pages 1–7, Taipei, Taiwan, 2002. doi: <https://doi.org/10.3115/1072228.1072262>.
- Karin Harbusch and Gerard Kempen. Clausal coordinate ellipsis in German: The TIGER treebank as a source of evidence. In *Proceedings of NODALIDA 2007 - Sixteenth Nordic Conference of Computational Linguistics*, Tartu, Estonia, 2007.
- Karin Harbusch and Gerard Kempen. Coordinate Ellipsis and its Varieties in Spoken German: A Study with the TüBa-D/S Treebank of the VERBMOBIL Corpus. In *Proceedings of TLT8 - 8th International Workshop on Treebanks and Linguistic Theories*, Milano, Italy, 2009.
- Karin Harbusch and Gerard Kempen. Automatic online writing support for 12 learners of german through output monitoring by a natural-language paraphrase generator: Karin harbusch and gerard kempen. In *WorldCALL*, pages 145–160. Routledge, 2011.
- Karin Harbusch and Ina Steinmetz. A computer-assisted writing tool for an extended variety of leichte sprache (easy-to-read german). *Frontiers in Communication*, 6, 2022. ISSN 2297-900X. doi: 10.3389/fcomm.2021.689009. URL <https://www.frontiersin.org/article/10.3389/fcomm.2021.689009>.
- Karin Harbusch, Camiel van Breugel, Ulrich Koch, and Gerard Kempen. Interactive sentence combining and paraphrasing in support of integrated writing and grammar instruction: A new application area for natural sentence generators. In *Proceedings of ENLG 2007 - 11th European Workshop on Natural Language Generation*. Dagstuhl, Germany, 2007.
- Karin Harbusch, Johannes Härtel, and Christel J. Cameran. Automatic feedback generation for grammatical errors in German as second language learners focused on the learner's personal acquisition level. In *Proceedings of EUROCALL 2014 - Call Design: Principles and Practice*. Groningen, The Netherlands, 2014.
- Shawn Lawton Henry. *Just Ask: Integrating Accessibility Throughout Design*. Shawn Lawton Henry (see www.uiaccess.com/JustAsk/), Madison, WI, USA, 2007. URL <http://www.uiaccess.com/justask/>.

- Troy Hicks. *The digital writing workshop*. Heinemann, Portsmouth, NH, USA, 2009.
- Marianne Hirschberg and Christian Lindmeier. Der begriff „inklusion“ - ein grundsatz der menschenrechte und seine bedeutung für die erwachsenenbildung. In *Zugänge zu Inklusion. Erwachsenenbildung, Behindertenpädagogik und Soziologie im Dialog*, pages 39–52. 2013. ISBN 978-3-7639-5107-9; 978-3-7639-5108-6. doi: <https://doi.org/10.25656/01:8573>.
- Andreas Holzinger. Usability engineering methods for software developers. *Communications of the ACM*, 48(1):71–74, 2005.
- Eduard H. Hovy. Planning coherent multisentential text. In *Proceedings of the 26th Annual Meeting on Association for Computational Linguistics, ACL '88*, page 163–169, USA, 1988. Association for Computational Linguistics. doi: [10.3115/982023.982043](https://doi.org/10.3115/982023.982043). URL <https://doi.org/10.3115/982023.982043>.
- Inclusion Europe. Informationen für alle – Europäische Regeln, wie man Informationen leicht lesbar und leicht verständlich macht. https://www.inclusion-europe.eu/wp-content/uploads/2017/06/DE_Information_for_all.pdf, 2009. Accessed 29-March-2021.
- Inclusion Europe. Inclusion europe - history, 2022. [Online] Accessed 09-May-2022.
- Jabbla. Steps to language – using the grammar pop-up page level 3. <https://www.jabbla.com/en/tutorials/steps-to-language-using-the-grammar-pop-up-page-level-3/>, 2022a. [Online] Accessed 11-January-2022.
- Jabbla. Steps to language – alphabet page in level 1. <https://www.jabbla.com/en/tutorials/steps-to-language-the-alphabet-page-in-level-1/>, 2022b. [Online] Accessed 11-January-2022.
- Jabbla. Steps to language – turning dynamic labels on or off. <https://www.jabbla.com/en/tutorials/steps-to-language-turning-dynamic-labels-on-or-off/>, 2022. [Online] Accessed 11-January-2022.
- Jabbla. Mind express - jabbla. <https://www.jabbla.com/en/mind-express>, 2022a. [Online] Accessed 11-January-2022.

- Jabbla. Mind express overview 1 – 2. <https://www.jabbla.com/en/tutorials/mind-express-overview-1-2/>, 2022b. [Online] Accessed 11-January-2022.
- Jabbla. Score 2.0 - us version. <https://mindexpress.jabbla.com/en/download/score-2-0-us-version/>, 2022c. [Online] Accessed 11-January-2022.
- Roxanna Mayer Johnson. *The picture communication symbols*, volume 1. Mayer-Johnson Company, 1987.
- Gerard Kempen and Karin Harbusch. Performance Grammar: A declarative definition. *Language and Computers*, 45(1):148–162, 2002.
- Gerard Kempen and Karin Harbusch. Verb-second word order after German weil ‘because’: Psycholinguistic theory from corpuslinguistic data. *Glossa: A journal of general linguistics*, 1(1): 3:1–32, 2016. doi: <http://dx.doi.org/10.5334/gjgl.46>.
- Gerard Kempen and Karin Harbusch. Mutual attraction between high-frequency verbs and clause types with finite verbs in early positions: Corpus evidence from spoken English, Dutch, and German. *Language, Cognition and Neuroscience*, 34(9):1140–1151, 2019. doi: <https://doi.org/10.1080/23273798.2019.1642498>. URL <https://doi.org/10.1080/23273798.2019.1642498>.
- Phillipe B. Kruchten. The 4+1 view model of architecture. *IEEE Software*, 12(6): 42–50, 1995. doi: 10.1109/52.469759.
- Tobias Kuhn. A survey and classification of Controlled Natural Languages. *Computational Linguistics*, 40(1):121–170, March 2014. doi: 10.1162/COLI_a_00168. URL <https://www.aclweb.org/anthology/J14-1005>.
- Giulio Lancioni, Nirbhay Singh, Mark O’Reilly, and Gloria Alberti. Assistive technology to support communication in individuals with neurodevelopmental disorders. *Current Developmental Disorders Reports*, 6(3):126–130, 09 2019. doi: 10.1007/s40474-019-00165-x.
- Bettina Laugwitz, Martin Schrepp, and Theo Held. Konstruktion eines Fragebogens zur Messung der User Experience von Softwareprodukten. [[construction of a questionnaire for the measurement of user experience of software products]. In A.M. Heinecke & H. Paul (Eds.): *Mensch & Computer – Mensch und Computer im Strukturwandel*, pages 125–134. Oldenbourg Verlag, 2006.

- Bettina Laugwitz, Theo Held, and Martin Schrepp. Construction and evaluation of a user experience questionnaire. In *Symposium of the Austrian HCI and usability engineering group*, pages 63–76. Springer, 2008.
- Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. *Chapter 16 - Working with research participants with disabilities*. Morgan Kaufmann, an imprint of Elsevier, Cambridge, MA, USA, 2017. ISBN 978-0-12-805390-4. doi: <https://doi.org/10.1016/B978-0-12-805390-4.00016-9>. URL <https://www.sciencedirect.com/science/article/pii/B9780128053904000169>.
- Dong-Seok Lee and Wan Chul Yoon. Quantitative results assessing design issues of selection support menus. *International Journal of Industrial Ergonomics*, 33:41–52, 2004. doi: <https://doi.org/10.1016/j.ergon.2003.07.004>.
- Christian Lieske and Melanie Siegel. Verstehen leicht gemacht. *Technische Kommunikation*, 1:44–49, 2014.
- LIFETool. ICH-Bücher in der Unterstützten Kommunikation - analog & digital. <https://www.lifetool.at/beraten/rat-tat/uk-tipp/uk-tipps/2016/2016-11-ich-buch/>, 2022. [Online] Accessed 06-November-2022.
- Janice Light and David McNaughton. The changing face of augmentative and alternative communication: Past, present, and future challenges. *Augmentative and Alternative Communication*, 28:197–204, 2012. doi: <https://doi.org/10.3109/07434618.2012.737024>.
- Janice Light, David McNaughton, and Jessica Caron. New and emerging AAC technology supports for children with complex communication needs and their communication partners: State of the science and future research directions. *Augmentative and Alternative Communication*, 35(1):26–41, 2019. doi: <https://doi.org/10.1080/07434618.2018.1557251>.
- Camilla Lindholm and Ulla Vanhatalo. *Handbook of easy languages in Europe*. Frank & Timme, Berlin, Germany, 2021.
- Cordula Löffler. Leichte Sprache als Chance zur gesellschaftlichen Teilhabe funktionaler Analphabeten. In *Didaktik Deutsch*, volume 38, pages 17–23, 2015.
- Stephanie M Lukin and Marilyn A Walker. A narrative sentence planner and structurer for domain independent, parameterizable storytelling. *Dialogue & Discourse*, 10(1):34–86, 2019. doi: 10.5087/dad.2019.103. URL <https://11library.net/document/y692d4gy-narrative->

sentence-planner-structurer-domain-independent-parameterizable-storytelling.html.

Christiane Maaß and Ursula Bredel. *Leichte Sprache: Theoretische Grundlagen Orientierung für die Praxis*. Bibliographisches Institut, Berlin, 2016. ISBN 9783411911783. URL <https://books.google.de/books?id=RD4pDAAAQBAJ>.

Christiane Maaß, Isabel Rink, and Christiane Zehrer. Leichte Sprache in der Sprach- und übersetzungswissenschaft. *Susanne Jekat, Heike Elisabeth Jüngst, Klaus Schubert & Claudia Villiger (Hg.), Sprache barrierefrei gestalten: Perspektiven aus der Angewandten Linguistik*, 53:86, 2014.

William C. Mann and Sandra A. Thompson. Rhetorical Structure Theory: Toward a theory of text organization. *Text - Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281, 1988. doi: 10.1515/text.1.1988.8.3.243. URL <https://doi.org/10.1515/text.1.1988.8.3.243>.

Mensch zuerst. Mensch zuerst - Netzwerk People First Deutschland e.V. - Wer sind wir? <http://www.menschzuerst.de/pages/startseite/wer-sind-wir/verein.php>, 2022. [Online] Accessed 09-May-2022.

Netzwerk Leichte Sprache. Die Regeln für Leichte Sprache. https://www.leichte-sprache.org/wp-content/uploads/2017/11/Regeln_Leichte_Sprache.pdf, 2013. [Online] Accessed 29-March-2021.

Netzwerk Leichte Sprache. Netzwerk Leichte Sprache - Die Geschichte der Leichten Sprache. <https://www.leichte-sprache.org/der-verein/die-geschichte/>, 2022. [Online] Accessed 09-May-2022.

Alan F Newell, Peter Gregor, Maggie Morgan, Graham Pullin, and Catriona Macaulay. User-sensitive inclusive design. *Universal Access in the Information Society*, 10(3):235–243, 2011.

James W. Ney. A short history of sentence combining: its limitations and use. *English Education*, 11(3):169–177, 1980. ISSN 00078204. URL <http://www.jstor.org/stable/40172300>.

Julius Njanji and Shawulu Nggada. Disability-aware software engineering for improved system accessibility and usability. *International Journal of Software Engineering and Its Applications*, 5:47–62, 07 2011.

- Jakob Nielsen. Usability Engineering at a Discount. In *Proceedings of the Third International Conference on Human-Computer Interaction on Designing and Using Human-Computer Interfaces and Knowledge Based Systems (2nd Ed.)*, page 394–401, USA, 1989. Elsevier Science Inc. ISBN 044488078X.
- Richard Nordquist. An introduction to sentence combining. ThoughtCo.: <https://www.thoughtco.com/an-introduction-to-sentence-combining-1692421>, 2018. Accessed 18-March-2021.
- Donald A Norman and Stephen W Draper. User centered system design: New perspectives on human-computer interaction. 1986.
- Nathalie D. Nüssli. Übersetzen in die Leichte Sprache: Übersetzungsprobleme, Übersetzungslösungen und Auswirkungen auf das Textverständnis von Menschen mit Downsyndrom: am Beispiel von Texten zum Thema Gesundheit. *ZHAW Züricher Hochschule für Angewandte Wissenschaften, Zürich, Switzerland*, 2019. doi: <https://doi.org/10.21256/zhaw-3344>.
- Charles K. Ogden. *Basic English: A general introduction with rules and grammar*, volume 1. Paul Treber&Co., London, UK, 1930.
- Daniel W. Otter, Julian R. Medina, and Jugal K. Kalita. A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2):604–624, 2021. doi: 10.1109/TNNLS.2020.2979670.
- H. Portal, G. Schmidt, R. Crespo Fernandez, B. Marcondes, M. Sverepa, V. Dragicevic, and D. Lysaght. Neglect and discrimination. Multiplied. How Covid-19 affected the rights of people with intellectual disabilities and their families. Inclusion Europe: <https://www.inclusion-europe.eu/wp-content/uploads/2020/11/COVID-report-Final.pdf>, 2021. Accessed 09-March-2021.
- Daniel M. Pottmann. „Leichte Sprache und Einfache Sprache” – German plain language and teaching DaF German as a foreign language. *Studia Linguistica*, 38:81–94, 2019. URL <https://wuwr.pl/slin/article/download/4767/10271>.
- Anke Reichardt, Norbert Kruse, and Frank Lipowsky. Textüberarbeitung mit Schreibkonferenz oder Textlupe. Zum Einfluss der Schreibumgebung auf die Qualität von schülertexten. *Didaktik Deutsch*, 19(36):64–85, 2014. ISSN 1431-4355.

- Robert Reid, Torri Ortiz Lienemann, and Jessica L. Hagaman. *Strategy instruction for students with learning disabilities*. The Guilford Press, New York, NY, USA., 2013.
- Marga Reis. „Weil-V2“-Sätze und (k)ein Ende? Anmerkungen zur Analyse von Antomo & Steinbach (2010). *Zeitschrift für Sprachwissenschaft*, 32(2):221–262, 2013. doi: <https://doi.org/10.1515/zfs-2013-0008>.
- Ehud Reiter and Robert Dale. *Building Natural Language Generation Systems*. Studies in Natural Language Processing. Cambridge University Press, 2000. doi: [10.1017/CBO9780511519857](https://doi.org/10.1017/CBO9780511519857).
- Ehud Reiter, Ross Turner, Norman Alm, Rolf Black, Martin Dempster, and Annalu Waller. Using nlg to help language-impaired users tell stories and participate in social dialogues. *Proceedings of the 12th European Workshop on Natural Language Generation, ENLG 2009*, 01 2009. doi: [10.3115/1610195.1610196](https://doi.org/10.3115/1610195.1610196).
- Peter Rödler. Totale Institution – Die Renaissance der geschlossenen Unterbringung im Zuge der Corona Pandemie? *Behindertenpädagogik*, 1(4):345–358, 2020. doi: <https://doi.org/10.30820/0341-7301-2020-4-345>.
- John R. Ross. *Constraints on Variables in Syntax*. PhD thesis, MIT, Cambridge, MA/UASA (published as: *Infinite Syntax!* Ablex, Norwood 1987, ISBN 0-89391-042-2), 1967.
- Bruce Saddler and Jennifer Preschern. Improving sentence writing ability through sentence combining practice. *Abstracts from Teaching Exceptional Children (TEC)*, 39, 2007. doi: [10.1177/004005990703900301](https://doi.org/10.1177/004005990703900301).
- Melissa Schmidt. Desperately seeking simplicity: How young adults with cognitive disabilities and their families adopt assistive technologies. volume 2, pages 1143–1152, 01 2006. doi: [10.1145/1124772.1124943](https://doi.org/10.1145/1124772.1124943).
- Rico Sennrich, Martin Volk, and Gerold Schneider. Exploiting Synergies Between Open Resources for German Dependency Parsing, POS-tagging, and Morphological Analysis. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 601–609, Hissar, Bulgaria, September 2013. INCOMA Ltd. Shoumen, BULGARIA. URL <https://www.aclweb.org/anthology/R13-1079>.
- Sean J. Smith, K. Alisa Lowrey, Amber L. Rowland, and Bruce Frey. Effective Technology Supported Writing Strategies for Learners With Disabilities. *Inclusion*, 8(1):58–73, 03 2020. ISSN 2326-6988. doi: [10.1352/2326-6988-8.1.58](https://doi.org/10.1352/2326-6988-8.1.58). URL <https://doi.org/10.1352/2326-6988-8.1.58>.

- Rosmary Stegmann, Heike Telljohann, and Erhard W. Hinrichs. Stylebook for the German Treebank in Verbmobil. Technical report, DFKI Report 239, Saarbrücken, Germany, 2000.
- Ina Steinmetz and Karin Harbusch. Enabling fast and correct typing in ‘Leichte Sprache’ (easy language). In *Proceedings of the The Fourth Widening Natural Language Processing Workshop*, pages 64–67, Seattle, CA, USA, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.winlp-1.17. URL <https://www.aclweb.org/anthology/2020.winlp-1.17>.
- Ina Steinmetz and Karin Harbusch. Easytalk: An assistive text-writing system for Leichte Sprache (easy-to-read german). In *Proceedings of Communication Matters International AAC Conference*, Leeds, UK, 2021a. URL https://userpages.uni-koblenz.de/~harbusch/20211207_CM2021_final.pdf.
- Ina Steinmetz and Karin Harbusch. Easytalk: A digital writer’s workshop for Leichte Sprache (easy-to-read German). In *The European Conference on Education 2021: Official Conference Proceedings*, 2021b. doi: <https://doi.org/10.22492/issn.2188-1162.2021.32>.
- Ina Steinmetz and Karin Harbusch. A text-writing system for Easy-to-Read German evaluated with low-literate users with cognitive impairment. In *Proceedings of the First Workshop on Intelligent and Interactive Writing Assistants (In2Writing 2022)*, pages 27–38, Dublin, Ireland, 2022. doi: 10.18653/v1/2022.in2writing-1.4.
- Julia Suter, Sarah Ebling, and Martin Volk. Rule-based automatic text simplification for German. In *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, pages 279–287, Zürich, Switzerland, 2016. URL <https://www.zora.uzh.ch/id/eprint/128601/1/suter-et-al-2016.pdf>.
- Heike Telljohann, Erhard W. Hinrichs, Sandra Kübler, Heike Zinsmeister, and Kathrin Beck. Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z). Technical report, Technical report of the Seminar für Sprachwissenschaft, Universität Tübingen, Tübingen, Germany, 2009. URL <https://sfs.uni-tuebingen.de/resources/tuebadz-stylebook-0911.pdf>.
- David Temperley. Rare constructions are more often sentence-initial. *Cognitive Science*, 43, e12714, 2019. doi: 10.1111/cogs.12714.

Nava Tintarev, Ehud Reiter, Rolf Black, Annalu Waller, and Joe Reddington. Personal storytelling: Using natural language generation for children with complex communication needs, in the wild. . . . *International Journal of Human-Computer Studies*, 92-93:1–16, 2016. ISSN 1071-5819. doi: <https://doi.org/10.1016/j.ijhcs.2016.04.005>.

tobii dynamox. Snap™ + core first® user's manual. http://tdvox.web-downloads.s3.amazonaws.com/Snap/documents/TobiiDynamox_SnapCoreFirst_UsersManual_v1-5_en-US_WEB.pdf, 2018. Accessed 19-January-2022.

tobii dynamox. TD Snap. <https://de.tobiidynamox.com/pages/td-snap-core-first>, 2022. [Online] Accessed 11-January-2022.

Wolfgang Wahlster. *Verbmobil: Foundations of speech-to-speech translation*. Springer, Berlin, Germany, 2000. doi: 10.1007/978-3-662-04230-4.

Annalu Waller. Telling tales: Unlocking the potential of AAC technologies. *International journal of language & communication disorders*, 54(2):159–169, 2019. doi: 10.1111/1460-6984.12449.

Bettina Zurstrassen. Inklusion durch Leichte Sprache? Eine kritische Einschätzung. In *Didaktik der inklusiven politischen Bildung*. 2015.