Diss. ETH No. 20009

Modeling and Evaluation of Computer-Assisted Spelling Learning in Dyslexic Children

A dissertation submitted to
ETH Zurich

for the Degree of
Doctor of Sciences

presented by
Gian-Marco Baschera
Dipl. rech. Wiss., ETH Zurich, Switzerland
born 22 April 1982
citizen of Switzerland

accepted on the recommendation of
Prof. Dr. Markus Gross, examiner
Prof. Dr. Lutz Jäncke, co-examiner
Prof. Dr. Joachim M. Buhmann, co-examiner

2011

Abstract

Dyslexia is a learning disability, which impairs the development of reading and writing skills. To support children with dyslexia in spelling learning, the multimodal therapy software Dybuster recodes the spelling information of words in visual and auditory representations. A main factor of the efficacy of such a computer-assisted therapy approach is the adequacy of the presented material for a given student. Like human tutors, intelligent tutoring systems have to adapt the training to the student's individual pace and needs.

In this thesis we present one closed loop in the data-driven development of an intelligent tutoring system: from user data analysis over student modeling to the evaluation of incorporated improvements. Based on the log file data of a large-scale user study of Dybuster, we analyze the learning and forgetting processes of children. We introduce a novel error taxonomy and a spelling knowledge representation to allow for a selection of appropriate remediation actions and for an adaptation of the training to the student-specific spelling difficulties. Based on the gained insights and developed models we extend the original Dybuster version with phoneme-based enhancements: (1) an improved word selection controller; (2) a textural code representing phonological information. This enhanced software is then evaluated in a second user study.

In addition to the appropriateness of the presented material, the student's current attentional state and attitude toward the training is a crucial factor for the effectiveness of a therapy. We first present a systematic approach to incorporate domain knowledge about affective dynamics into feature processing for affective modeling. We demonstrate how the method significantly improves the predictive power of features. Then, by quantitatively relating the processed input behavior and learning, a model of engagement is inferred from student input data, representing the student's short-term variations in attention.

Zusammenfassung

Dyslexie bezeichnet eine Lernschwäche, welche den Erwerb der Schriftsprache beeinträchtigt. Um dyslexische Kinder beim Schreibenlernen zu unterstützen, recodiert die multi-modale Therapiesoftware Dybuster die Information der Rechtschreibung in visuelle und auditorische Repräsentationen. Ein zentraler Faktor für die Wirksamkeit einer solchen Computer-basierten Therapie ist die Angemessenheit des Lernmaterials. Vergleichbar mit menschlichen Betreuern muss eine intelligente Lernumgebung Inhalt und Geschwindigkeit des Trainings an die Bedürfnisse eines Studenten anpassen.

In dieser Dissertation beschreiben wir einen kompletten Zyklus in der datengestützten Entwicklung einer intelligenten Lernumgebung: Von der Analyse gesammelter Benutzerdaten, über die Modellierung eines Studenten, bis hin zur Evaluation der eingearbeiteten Verbesserungen. Basierend auf den Logfiles einer grossangelegten Dybuster Benutzerstudie untersuchen wir die Lern- und Vergessprozesse von Kindern. Wir präsentieren eine neue Fehlertaxonomie und ein darauf basierendes Rechtschreibwissen-Model, welches eine Anpassung des Trainingsinhaltes an die Bedürfnisse individueller Kinder ermöglicht. Basierend auf den gewonnen Erkenntnissen und entwickelten Modellen wird die ursprüngliche Dybusterversion mit Phonem-basierten Elementen erweitert. Diese beinhalten einerseits einen verbesserten Wortselektionsmechanismus und andererseits einen zusätzlichen Texturcode, welcher die phonologische Struktur eines Wortes repräsentiert. Die Phonem-basierten Erweiterungen werden dann in einer zweiten Benutzerstudie evaluiert.

Ein ausschlaggebender Faktor für die Effektivität einer Therapie ist, neben der Angemessenheit des Lernmaterials, die Einstellung und Aufmerksamkeit des Studenten. Wir beschreiben einen systematischen Ansatz zur Verarbeitung von extrahierten Merkmalen für die Entwicklung affektiver Modelle, basierend auf Grundlagenwissen über die Dynamik affektiver Zustände. Wir demonstrieren, wie die Methode die Voraussagekraft der extrahierten Merkmale signifikant erhöht. Durch das Inverbindungsetzen von Eingabeverhalten und Lernen leiten wir aus den Benutzerdaten ein Aufmerksamkeitsmodell ab, welches die kurzfristigen Veränderungen der Zustände eines Studenten repräsentiert.

Acknowledgments

First of all, I would like to sincerely thank my advisor Prof. Markus Gross. He's the father of the Dybuster project and has originated the multi-modal concept of the spelling software. His involvement is based on both academic as well as personal interest in an effective dyslexia therapy. I'm grateful for his great trust and offered freedom in exploring new fields at the Computer Graphics Laboratory. His unconditional support and guidance based on his strong scientific intuition was invaluable during my Ph.D.

Furthermore, I would like to convey my gratitude to the collaborators at the Institute of Neuropsychology of the University of Zürich. My thank goes to Prof. Lutz Jäncke, Dr. Martin Meyer and especially to Dr. Monika Kast, for the fruitful and friendly team work. I really enjoyed both the scientific and personal elements of this interdisciplinary collaboration. It was also a pleasure to work with Christian Vögeli, Miriam Flühler and all the long- and short-term coworkers at the Dybuster AG. I wish them all the best for their future with Dybuster.

I'm deeply thankful to the collaborators of the Machine Learning Group at ETH Zürich for their enduring support. Especially, Prof. Joachim Buhmann, Alberto Giovanni Busetto and Kay Brodersen offered great help for many of the technical aspects of this thesis.

Many thanks go also to my coworkers at CGL for creating such an enjoyable work environment. In particular, I want to thank Peter Kaufmann - who stayed my office mate during all our relocations - and Sebastian Martin for their work and non-work related inspiring discussions. You guys were a great support during the good as well as the sometimes harder moments of the Ph.D. I further want to mention Dr. Alex Hornung who spent much time and effort to provide me with help and guidance during my work.

Last but not least, I want to thank my family and friends, especially my parents, for their support during my Ph.D. My gratitude also goes to Dr. Martin Krafft for the challenging discussions and the demonstration of novel points of view by special means. Finally, I would like to thank my girlfriend Selina Müller for her enduring support and understanding. This work would not have been possible without you.

This thesis was funded by the CTI-grant 8970.1.

Contents

CHAPTER 1

Introduction

Reading and writing skills are essential in modern societies where information is commonly provided by written media. In the case of dyslexia the acquisition of these cultural techniques is impaired. Documents composed by individuals with dyslexia exhibit significantly higher error rates, which influence the judgment about the quality of writing and distracts a reader from the message.

The work presented in this thesis is concerned with the efficacy of computer-assisted therapy approaches for the reading and writing disability dyslexia. It spans several areas of interest, including elements of psychology, linguistics, and student modeling. In general, we address the question of how tutoring systems can identify and represent the knowledge and affective states of a student and adapt the training correspondingly. The question is approached based on the data collected in a large-scale user study of Dybuster, a multimodal spelling software for dyslexic children. The data-driven investigations lead to a novel spelling knowledge representation and to models of long-term and short-term variations of a student. The gained insights and developed student representations are incorporated into an enhanced software version, which is evaluated in a second user study.

Section 1.1 describes the relationship between the different components of the thesis and the motivation for the presented work. In Section 1.2 we

summarize the main contributions and Section 1.3 gives an outline of the rest of the thesis.

1.1 Rational

This section gives a brief introduction in the main areas covered by the presented work: starting from fundamental concepts of language, covering learning disorders, and finally reaching intelligent tutoring systems and their evaluation. It describes the association between the different fields of research and motivates the work presented in this thesis.

1.1.1 Language and Dyslexia

Humans are unique in the sophisticated way they communicate with language. The ability of humans to transfer concepts and ideas through speech and writing is unrivaled in known species. Hockett's list of essential features to describe human language [Hoc60] contains two main elements: (1) productivity, i.e., the possibility of creating and understanding completely novel messages; (2) duality, i.e, that a large number of meaningful elements can be made up of a conveniently small number of independently meaningless elements. The latter one can easily be noticed in the concept of written language, where meaningful words can be made up of meaningless letters.

The writing of Western world languages has developed over thousands of years and is still developing. In the beginning, literal drawings representing concrete events and objects were used, which gradually changed to include abstract concepts like a number of days represented by symbolic drawings of the sun. Approximately 3000 years ago the Phonecians started to combine symbols to construct longer words. Later, descending from this non-pictographic consonantal Phoenician alphabet, the Greek invented the first alphabet, in the narrow sense that each consonant and vowel is represented by an individual symbol. The Greek alphabet has given rise to many other alphabets, including the most widely used Latin alphabet. Originally, words were spelled as they sounded until the advent of printing and the increasing literacy inspired people to standardize the writing. Scribes started to spell words irregularly to ease the flow of writing and to indicate the historical origin of words in Latin or Greek. After several hundred years of language evolution German, as most Western languages, ended up with many spelling irregularities and a non-bijective mapping between sounds and symbols.

There can be no doubt that writing is a fundamental skill for human life in modern civilization. Although the message to be communicated is more important than the correct spelling, the competence in spelling has a high influence on the quality of written work [Mac99]. Spelling errors influence judgments that others make about overall quality of writing, distract readers from the message, and in extreme cases render the message incomprehensible. Perhaps even more important: problems with spelling interfere with higher writing processes and affect the quality of writing. The orthographic depth of Western languages, i.e., the non-bijective correspondence between phonemes (sounds) and graphemes (symbols), raises difficulties for children learning to spell. The process of spelling includes several additional challenges, from precise hearing to the mechanics of writing of words.

Correct spelling is especially hard to learn for dyslexic children. Developmental dyslexia is characterized through low reading and writing skills, in spite of an average or above average IQ, adequate education and inconspicuous social background [WHO93]. Dyslexia occurs predominantly in Western world languages, including English, French, German, or Spanish. It is estimated that about 5-7% of the Western world population suffers from minor or major forms of dyslexia [Rei89]. The definition of dyslexia is purely symptom-oriented, and does not describe the causes of the disorder. These are strongly debated and discussed in more detail in Chapter 2. The diversity of hypotheses is partly based on the differing characteristics of spelling difficulties of dyslexic children. They range from visual (*'d'-'b'*) and auditory confusions (*'n'-'m'*) to difficulties in the phoneme-grapheme matching process (/f/: *'f'-'v'*). However, the common denominator of most theories is the presence of a phonological processing deficit in dyslexic subjects.

1.1.2 Therapy

Since dyslexia is widespread in the Western world, there exist various remediation approaches. The therapies are commonly based on a specific hypothesis of dyslexia. For example, the well-known Davis therapy [Dav85] assumes that dyslexic individuals experience perceptual disorientations in the senses of time, vision, hearing, and coordination. He tries to resolve the factors that trigger the disorientation in a multi-sensory treatment, e.g., by modeling letters with clay or associating pictures with words.

Due to todays prevalence of computers, there arose many computer-assisted therapy approaches. The advantage of computer-assisted therapies is that they combine recreational and didactic goals. The game-like learning envi-

ronment of successful educational games leads to an increased interest and motivation of the student to acquire knowledge.

Gross and Vögeli developed such a computer-assisted, multi-modal spelling training for dyslexic children, called Dybuster. The entire framework is based on the concepts of information theory and multi-modal learning [GV07]. The central idea of the training software is to recode the spelling of words into a multi-modal representation using a set of codes. These codes reroute information about letters and syllables through multiple perceptual cues, including topological, color and shape, as well as auditory representations. These multi-modal learning aids are employed in the different training games. In the main game, the children repetitively enter dictated words on the keyboard, supported by the multi-modal representations.

This therapy software and the log file data collected in a Dybuster user study are the basis for the presented work and described in more detail in Chapter 3 and 4. During the progression of the work, the software is extended with phoneme-based enhancements to account for the gradually gained insights presented in this thesis.

1.1.3 Student Modeling

The success of every therapy approach is dependent on the adequacy of the training for a given student. School teachers, tutors or therapist are trained to identify the student's needs and adapt the therapy accordingly. A computer-assisted training software needs the ability to provide a comparable adaptivity. Students should be able to learn in their own way without having to follow already made up tracks and pre-made levels [BB07]. Such an intelligent tutoring system has to analyze the student input, build a representation of the student and choose remediation actions correspondingly.

To account for the diversity of dyslexic spelling difficulties, committed errors needs to be investigated according to a fine-grained error taxonomy. In Chapter 5 we present a novel phoneme-based taxonomy for spelling errors with corresponding error generating rules, called mal-rules. These describe committed errors on a letter and phoneme level and are designed for the specific setting of recent spelling software. Based on this set of mal-rules, the strengths and weaknesses in spelling of a child are modeled by a Poisson-based student knowledge representation described in Chapter 6. The student model provides the tutoring system with a word difficulty measure and a classification of committed errors. These allow for an adaptation of the word selection to individual students and for an optimal scheduling of repetition prompts.

In addition to the long-term modeling of spelling knowledge, a tutoring system should also keep track of the attentional state of a student, since attention is essential for learning [AH02]. However, the variety of influences acting on observable input behavior results in significant noise levels and non-i.i.d. data, which makes the modeling of affective dynamics a challenging task. In Chapter 7 we present a novel method which allows for a systematic identification of an optimal processing of features for affective modeling. By relating processed input behavior and learning we develop a model of engagement, representing the short-term variation of focused and receptive states, as described in Chapter 8. Based on the information provided by such affective models, an intelligent tutoring system is able to respond to a loss of attention by switching between different games, by including attention-capturing elements, or by establishing flexible training schedules.

1.1.4 Evaluation

Model evidence and intuitive adequacy of results and conclusions extracted from developed models provide an indication for their appropriateness. However, the goal of the modeling process is to finally improve a tutoring system and allow for an adaptation to the student. Therefore, the focus of evaluation lies on the applicability of developed models. Based on the novel insights gained from the analyses of the collected user data, the Dybuster therapy software is extended with phoneme-based enhancements. These include an additional textural code representing phonological information of a word, and an improved word selection controller, which relies on the error prediction and classification of the presented phoneme-based spelling knowledge representation. Since the model of engagement has not been developed at this stage, it is not incorporated in the enhanced software version.

The phoneme-based enhancements are evaluated in a second user study. The error classification enables a comparison of the spelling progress on individual error categories. This categorization allows us to investigate specific difficulties, such as the mechanical typing process (typos) or dyslexic spelling difficulties (e.g., phoneme-grapheme matching). The spelling progress of children working with the original and the enhanced software version is compared by means of learning curves [NR81], as described in Chapter 10. In addition, we investigate the influence of different cognitive factors on the learning progress, based on data collected in the second user study. These factors include the indication of dyslexia, memory performances, and attention functions, which all are considered to be related to the process of learning.

1.2 Principal Contributions

In the following we summarize the principle contributions of the work presented in this thesis:

- *Error model:* We introduce an error taxonomy and corresponding phoneme-based mal-rules to describe errors in isolated word spelling. The presented mal-rules result from the investigation of the collected user data and are the first designed for the special setting of recent spelling software: immediate feedback on committed errors, which restricts the error analysis on the input up to the error letter.
- *Spelling knowledge representation:* We present an inference algorithm for perturbation models based on a Poisson regression. The algorithm is designed to handle unclassified input with multiple errors described by our set of mal-rules. This knowledge representation provides an intelligent tutoring system with local and global information about a student, such as error classification (local) and prediction of further performance (global).
- *Phoneme-based enhancements:* We introduce two enhancements, which are implemented in the improved Dybuster version: (1) an additional textural code representing the phonological structure of a word; (2) an improved word selection controller, based on the gained insights and the information provided by the novel spelling knowledge representation.
- *Evaluation:* The phoneme-based enhancements are employed in a second user study. This enables an evaluation of the spelling progress improvements induced by the enhanced software version. Additionally, we are able to present empirical evidence for the influence of different cognitive factors on the learning behavior in computer-assisted learning.
- *Affective feature processing:* We present a systematic approach to incorporate domain knowledge in feature processing for affective modeling. The presented method employs time scale separation and normality-maximizing scaling and is implemented in a modular affective modeling framework. We show how the processing of features significantly increases the predictive power of the extracted input behavior.
- *Model of engagement:* We introduce a model of engagement dynamics in spelling learning. By quantitatively relating processed input

behavior to learning, our model enables a prediction of focused and receptive states, as well as of forgetting, without any additional assessment of affect.

1.3 Thesis Outline

The thesis is structured in three main parts: Part I: Data, Part II: Modeling, and Part III: Evaluation.

In Part I we first describe the dyslexia therapy software Dybuster in detail (Chapter 3). We introduce the general concepts, the different games as well as the phoneme-based enhancements of the spelling software. Then, Chapter 4 gives the details of the two user studies. It describes the data collected during the studies conducted in 2006 and 2008.

In Part II we present the student models developed based on the available user data: first, we introduce our error model (Chapter 5) with corresponding mal-rules; second, Chapter 6 describes the student knowledge representation; third, we present our systematic approach to affective modeling (Chapter 7) and the final model of enhancement (Chapter 8).

In Part III we describe how the insights gained from the student models are incorporated into the enhanced Dybuster version (Chapter 9) and present the results from the learning progress comparisons (Chapter 10).

1.4 Publications

In the context of this thesis, the following publications have been published.

G.M. BASCHERA and M. GROSS. A Phoneme-Based Student Model for Adaptive Spelling Training. *Proceedings of the 14th International Conference on Artificial Intelligence in Education*, Brighton, UK, 2009.

This paper sketches the error model, the phoneme-based mal-rules and the student model representation.

G.M. BASCHERA and M. GROSS. Dybuster - Ein adaptives, multi-modales Therapiespiel für Legastheniker. *Spielend Lernen*, Rostock, DE, 2010.

In this publication we describe the game like learning environment of Dybuster. The motivation enhancing elements, such as the physically animated graph and the virtual shop are presented.

G.M. BASCHERA and M. GROSS. Poisson-Based Inference for Perturbation Models in Adaptive Spelling Training. *International Journal of Artificial Intelligence in Education*, 2010.

This publication presents an extended version of the student knowledge representation. It describes error model, mal-rules and Poisson-based inference in detail and provides a verification of the error prediction and classification.

M. KAST, G.M. BASCHERA, M. GROSS, L. JÄNCKE and M. MEYER. Computer-based Learning of Spelling Skills in Children with and without Dyslexia. *Annals of Dyslexia*, 2011.

In this paper we present the evaluation of the phoneme-based enhancements and the investigation of the influence of different cognitive factors on the learning progress. Please note that M. Kast and G.M. Baschera contributed equally to the manuscript.

G.M. BASCHERA, A.G. BUSETTO, S. KLINGLER, J.M. BUHMANN and M. GROSS. Modeling Engagement Dynamics in Spelling Learning. *Proceedings of the 15th International Conference on Artificial Intelligence in Education*, Auckland, NZ, 2011.

This publication presents our systematic approach to affective modeling and the final model of engagement. The paper was awarded with the Best Student Paper Award.

CHAPTER 2

Related Work

This chapter describes the related work in the different areas of research influencing this thesis. First, it covers work in the field of language and learning. This includes general models of language processing, learning deficits and relevant factors for learning, as well as specific therapy approaches and error taxonomies for spelling. Second, it describes the related work in student modeling, ranging from general concepts of modeling students to specific applications for spelling and affective modeling.

2.1 Language and Learning

Investigations into the difficulties of processing written language have been focused on the "input" side of the problem, namely reading. Less attention has been paid on the "output" aspect of written language, namely writing [Kas11]. However, models of reading have often been applied to spelling because of the similarities between the two cognitive processes; spelling is summarized as only the reverse of reading, at least to some extent [BAZ+99]. One of the most popular and influential theories of word processing is called dual-route theory [CCAH93]. It proposes that there are two functionally independent channels of processing words. The graphophonological route relies on a non-lexical conversion of phonemes to graphemes, i.e., individual

sounds are mapped to its letter representations. This route relies strongly on phonological processing of spoken words. In contrast, the lexical route is active when the spelling of a word is retrieved by the representation in the orthographic lexicon. This means that a spoken word is directly linked two the entire spelling of a word.

Dyslexia Reading and writing are fundamental skills in the modern society. In the case of developmental dyslexia the acquisition of written language is impaired. Individuals affected by dyslexia are characterized by low reading and writing skills in spite of having an average IQ, good educational support and a solid social background [WHO93]. The causes of writing and reading failure are still debated. There are several theories focusing on the various impairments suffered by individuals with dyslexia, namely, (1) auditory, (2) visual, or (3) motor impairments:

1. The rapid auditory processing deficit theory states that the source of the reading and writing impairment can be found in the processing and integration of rapid sequential auditory stimuli. This leads to difficulties in the discrimination of sounds and the identification of the correct phoneme-to-grapheme correspondences [Tal80, BRW+99].
2. The visual processing deficit hypothesis assumes an impairment of the magnocellular pathway. This visual processing pathway is responsible for the inhibition of visual stimuli during the eye movements. A deficit of this inhibitory function leads to visual confusions and transpositions of letters [LWGN90, LH88].
3. The cerebellar deficit hypothesis states a general impairment in the automation of abilities. These involve coordinative, time estimation, as well as reading and writing abilities [NF90].

The most accepted theory, however, is the phonological processing deficit hypothesis [BB83]. This theory claims poor phonological awareness that manifests as an impairment in the phoneme to grapheme conversion [Fri85]. The phonological learning difficulties are linked with a reduced phonics-based memory as exhibited by individuals with dyslexia. While children with dyslexia rely on a non-phonological, visual coding strategy for the mediation of the written words in working memory, children without dyslexia use phonological coding [MK09]. Ramus et al. [RRD+03] give a more detailed overview on the most prominent theories on dyslexia.

Relevant cognitive functions In the evaluation of the learning progress we included other cognitive factors beside the indication of dyslexia. The investigation focused on the relevance of memory performances and attention functions on the process of learning. We examine these two cognitive functions, since there is evidence that they have a strong influence on language learning. On the one hand, it has been suggested that impaired memory functions can cause reduced phonological representations of words and lead to reading problems [SKD+04]. Attention functions, on the other hand, build the general basis for learning. Attention processes control all functions of our cognitive system, provided that tasks are not over-learned and automated [ZGF02]. Generally, attention helps people focus on the relevant information [PP87].

Multi-sensory learning Studies pertaining to learning, as well as investigations of memory, have predominantly focused on learning stimuli consisting of a single sensory modality or on uni-sensory memories. In recent years it has been suggested that in natural environments information is mostly integrated across multiple sensory modalities [SS08]. Thus, the human brain has evolved to develop, learn, and operate optimally in multi-sensory environments. There is evidence that multi-sensory training, as opposed to uni-sensory training, promotes more effective learning of information. Additionally, behavioral data indicates that multi-sensory encoded experiences enhance perception and facilitate the retrieval of memory [LM05]. This occurs even if the stimuli were only uni-modally presented in the retrieval condition.

Therapy software The benefits of multi-sensory learning and the indication that children with dyslexia use a non-phonological, visual coding strategies were integrated in the production of new computer-based training programs. The advantages of such computer-game-like training programs are that they have both recreational and didactic goals [GKG08]. Successful educational games aim at capturing the student's interest; thereby, motivating them to acquire knowledge.

A multi-modal training program based on associative learning was presented by Kujala et al. [KKC+01]. They were able to show that reading improved strongly after an association learning of abstract audio-visual material. Their computer-based training of basal components of reading and writing incorporates nonverbal tasks that require audio-visual matching of rhythm, pitch, and intensity. As a result, the trainee's multi-modal coding of speech stimuli improves, which consequently enhances reading and writing capabilities in seven year old children.

Other training programs focus on the core phonological processing deficit. Ecalle et al. [EMBG09] present a learning software package that includes audio-visual phoneme discrimination tasks. In their training tasks, orthographic units have to be discriminated based on simultaneously presented phonological units. This helps to improve both reading and spelling skills in children with dyslexia. An additional multimedia program was developed for children, in order to help them build up a phonologically underpinned orthographic representation, particularly for learning words with irregular phoneme-grapheme correspondence in Dutch [HR06]. The findings of this study indicate that practice with visual preview is significantly more effective than practice with normal dictations alone. The strong effect of the visual preview highlights the need for a prevention of misspelled words.

The spelling software presented by Gross and Vögeli [GV07] comprises meaningful multi-sensory stimuli and provides support for the visual coding strategies of dyslexic children. Previous findings of behavioral data indicate that meaning (e.g. discriminative environmental sounds compared to monotone sine wave tones) is necessary to facilitate the retrieval of multi-sensory encoded information [LM05]. The visual cues implemented in the learning software use colors, shapes, and graphs reflecting information about individual letters and syllables. Additionally, the system prevents the display of misspelled words by immediate auditory and visual feedback on committed errors. The software is described in detail in Chapter 3.

Error taxonomy The investigation of isolated word spelling errors has been strongly driven by the development of spell checking algorithms (e.g., [Atk06, SE97]). The classic taxonomy comprises four error types: insertion, deletion, substitution, and transposition [Dam64]. If only one of these letter-based operations needs to be applied to transform the misspelled word into the correct one, it is considered as an error with edit distance one. On average, 80 to 95 percent of errors committed by regular spellers were found to have edit distance one [Dam64, PZ84]. However, this is not the case for dyslexic subjects. James and Draffan found only 31% of all errors of dyslexics to be edit distance one [JD04]. The edit distance of an error strongly depends on the point of origin in the spelling process. As illustrated in Figure 2.1, errors occur in the auditory perception, in the phoneme to grapheme transformation, and in the mechanical writing of the word. Typing errors, originated in the last step, result predominately in single insertions, substitutions, or transpositions. However, errors generated in the listening or the phoneme-grapheme conversion step often feature an edit distance greater than one, due to their phonological error source.

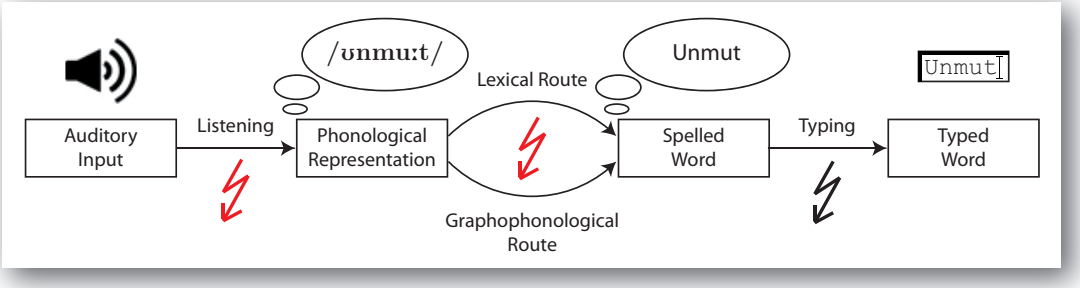

Figure 2.1: *Error possibilities in the spelling process: The transformation of auditory input to a phonological representation and its conversion to the mental representation of spelling originates errors with edit distance mostly greater one. Errors generated in the mechanical process of typing commonly feature an edit distance of one.*

To cope with errors of edit distance greater than one, Wagner and Fischer employed string matching algorithms in their spell checker development [WF74]. String matching on a letter level enables a description of errors with arbitrary large edit distance. However, Veronis raised the concern that these algorithms consider only letter-based errors, and most errors made by dyslexics are of phonological nature [Ver88]. In his work, he proposed a string matching approach on a phonological level. The algorithm is based on ordered pairs composed of a grapheme and its phonemic counterpart. Veronis found 141 of these couples - called graphonemes - in the French language, of which 40 are sufficient to represent 90% or the analyzed corpus. This enables the identification of similarities between, e.g., the word *hypoténuse* and its misspelled representation *ippeauttainnuze*.

However, the application of a spelling training software implicates special requirements on the error representation. The main factor which limits the applicability of the surveyed error descriptions is, that the available information for error analysis varies from the spell checking setting. On the one hand, the immediate feedback of recent spelling software on committed errors restricts the error analysis on the input up to the error letter. Therefore, string matching algorithms requesting the entire misspelled word are inapplicable or have to be adapted to the special setting. On the other hand, the intended word is known and has not to be estimated. The student model is not concerned about the correct word, but about the category and source of an error.

James presents an error taxonomy structured according to the source of errors [Jam98]. In ”Errors in Language Learning and Use” he describes language errors of different levels, including a phonological/graphological,

a grammatical, and a discourse level. On the phonological/graphological level, which is the only one relevant for isolated word spelling, the distinction between errors is made according to their origin. For example, errors are classified as dyslexic if they match common dyslexic error patterns, such as visual confusion of *'d'* and *'b'* as well as phoneme-grapheme matching errors. However, the taxonomy is indifferent on the required information to identify an error category. Many described error types become manifest only in higher level context information, which is not available in the isolated word spelling setting. Additionally, the taxonomy comprises second language (L2) errors and typing error patterns of experienced typists, which can be neglected for native German speaking children.

2.2 Student Modeling

The ability to adapt to student needs is a central feature of an intelligent tutoring system. This ability is based on an abstract representation of the student, which is called the student model. These adaptive systems have produced impressive gains in user studies [SP96].

Knowledge representation A student model is mainly characterized by its form of representing the student knowledge in one specific domain, e.g., spelling. The most prominent representatives are overlay models, perturbation models, and cognitive models. The three categories are described in the following and illustrated in Figure 2.2.

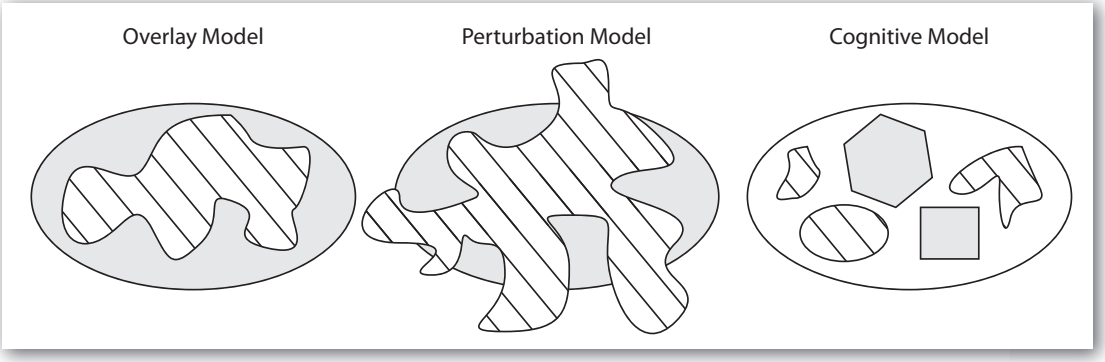

Figure 2.2: *The three student representations: Expert knowledge is shaded in gray and the student knowledge is depicted by the ruled area. In the cognitive model, the full expert knowledge is rather a set of skills than an area of knowledge.*

The first method employed for student modeling was the concept of overlay (e.g., BIP [BBA76]). In such a system, the student's knowledge is treated as a subset of an expert's knowledge. The goal of training is to extend the student's knowledge until the congruence between the two is established. This approach however assumes that all differences between the student behavior and expert model is caused by a lack of knowledge of the student.

Another instance of student knowledge representations is the perturbation model (e.g., DEBUGGY [Bur82]). In contrast to the overlay model, where students are represented only in terms of correct knowledge, a perturbation model takes also faulty knowledge into account. The learner is not only considered as a subset of the expert, but is represented based on misconceptions, called mal-rules.

The third main category of knowledge representations is called cognitive student models (e.g., model tracing [ABCL90]). In these systems students are represented as a subset of a cognitive model for the domain. The difference to an overlay model is the fact that it does not directly model domain knowledge, but independent production rules or skills which allow to solve the exercises of the domain.

Due to the strongly differing spelling process of children, we decided to employ a perturbation model for spelling. It allows for the design of error generating mal-rules and renders a specific cognitive model for spelling unnecessary.

Inference algorithm A second important element of a student model is an inference algorithm for the estimation of the student mastery of the domain, and the subsequent update of the student model, while the student is training. In recent years various methods have been developed for a broad range of educational applications, specialized on application-specific student input data. However, most inference algorithms rely on the constraint that every input can be assigned to one single rule or mal-rule. A well-known example is Corbet and Anderson's knowledge tracing approach [CA95]. This constraint requires a decomposition of tasks into pieces of single skills as provided in cognitive modeling approaches. However, these decompositions are not desired or possible in many applications. For example, the commonly cited mixed-number fraction subtraction [Tat85] requires multiple skills for one calculation. Similar issues arise in spelling tasks, where the input of a single letter cannot be broken down further. A unique association of an error with a mal-rule is not possible. To overcome the ambiguity of the error source in mixed-number fraction subtraction, Mislevy employs a Bayesian inference

network for skill estimation [Mis96]. Other approaches dealing with the ambiguity are multiple classification latent class models [Mar99] or linear logistic test models based on item response theory [Fis73]. These methods estimate the probability of success or failure on one given item. However, the immediate feedback on committed errors and the subsequent correction by the student in spelling software allows for multiple errors at one single letter position in a word. Previous methods do not make allowance for multiple errors in one single task. This multiplicity requires a different viewpoint on student errors. Therefore, we regard spelling errors as randomly occurring events, which are best described by a Poisson distribution.

In addition, many inference algorithms, such as knowledge tracing or higher-order latent trait models [dlTD04], assume the student attributes to be either in a learned or unlearned state. In a learned state, errors can only be committed due to slipping; in a unlearned state, correct answers can be achieved through guessing. In contrast, mal-rules describe the difficulties in spelling and divide them into different categories. These mal-rules do not represent a concept of spelling which can simply be acquired and once mastery is reached, only slipping would cause subsequent errors of the same type. For example, visual and auditory confusions of letters in dyslexic children cannot simply be comprehended and removed. Therefore, we represent the strengths and weaknesses in spelling by the error rates on mal-rules for every individual student. For the estimation of these error rates we propose a Poisson regression [MN89] with a linear link function to assure the independence of the factors.

Models of spelling Student modeling for spelling has been mostly neglected so far. Commercial spelling learning environments focus on sustaining the children's attention and motivation by utilizing the multi-media abilities of recent computer systems [ISW, SS, US]. However, their adaptation of the training to the student is limited to the repetition of erroneously entered words.

A core challenge in building a student model for spelling is the identification of patterns and similarities in spelling errors across the entire word database, and to represent them using as few mal-rules as possible. Bodén et al. propose a language-specific set of 68 letter patterns to describe spelling difficulties. In their evolutionary approach of adapting spelling exercises, they respond to erroneous inputs by selecting similar words [BB07]. Error localization and classification with respect to specific difficulties are not considered. This leads to an inappropriate adaptation for many error categories, e.g., capitalization: *Spiel* - spiel (engl. game). This capitalization error results in a selection of

a similar word, such as *viel* (engl. plenty), which does not contain an error possibility for the previously committed capitalization error.

Spencer predicts spelling difficulties from measures of orthographic transparency, phonemic and graphemic length, and word frequency based on English language corpora [Spe07]. The resulting difficulty measure takes only a limited set of error types into account and neglects for example visual and auditory confusions or typing difficulties. The measure is independent of the training student and does not allow for an individual adaptation. Additionally, language corpora are usually based on extensive collections of print samples, of which the word frequencies may be quite different from those of spoken, heard or hand-written language of children [BPC01].

Similar drawbacks appear in Bader-Natal and Pollack's SpellBEE peer-tutoring system. Their difficulty estimation on the word database, computed from the input data of 17 000 students using the tutoring system, provides a relative spelling difficulty between words, which is independent of individual students [BNP07].

Affective modeling Due to its recognized relevance in learning, affective modeling is receiving increased attention. The goal of affective models is to represent emotional, motivation, and attentional states of students. The developed models can be grouped into systems utilizing sensor data (e.g., camera [CMA+10], EEG-measurements [HF09], and heart-rate sensors [Con02]) and systems that rely on student input data only (e.g., [Bec05, BCK04, JW06, AW05]). These sources differ in quality and quantity. On the one hand, sensor measurements tend to be more direct and comprehensive. They have the potential to directly measure larger numbers of affective features. On the other hand, input measurements are not limited to laboratory experimentation. The measurement of student interaction with a software tutoring system offers a unique opportunity: large and well-organized sample sets can be obtained from a variety of experimental conditions.

Recorded inputs have the potential to characterize the affective state of the student in a learning scenario. For example, Arroyo and Woolf presented a data-driven construction of a Bayesian net based on features extracted from log files, such as seconds per problem, hints per problem, and time between attempts [AW05]. The goal of the affective model is to support an intelligent tutoring systems (ITS) in adapting to the student's emotion and attitudes, or even to direct the student into a desired state. For example, Arroyo et al. created affective learning companions to influence emotions of students [AWRT09].

A core challenge in affective modeling is that experimental readouts and state emissions often exhibit partial observability and significant noise levels. Additionally, the observed input behavior can be subject to long-term progress. However, this violates the assumption of independent and identically distributed (i.i.d.) data, which is a necessary condition for many statistical inference methods. To enable the usage of these standard statistical inference algorithms, one either have to extend the affective model to allow for long-term variations in the observed input behavior or interpose a feature processing step to attain the desired properties. The latter alternative of incorporating domain knowledge into the feature processing, either as implicit or as explicit assumptions, can substantially increment the predictive power of the inferred models [BOB09]. However, this feature processing has mostly been neglected so far in affective modeling. Baker et al. used standard deviations from the mean of features as a possible scaling as well as measures over several inputs as filtering [BCK04]. Arroyo and Woolf as well as Johns and Woolf divided features into *High* and *Low* by a medium split [AW05] or manual thresholds [JW06] respectively. However, the appropriateness and optimality of these decisions have barely been studied so far. In this thesis we will introduce an systematic approach to incorporate domain knowledge about affective dynamics into the processing of features.

Part I

Data

CHAPTER 3

Dybuster

Dybuster is a multi-modal spelling software for dyslexic children and constitutes the very heart of this thesis. The goal of the presented work is the modeling and evaluation of learning processes in intelligent tutoring systems, which will allow for an improvement of the Dybuster training software. All developed models and analyses rely on the data collected in Dybuster user studies, described in detail in Chapter 4. This chapter gives a brief description of the main components of the original software and specifies the enhancements incorporated in the second version. A more detailed description of the entire multimedia framework for the dyslexia therapy can be found in [GV07].

3.1 Overview

The Dybuster spelling software combines concepts of visualization, statistical modeling of language, information theory and psychology. The core idea is the multi-modal recoding of spelling information to bypass the distorted cognitive cues of dyslexic children and build alternative cerebral retrieval structures. The multi-sensory representation consists of a spatio-topological, a color, a shape, and an auditory code. The codes represent spelling information on a letter and syllable level (see Section 3.2). This combination

of meaning and visual as well as auditory representations has shown to facilitate the retrieval of multi-sensory encoded information [LM05]. The abstract, graphical recoding of the word supports the visual coding strategy of dyslexic children.

Dybuster is structured into three different games. The first two games focus on learning the multi-modal representations, to facilitate the usage of the learning aids. The goal of the third game is the actual spelling training. The three games are described in detail in Section 3.3. To additionally support the effectiveness of the multi-modal therapy approach, the software comprises different motivation enhancing elements, which are presented in Section 3.4. This original Dybuster training software is evaluated in a large-scale user study. The analysis of the collected input data and the development of student model are subject of this thesis. The gained insights and developed models lead to a set of enhancements incorporated in an improved version of the training software. These phoneme-based enhancements are described in Section 3.5.

3.2 Information-theoretical Model

The information-theoretical models of Dybuster rely on the concept of entropy [Sha49]. Entropy is a measure of uncertainty associated with an given information source. It represents the average amount of information contained in a message obtained from that source. The concept of entropy is employed in two components of the software. On the one hand, as a measure of spelling information in written language for the optimal design of codes. On the other hand, the entropy serves as a measure of the potential of improvement in the word selection process. In the following, we will discuss the two information-theoretical models.

3.2.1 Information Cues

The spelling of words is modeled by a Markovian language model derived from linguistic analysis of the ECIGer language corpus [ECI94]. The symbol entropy of the Markov-1 model serves as a measure for the information contained in the spelling of words. The central idea of the training software is the recoding of this spelling information into a multi-modal representation by using a set of codes. Figure 3.1 shows the general concept involving a topological, an appearance (shape and color), and a musical encoder. These encoders transform an input string into the following four codes:

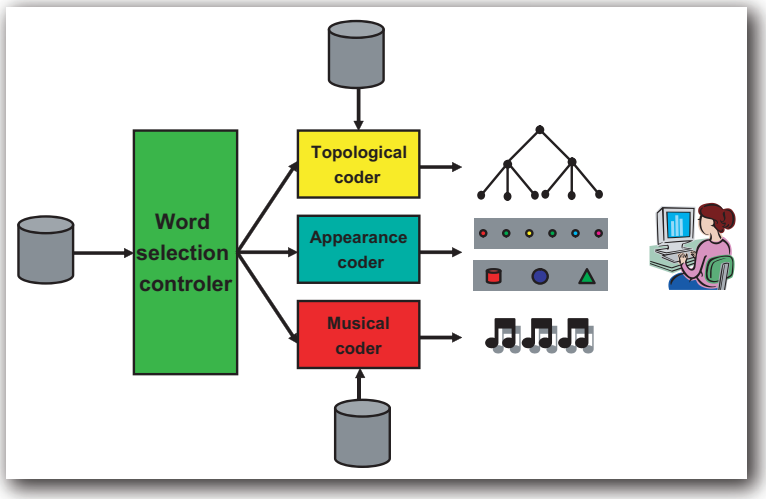

Figure 3.1: *Conceptual components of the learning software [GV07].*

1. *Topological code:* The topological encoder parses the word recursively and decomposes it into a syllable tree. This tree generates a topological representation of the syllable structure of the word. The purpose of the topological code is to provide a clear structure. It supports the students in their serial behavior during spelling because it assists them with putting the letters in the right position.

2. *Shape code:* The appearance encoder assigns shape primitives to letters to distinguish between regular letters (sphere), capital letters (cylinder), and umlauts (tetra).

3. *Color code:* The appearance encoder assigns each letter to a color value of the color alphabet **C**. The mapping of letters to colors is the result of a multi-objective optimization, taking into account that, e.g., letters easily confused by dyslexics, such as *'t'* and *'d'*, map to different colors. The idea is that associating colors and letters will allow eliminating mistakes.

4. *Auditory code:* The musical encoder translates the visual representation, i.e., its topological and appearance code, into a set of midi events. The musical attributes include pitch for color and instrument for shape, as well as duration and rhythm for syllable lengths. Thus, the created melody forms a multi-sensory support to the information transfer.

The topological and appearance representations are designed to have a greater or equal joint entropy than the original spelling of the word. To fulfill this

requirement, the number of colors $|\mathbf{C}|$ was chosen to be eight in the German Dybuster version. As described above, the information of the visual representations is additionally encoded in the auditory representation. This ensures that all information is available on a visual as well as on an auditory cue.

3.2.2 Word Selection Controller

The Dybuster version employed in the user studies contains a dictionary of 1500 words with the level of difficulty corresponding to the student's elementary grade. The words are grouped in modules with respect to their frequency of occurrence in the language corpus as well as a word difficulty measure. The later is computed based on word length, number of dyslexic pitfalls and silent letter pairs. The students start with the easiest module first and work through one module after the other.

Inside one module, it is the objective of the word selection controller to select a word from the module in such a way that the user makes most progress. Progress means to reduce the uncertainty in the student's knowledge about the correct spelling of words in the module. This uncertainty is represented by the error entropy computed based on a global symbol confusion matrix and a local word error history:

- *Symbol confusion matrix:* The conditional probability $P(x_k|x_l)$ of confusing a letter x_l with another letter x_k is stored in the symbol confusion matrix. After each student input this matrix gets updated.
- *Word error history:* To account for the word specific spelling difficulties not represented by the individual letters, Dybuster keeps track of the student's error history of the last two inputs for each word.

The word selection is computed according to a cost function based on the global and local error entropy and the word frequency. This results in a progress maximizing training. The modules are switched when the local error entropy of its words falls below a certain threshold.

3.3 Games

The Dybuster software is structured into three different games. In the first game - the color game - the students have to learn the association between letters and colors. Initially, the letters are displayed in their corresponding color, as illustrated in Figure 3.2, left. The goal of the game is that the student selects the correctly colored button below the letters. After correct inputs the

Figure 3.2: *The color game trains the association between letters and colors. After correct inputs, the colored letters fade to white.*

color saturation of the presented symbol fades to zero, requiring the student to memorize and recall the color (see Figure 3.2, right).

In the second game - the graph game - the students have to segment a word into its syllables and letters graphically, as shown in Figure 3.3, left. The student has to draw the correct tree of a given word by clicking onto arrays of nodes and by drawing the correct connections.

In the third game - the actual word learn game - Dybuster presents the alternative representations of a word. The graph appears on screen and the colors and shapes are displayed for all letters (see Figure 3.3, right). A voice dictates the word and the student hears the melody computed based on the involved letters. Supported by the presented learning aids, the student has to enter the word using the keyboard. Upon each keystroke, the auditory representation of the typed symbol is played again. The entire string, entered by a student after a spelling request, is in the following referred to as an input.

To avoid the display of entire misspelled words, Dybuster provides immediate visual and auditory feedback on erroneous keystrokes. Similar approaches are followed in other spelling training software [GKG08]. This immediate correction is paramount to effective training [Bro90]. However, it makes an unambiguous classification of errors more difficult, as described in Chapter 5. The immediate correction also leads to the possibility of committing several errors at the same letter position of a word. This characteristic strongly influences the design of the student knowledge representation introduced in Chapter 6.

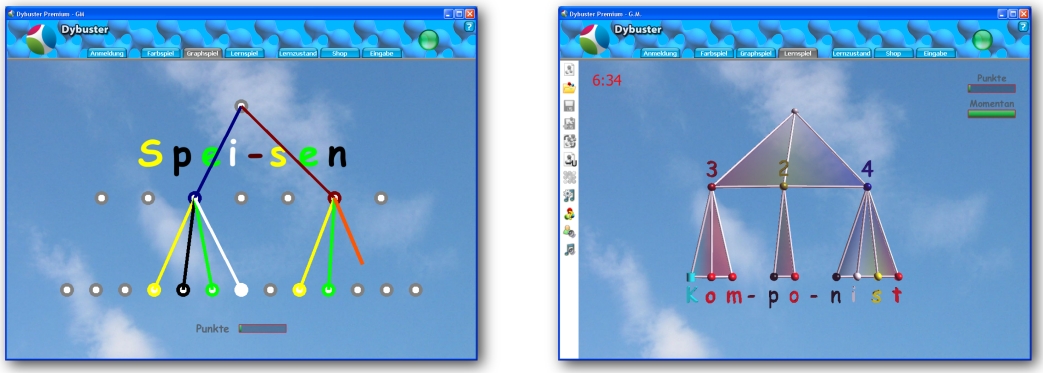

Figure 3.3: *Graph game (left) and actual word learn game (right) of Dybuster.*

3.4 Motivation

The training motivation is an essential component of the effectiveness of a dyslexia therapy. The games of Dybuster described above are not complex computer games. Game concepts, such as the memory-like color game, are only a part of the motivation enhancing strategy of Dybuster. The central idea in the design of the different training games is the incorporation of elements known from computer games. In the following we describe two components, which mainly contribute to the student training motivation.

3.4.1 3D Graphics

The game-like learning environment with 3D graphics and interaction components rather resembles a common computer game than a spelling training software. Children with dyslexia, who show increased aversion to spelling training, are able to perform their spelling training in an environment with positive associations. The interaction with the physically animated graph permits an immersion into the 3D-world (see Figure 3.4, left). In the enhanced Dybuster version, employed in the second user study, the individual information cues interact with the user them self. In addition to the immediate auditory feedback on erroneous inputs of the original version, the visual cues which could be used to prevent the specific error are highlighted. For example, a colored halo at the error position indicates a wrong color mapping, as shown in Figure 3.4, right. These interactions support the student in learning the correct usage of the presented information codes.

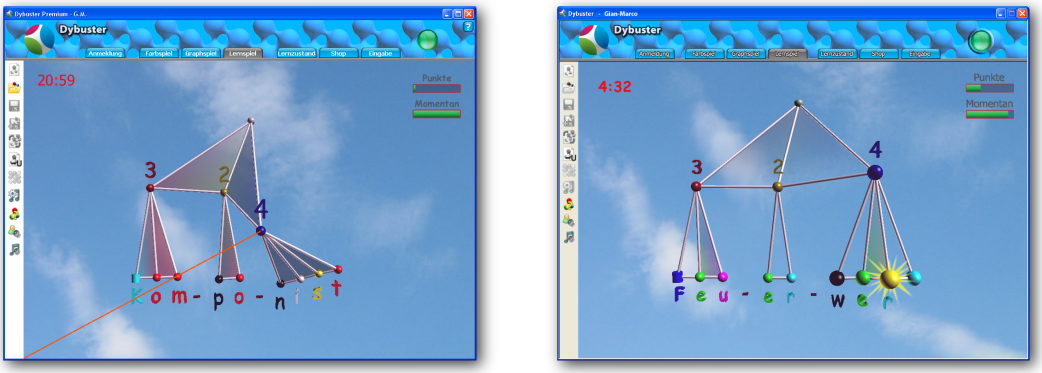

Figure 3.4: *Interactive learning aids: The physically animated graph (left) allows for user interaction; the different information cues, such as the letter color (right), are highlighted, if they could be used to avoid committed errors.*

3.4.2 Virtual Shop

Common role play computer games demonstrate how users are driven to collect points and items. To make use of this attraction in Dybuster, a score counter provides feedback on the actual learning state during the training. After a certain interval the points are converted into virtual money. This can be used to buy various items in a virtual shop implemented in Dybuster. These include new background images, visual effects after correctly entered words, and different instruments to play the auditory code (see Figure 3.5). Such extrinsic motivation may not increase the intrinsic motivation for a spelling training [DKR99], however, it drives the students to continue the training.

3.5 Phoneme-based Enhancements

The original Dybuster version is extended for the second user study with a set of phoneme-based enhancements. They are implemented based on the notion that the core problem of dyslexia is a phonological processing deficit. The enhancements consist of an additional textural code and a phoneme-based spelling knowledge representation with a correspondingly adapted word selection controller.

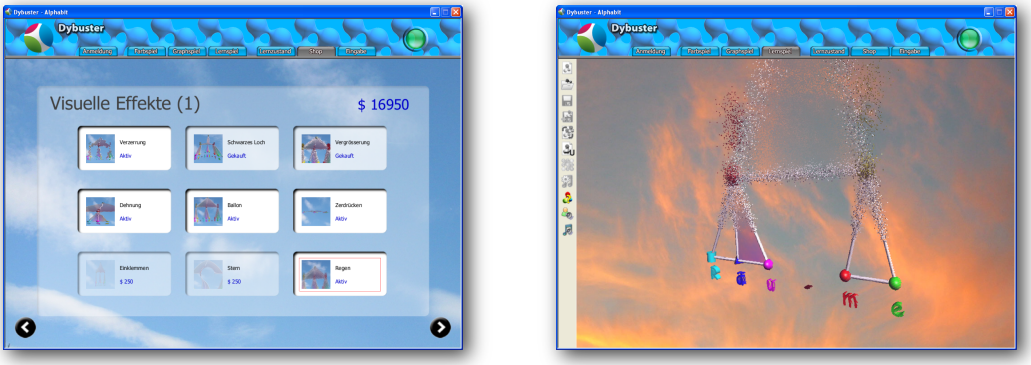

Figure 3.5: *The virtual shop (left) and an example item, which can be bought with the earned virtual money: a visual effect played after correctly entered words (right).*

3.5.1 Textural code

The additional textural code provides supplementary information to the existing visual codes. Whereas the topological and appearance encoder represent letter and syllable information, the textural code supplies easily extractable information about the phonological word structure (see textural code on Figure 3.4 and 3.5, both right). This textural code visualizes the catenation of multiple letters to one grapheme that represents the corresponding phoneme (e.g., *'sch'*, *'ch'*, *'ie'*, and *'ei'*). As can be seen in Figure 3.4, the correspondence of the graphemes *'eu'* and *'eh'* to the phonemes /ɔ_y/ and /e:/ respectively, is visualized by the textured triangles between the letters. This additional visual code supports the association between phonemes and graphemes, which has shown to strengthen the phonological awareness [EMBG09].

3.5.2 Adaptive Word Selection Controller

The major enhancement for the second user study is the phoneme-based spelling knowledge representation with the correspondingly adjusted word selection controller. This controller identifies the children's individual spelling weaknesses based on their error behavior and prompts words containing these difficulties. In contrast to the original word selection method of Dybuster, which relies on a letter-based analysis of errors, the novel controller accounts for spelling difficulties on a phonological level. For example, if a child struggles with spelling the word *Zahl* (engl. number) because it does not know that the word contains a silent *'h'*, then the controller selects and

prompts more words containing silent sounds, such as *sehr* (engl. very) or *ahnen* (engl. guess). Therefore, the child is enabled to train on its individual spelling problems. Consequently, the child learns the linguistic spelling rules based on the German language and generalizes them to other words after training. The spelling knowledge representation and the word selection controller are described in more detail in Chapter 6 and 9.

3.6 Conclusion

The Dybuster spelling software described in this chapter is fundamentally different from earlier approaches in that it combines theories from different areas of research, such as information theory and psychology. The concept of routing spelling information through different information cues is not based on one specific neuropsychological theory of dyslexia. The framework provides cues operating on different senses and is therefore prepared for the diversity of deficits of dyslexic students. Additionally, the multimedia-based software comprises motivation supporting elements and provides a learning environment appealing to children.

The original as well as the phoneme-enhanced software version are evaluated in large-scale user studies, which are described in the next chapter. The data collected in these Dybuster user studies builds the basis of the work presented in this thesis.

CHAPTER 4

User Studies

So far, we introduced the theoretical foundation of Dybuster and its specific implementation in the three games. The actual clinical effectiveness of such a therapy approach can only be assessed by means of evaluation studies. In this chapter we describe the two large-scale user studies conducted with the Dybuster spelling software. The data collected during the first study allows for an analysis of learning processes in the first place. In addition, a follow-up study enables an evaluation of the enhancements incorporated into the software.

4.1 Overview

In the year 2006 and 2008, two large-scale user studies were conducted with the Dybuster spelling software. This chapter describes the two studies and the collected data in detail. We first specify the subjects participating the studies, the test battery they had to run through, and the general study design and procedure. This setting is maintained for both studies to allow to draw comparisons between them. Then, we will give more details of the first and second user study individually. This includes the description of participants, employed software version and the presentation of first results based on the pre-, mid-, and post-spelling tests.

At the end of the chapter we describe the log file data collected during the user studies. Based on the extracted training times we demonstrate the comparability of the two studies. The log files collected in the studies serve as the foundation of the further analysis and modeling.

4.2 Subjects

In total, 142 children (80 dyslexic and 62 control) participated in the two user studies. All participants were native Swiss-German-speaking and aged 8-to-12. The IQ of the subjects was above 85. Children with an IQ below 85 were excluded from the studies. Children were categorized as dyslexic based on previous diagnosis by trained diagnosticians, such as, therapists or school psychologists. In order to further validate the diagnosis, children with dyslexia were categorized as reading and spelling disabled if their scores were below the 10^{th} percentile on the standardized spelling and reading tests. In contrast, the reading and spelling skills of children without dyslexia were not more than one standard deviation below the mean ($\geq 15.9\%$). Children without dyslexia were recruited from responses to letters distributed in elementary schools or presentations in school classes where the program was demonstrated. The recruitment of children with dyslexia was conducted primarily with the assistance of therapists or educational psychology services. Both children with and without dyslexia attend public schools. All of the children's parents gave their informed consent for participation in the study as per the Declaration of Helsinki. Experimental procedures were approved by the local Ethics Committee (SPUK).

4.3 Test Battery and Procedure

Before the training took place, an information event was organized for both children and their parents of the first and the second study, in order to distribute detailed instructions about the study design and the concept of the learning software. Notably, the software is designed in a way that children can accomplish the training for themselves and do not need additional help or parental assistance. Detailed information about the handling of the learning software was presented on the first training day.

After providing general study information and before the actual training began, all study participants underwent a series of standard psychological tests (see Appendix A for results). The test battery for the participants in the first and second study differed slightly. In the first study, children performed

the classical German spelling tests, "Salzburger Lese- und Rechtschreibtest SLRT" [LWM97] or "Diagnostischer Rechtschreibtest für fünfte Klassen DRT5" [GHNW95]. This enabled us to quantify their spelling skills. There were two different spelling tests applied because the SLRT contains only norms from the first to the fourth grade. Thus, the DRT5 was administered to the fifth graders. Additionally, all children were required to accomplish a standardized reading test, "Zürcher Lesetest ZLT" [LG00], which permitted the quantification of their reading skills. This reading test contained two subtests, namely, reading of word lists and texts; performance was measured as time used and errors made. A German intelligence test named "HAWIK III" [TR99] was also administered, in order to assure average or above-average general cognitive skills in all subjects.

In the second study, the aforementioned test battery was expanded with a pseudoword reading test from the "Salzburger Lese- und Rechtschreibtest SLRT". Additionally, to evaluate verbal memory functions, a verbal learning and retentivity test, that is the "Verbaler Lern- & Merkfähigkeitstest VLMT" [LHE99], was administered. This test measures learning performance, as well as short- and long-term memory by using word lists that must be repeated five times and recalled after half an hour. In addition, the attention functions were tested by a version of computer based program called KiTAP that is specifically designed to examine children [ZGF02]. This allowed us to test alertness, flexibility, and impulse control. Alertness forms a crucial role in attention intensity; it constitutes the processes of tonic and phaseal arousal [PR87]. Flexibility is the ability to adapt to a new situation. The disability to realign the focus of attention causes preservative and stereotypical behavior [Lez95]. Impulse control is the ability to refrain an inadequate reaction and is tested by a Go/No-Go Task [Dre75]. While the KITAP computes the percentile of reaction time as a measure for alertness, the percentile of errors is used as a measure of flexibility and impulse control.

Low scorers performed half a standard deviation below ($\leq 30\%$), and high scorers performed above ($\geq 70\%$) the mean for a given attention or memory function. The spelling tests were accomplished in a classroom setting. Reading, verbal memory, attention, and IQ tests were conducted in an individual test setting.

4.4 Study design

The dyslexic and control subjects were randomly assigned to training or waiting groups. The training groups (dyslexics: DW; control: CW) immediately began with the Dybuster training. The waiting groups (dyslexics:

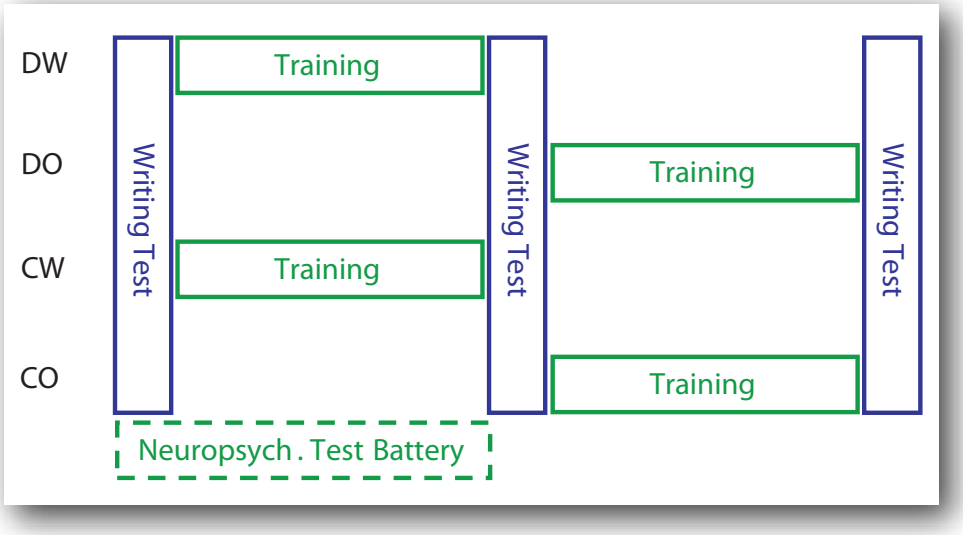

Figure 4.1: *Sequence of actions in the design of the two user studies [KMV+07].*

DO; controls: CO) performed the Dybuster training after a waiting period of 3 months. In the first three months the training groups were asked to practice about five times a week for 20 minutes each. After this training period, no further training was undertaken in the second period of the study. The children of the waiting group now began their training.

Beside the above mentioned psychological tests, the children's writing amelioration was measured by a dictation containing 100 words. A random half of the words were used during the training session and the second half served for testing the children's ability to generalize to novel words. All the children had to pass the writing test before training, after three months and at the end of the study. Figure 4.1 illustrates the study design.

The training generally took place on participants' home computers. Participants were offered the option of undergoing supervised training at our lab once a week (see Figure 4.2). The meeting at our lab enabled us to monitor the data, which included checking children's working behavior and making sure that no technical problems occurred. For monitoring reasons, the parents of children who did not come to our lab once a week were requested to send us the log file data. During training, the children worked at their own individual pace.

Figure 4.2: *A child working with the Dybuster spelling software in a supervised training session at the ETH.*

4.5 First User Study

The first Dybuster user study took place in the year 2006. This study was conducted with the original version of Dybuster. The goal of the study was to examine the general concepts of Dybuster and evaluate the efficacy of the training. Kast et al. [KMV+07] describe the user study and its outcome in more detail.

4.5.1 Detailed Description

43 dyslexic children (15 females) and 37 controls (17 females) with an average age of 10.3 and 10.2 years, respectively, participated in the study (IQ dyslexics: 105; IQ controls: 113). No further distinction was made between possible subgroups. Two children were excluded from the study because of poor performance in the classical writing tests and in the Dybuster writing test. Two additional children were excluded because they performed the writing test only once.

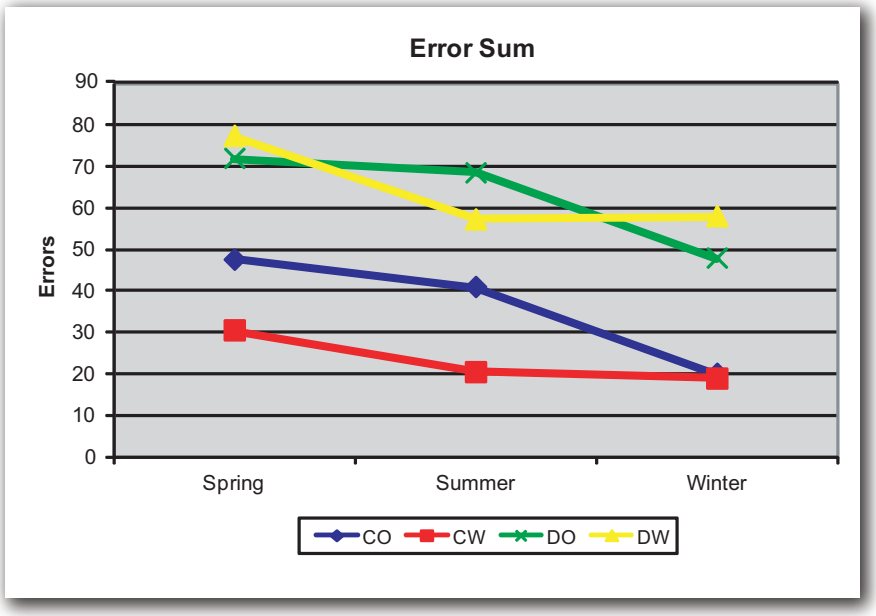

Figure 4.3: *Spelling progress in the first Dybuster study measured in the three writing tests [GV07].*

4.5.2 Results

The results of the spelling tests showed a significant improvement. The writing skill of the children with dyslexia DW improved on average by 27% in their training period. Whereas the counterparts DO without training improved only 4%. There was no improvement at all on 1/3 of the DO group. These results provide evidence for the effectiveness of the multi-modal training method. Furthermore, the DW group improved by 32% on words from the learned subset and 23% on the generalization dataset. This result leads to the conclusion that the recoding can effectively generalize to new and unknown words - a highly desirable property. Finally, compared to non-dyslexic children, the groups CW and CO improved by 27% and 17%, respectively, during the first period.

The spelling tests after the second period indicate that the spelling knowledge is stable after suspending training. DW and CW were able to maintain their error rate. As expected, DO and CO improve the spelling skills significantly (27% and 33%) in their training period. Figure 4.3 gives a graphical summary of the results.

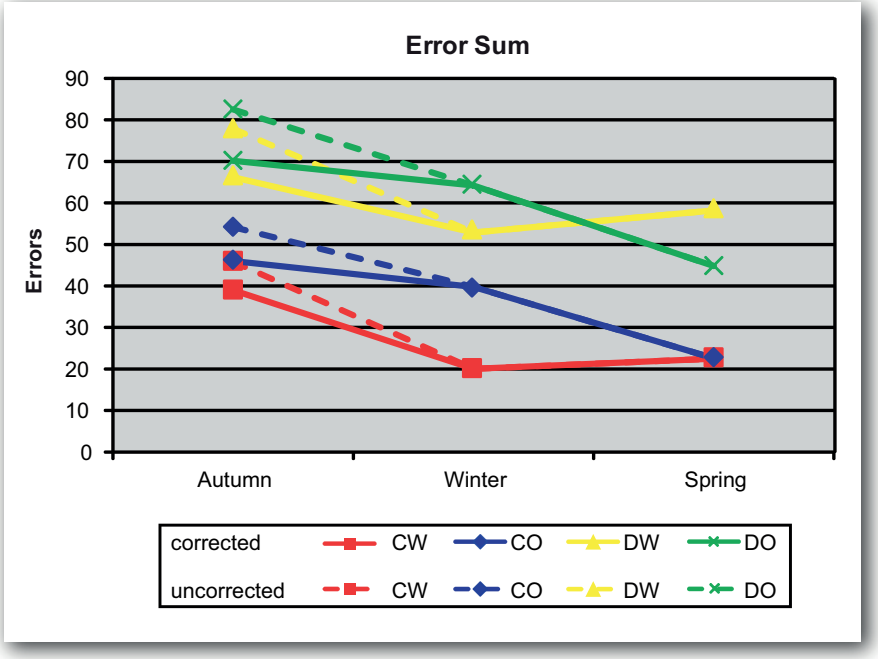

Figure 4.4: *Spelling progress in the second Dybuster study measured in the three writing tests. Dashed lines indicate the actual spelling progress. Solid lines depict the progress corrected for the changing settings.*

4.6 Second User Study

The second Dybuster user study was conducted between summer 2008 and spring 2009. In this study we employed the enhanced version of Dybuster, featuring all phoneme-based enhancements. There were two main goals for this study: first, to evaluated the influence of the novel phoneme-based enhancements; second, to investigate the influence of dyslexia and different cognitive factors, such as attention and memory functions, on the learning progress.

4.6.1 Detailed Description

37 children with dyslexia (10 females) and 25 children without dyslexia (12 females) with an average age of 10.9 and 10.3 years respectively were recruited for the second study. As described in Section 4.3, we run a series of additional tests to measure attention and memory functions of the children.

The three Dybuster spelling tests - building the foundation for the progress analysis of the first study - were also administered in the second study. However, in the first spelling test at the beginning of the study prerecorded words were played over loudspeakers. Due to feedback of participating children, the setting has been changed to an oral dictation in the following two spelling tests. This decision changed the difficulty of the dictations and made the comparison of results defeasible. For a comparison to the first study, we corrected the spelling performance of the first test by an extrapolation of the average spelling progress from the second to the third test (see Figure 4.4). However, this correction is based on a strong assumption that the progress in the first and second period are equal. Therefore, the following evaluations of the spelling progress are purely based on the collected log file data.

4.7 Log Files

During the Dybuster training, all user interactions were recorded. Every keystroke is time-stamped and stored in a log file. Listening 4.1 shows 11 lines from an example log file: (1) time and word of the prompt; (2-9) student inputs including the correction (the *'8'* depicts the backslash key) of the capitalization error, illustrating the immediate correction of errors; (10-11) information about the student representation of the word selection controller. This log file data allows for an exact reconstruction of the training process and serves as the basis for the analysis and student modeling presented in the following chapters.

Listing 4.1: *A snippet from an example log file.*

```
2:9:2006:9:58:16:413#Adrian#LearnVocGame#−4#Sei−te
2:9:2006:9:58:21:370#Adrian#LearnVocGame#S#s
2:9:2006:9:58:23:593#Adrian#LearnVocGame#8#8
2:9:2006:9:58:25:586#Adrian#LearnVocGame#S#S
2:9:2006:9:58:27:328#Adrian#LearnVocGame#e#e
2:9:2006:9:58:27:579#Adrian#LearnVocGame#i#i
2:9:2006:9:58:27:839#Adrian#LearnVocGame#t#t
2:9:2006:9:58:28:89#Adrian#LearnVocGame#e#e
2:9:2006:9:58:28:600#Adrian#LearnVocGame#13#13
2:9:2006:9:58:32:446#Adrian#LearnVocGame#SymbolErrorProb#0.63424
2:9:2006:9:58:32:446#Adrian#LearnVocGame#WordErrorProb#0#0.0202863
```

Mean over		First 30 sessions			Entire study	
Study	Subjects	Minutes per session	Inputs per session	Total minutes	Total sessions	Total minutes
1^{st}	dyslexic	16.3	54.7	581.3	54.4	863.7
	control	17.6	69.8	621.3	45.8	809.1
2^{nd}	dyslexic	16.1	51.9	573.2	58.0	947.3
	control	16.5	65.3	574.2	56.6	931.1

Table 4.1: *Information about the training frequency (only word learn game).*

Due to technical challenges in the first user study, some log files were corrupted during the training. Despite this loss of data, we obtained 54 completely and correctly recorded log files from this study. In total, the collected user data available for further analysis consists of the 54 correctly recorded log files from the first user study (28 dyslexic/26 control) and the 62 log files from the second user study (37 dyslexic/25 control). Table 4.1 lists the recorded training times in the word learn game of the two studies. Children of the second user study performed slightly more training minutes over the entire training period. This increased motivational continuity can be traced back to the extended shop concept in the second user study. However, between the three groups dyslexic 1^{st}, dyslexic 2^{nd} as well as control 2^{nd}, employed for the spelling progress analysis in Chapter 10, no significant differences were found in the training times of the first 30 training days.

4.8 Conclusion

The two large-scale user studies described in this chapter provide evidence for the efficacy of the Dybuster training. The pre-, mid-, and post-spelling tests show a strong learning progress over the entire training period. For a more detailed modeling and analysis of the training process we employ the collected log files. This user input data allows for a detailed reconstruction of the training process and enables the modeling and evaluation of learning, forgetting, and affective dynamics presented in the following chapters.

The consistency of the specification of the two studies facilitates a comparison of the learning progress between children of both studies. Additionally, the large set of neuropsychological testings will allow for an investigation of the influence of different factors on the learning process.

Part II

Modeling

CHAPTER 5

Error Model

Preliminary investigations of the collected input data have shown that a detailed analysis and modeling of the student require a classification of committed errors in different categories. The information gained from a simple typing error or a severe spelling misconception differs strongly and has to result in adapted remediation actions of the spelling software. Therefore, we introduce an error taxonomy for isolated word spelling and a corresponding set of error generating mal-rules to describe errors in detail. Our taxonomy takes account for the various spelling difficulties dyslexic children suffer from. It is structured according to the level of information errors become manifest in. This illustrates, which errors will be detectable in the specific setting of recent spelling software with immediate feedback on committed errors. Based on the described set of mal-rules we will be able to further analyze the student input data in more detail.

5.1 Overview

The Dybuster user studies provide us with task observations of isolated word spelling. The participating subjects are native Swiss-German-speaking children aged 8-to-12. In this chapter we describe an error taxonomy with corresponding set of mal-rules designed for this specific task and setting.

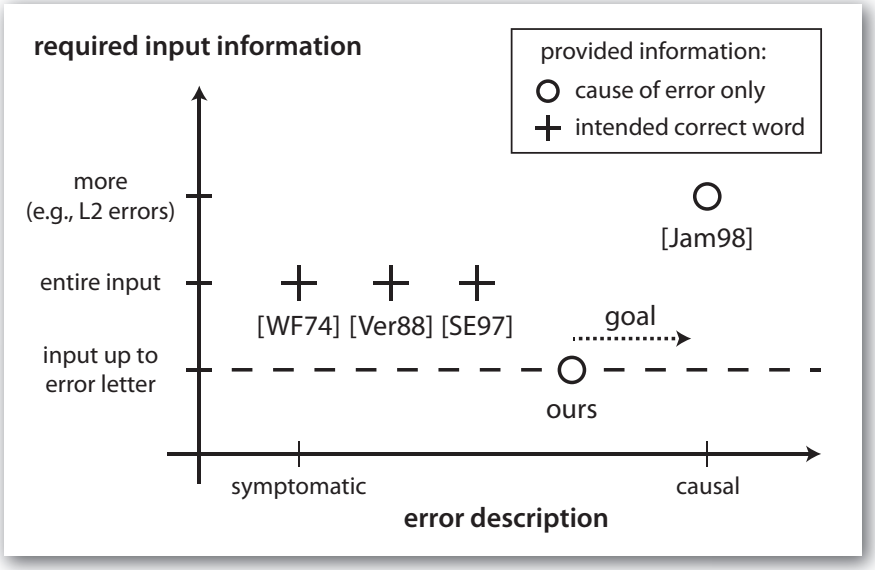

Figure 5.1: *Different error representations arranged according to type of error description and required information for it. Spell checker algorithms (+) not only provide a description of an error, but also an estimation of the intended correct word. In the spelling software setting information about input is only available up to error letter.*

The error taxonomy introduced by James gives a very detailed description of errors structured according to the source of errors [Jam98]. However, the taxonomy comprises various error categories not relevant for isolated word spelling, such as grammatical and semantical errors. Additionally, errors of the presented categories often become manifest in high level information only. For example, the identification of second language (L2) errors would request not only information about input and correct word, but also about the specific student and other languages.

In contrast, error representations based on the classic four error types insertion, deletion, substitution, and transposition [Dam64] are not sufficient for the analysis of dyslexic spelling errors. String matching methods [WF74] based on this very symptomatic description do not provide enough information about the origin of errors, which is inevitable for a detailed investigation of the collected input data. As illustrated in Figure 5.1, extensions of string matching approaches based on graphonemes [Ver88] or user models [SE97] already incorporate some additional information about the error cause. However, the design of spell checker algorithms (depicted by a '+') generally relies on the entire input string.

Our error taxonomy with corresponding set of mal-rules for isolated word spelling, has to be designed according to the following spelling software specific settings:

- Due to immediate feedback on committed errors, the user input is available up to the error letter only. The mal-rules can not rely on the entire erroneously spelled word. Therefore, structuring the taxonomy according to the level of information errors become manifest in, allows to identify which categories can actually be described based on the available information.
- The correct word intended to spell by the student is known. In contrast to the spell checking setting, the correct word has not to be estimated.
- The intention to model the student spelling knowledge based on the provided set of mal-rules demands an as simple as possible description of errors, which still allows for a representation of the very diverse spelling difficulties of dyslexic children.

Since many errors committed by dyslexic children are of phonological nature our taxonomy and mal-rules strongly rely on the phonological structure of words. In the following, we first describe the phoneme-grapheme correspondence in the German language in more detail. Then we present the developed error taxonomy for isolated word spelling and the corresponding set of mal-rules.

5.2 Phoneme-Grapheme Correspondence

In this section we address the relationship between speech sound (phoneme) and written symbol (grapheme). First, the basic terms phoneme and grapheme are introduced. Second, we describe their correspondence in the German language and the associated difficulties.

5.2.1 Phoneme

Speech consists of a series of individual sounds. A phoneme is the abstract class of sounds with indistinguishable meaning. The physical production of the sounds may vary with respect to the context, as can be seen at the phoneme /x/ in the words *i**ch*** (engl. me) and *a**ch*** (engl. alas). Nevertheless, the different articulations of the phoneme /x/ never affect the meaning of a word. However, the alteration of the sound /a:/ in *l**ah**m* to /a/ in *L**a**mm*

changes the meaning and is therefore described by two different phonemes. Hence, we can define a phoneme as:

The smallest phonetic unit in a language that is capable of conveying a distinction in meaning [AHD00].

The phonological representation of the 1500 words used in the Dybuster user studies, requires a set of 54 phonemes. They are divided into vowel and consonant phonemes. In this paper we employ the ETHPA ASCII-notation [Pfi05] of phonemes, illustrated in Appendix B.

5.2.2 Grapheme

The written language in German is based on 29 letters. To represent the various sounds of the spoken language, multiple letters have to be combined. These letter groups representing a phoneme are called grapheme. For example, the word *schieben* (engl. push) consists of eight letters but only of five graphemes (*'sch'-'ie'-'b'-'e'-'n'*). A grapheme is defined as:

All of the letters and letter combinations that represent a phoneme [AHD00].

The non-bijective interrelationship between graphemes and phonemes will be investigated in the next section.

5.2.3 Correspondence

This section discusses the relation between phonemes and graphemes. In a phonemic language, each phoneme corresponds to one grapheme. Georgian, Esperanto and Sanskrit are examples of strictly phonemic languages. However, German, like most of the Western languages, is not a phonemic language. There exists no bijective function which maps the phonemes to the graphemes. Several phonemes can be represented by multiple graphemes as well as some graphemes can belong to more than one phoneme. Veronis denotes such phoneme-grapheme couples graphonemes [Ver88]. Appendix B shows all graphonemes occurring in the 1500 words employed in the learning software during the user studies.

The non-bijectivity of the phoneme-grapheme correspondence causes difficulties in generating the written representation from spoken words. The correct grapheme for the written representation of a phoneme in a given word can not be determined by attentive listening, but can only be learned by heart or constructed using language specific rules. E.g. the phoneme /a:/ in *Tal*

([ta:l], engl. valley), *Z**ahl*** ([t_sa:l], engl. number) and *S**aal*** ([sa:l], engl. hall) is pronounced exactly the same way, but is represented by three different graphemes (*'a'*, *'ah'* and *'aa'*).

5.3 Error Taxonomy

In this section we describe the error categories relevant for isolated word spelling of dyslexic children. The taxonomy is structured according to the requested information to detect the underlying source of the error (see Figure 5.2).

5.3.1 Capitalization

Capitalization (Cap) errors are upper and lower case confusions, which occur more in the German language versus other Western languages. Over 15% of the spelling errors made in scholarly essays are capitalization errors [Aug85]. The German-specific difficulty is that letters are not only written uppercase if they are at the beginning of a sentence or a name, but also every noun is capitalized. However, detecting capitalization errors is simple and unambiguous, and requires only local information about correct and error letters.

5.3.2 Typing Error

Typing errors (Typo) are errors accidentally committed due to typing difficulties. Their occurrence can be characterized as unsystematic and self-corrigible. James states that typing errors can be divided into spatial and temporal errors. In spelling software utilization we mostly face users of the second or third grade of school, which do not master the touch typing system yet. Due to their slow average typing speed, permutations of the letter sequence (temporal errors) like *s**eh**r* (engl. very) to *s**he**r*, are not good indicators for typing errors. The empirical data collected in the first user study evidences that the probability of typing errors strongly depends on the distance between correct and error letters on the keyboard (spatial errors), e.g., *s**e**hr* to *a**e**hr*. Therefore, a detection of the typing error category requires only correct and error letters, and is dependent on the input device used for the training.

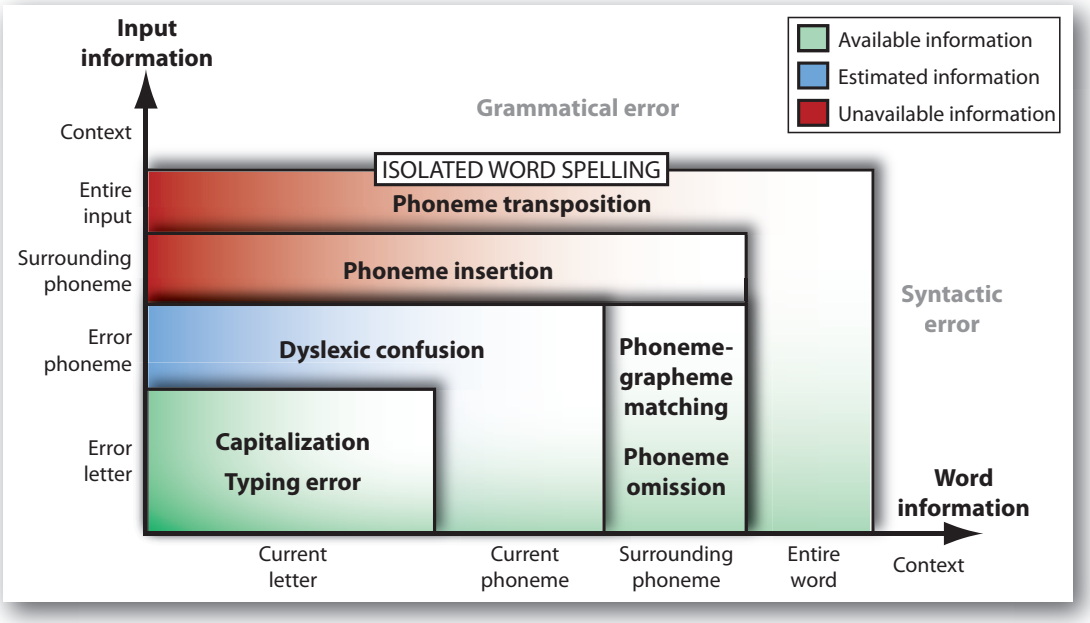

Figure 5.2: *Error taxonomy for isolated word spelling, structured according to the information required about correct word and student input to detect the error category. The correct word and its phonological representation are known. Reliable information about the erroneous input is only available with respect to the error letter. Additionally, the error phoneme can be estimated, but the surrounding phonemes and the entire input are not available.*

5.3.3 Dyslexic Confusions

Dyslexic confusions (DysC) denote permutations of letters or phonemes. These can occur due to visual similarity of letters, e.g., *'d'-'b'*, or due to an auditory similarity of corresponding phonemes, e.g., /n/-/m/. The visual confusion can be detected on a letter level, e.g., the horizontally mirrored image of *'d'* equals *'b'*. Confusions caused by an auditory similarity require information about the current correct and error phoneme. The underlying difficulty of confusing Niete [ni:te] (engl. rivet) with Miete [mi:te] (engl. rent) is only revealed on a phonological level.

5.3.4 Phoneme-Grapheme Matching

Phoneme-grapheme matching (PGM) is one of the most frequent error categories. PGM errors are caused by the non-bijectivity of the phoneme-grapheme correspondence, i.e., many phonemes can be represented by mul-

tiple graphemes. In German, this non-bijectivity can be divided into three subcategories:

- *Elongation:* Several vowel phonemes have a short and multiple elongated grapheme representations. For example, the phoneme /aː/ can be represented by the grapheme *'a'*, as well as by the graphemes *'aa'* and *'ah'*. This doubling of vowels and the additional silent *'h'* are called elongation. These different grapheme representations appear, e.g., in *T**a**l* (engl. valley), *S**aa**l* (engl. hall), and *Z**ah**l* (engl. number).
- *Sharpening:* Doubling of consonants, such as *'s'-'ss'*, *'n'-'nn'* or *'p'-'pp'*, are called sharpening. *ke**nn**en* (engl. to know) is an example containing two different grapheme representations of the same phoneme /n/.
- *Phoneme Matching:* The remaining phoneme-grapheme correspondences are simply named phoneme matching. Common error sources are the phonemes /ɔ_y/ (*'äu'*, *'eu'*) and /f/ (*'v'*, *'f'*).

The different grapheme representations of one phoneme are pronounced the same way. The correct one cannot be determined by careful listening alone. The phoneme-grapheme correspondence must be deduced from language rules, or learned by heart. PGM errors become manifest in the local environment of an error. To detect these errors, the surrounding phonemes of the correct word and the current error phoneme needs to be known, as will be shown in Section 5.4.

5.3.5 Phoneme Omission

The error of omitting a complete phoneme while entering a word is called phoneme omission (PhoO). It is a common error among dyslexic children. A typical cause of omitting an entire phoneme is, e.g., the phoneme /ɛ/ in the word *Verein* [fɛɐain] (engl. club). To detect a phoneme omission, the information about surrounding correct phonemes and current error phoneme is required.

5.3.6 Phoneme Insertion & Phoneme Transposition

The insertion of an additional error phoneme and the transposition of two phonemes in a word are called phoneme insertion and phoneme transposition respectively. Errors of these two categories become manifest in the entire

input only, which is not available to the error analysis. Therefore, these error categories are not detectable from the available student inputs.

5.4 Mal-Rules

In this section we present our set of independent mal-rules, which allow for a detection of the previously introduced error categories. The design of mal-rules is driven by insights gained from preliminary investigations of the user input data. It is conditioned by the spelling software specific setting and objective: (1) due to the immediate feedback, the error analysis is limited to the input up to the error letter; (2) the target goal of representing the student spelling knowledge based on the set of mal-rules demands an as simple and high-level description of errors, which still accounts for the diverse spelling difficulties of dyslexic children.

The mal-rules described in the following are divided into a letter and a phoneme level according to the information they operate on.

5.4.1 Letter Level

Capitalization

Capitalization errors are very simple and unambiguous to detect. A capital letter can be typed lower case and a lower case letter can be typed upper case. We introduce the two mal-rules:

- *ToLowerCase (binary):* Typing a capital letter lower case.
- *ToUpperCase (binary):* Typing a lower case letter upper case.

Typing Error

The mal-rules for the typing error category are chosen with respect to the average user age and typing abilities:

- *KeyDistance (categorical):* The spatial distance between correct and error key. The student data collected in the first study shows that the typing error probability for all keys more distant than the surrounding keys of the correct letter does not differ significantly from zero. Therefore, we discarded attempts to model the error probability - key distance relation by, e.g., a Gaussian decay function and introduced

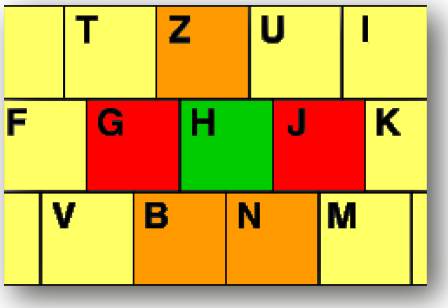

Figure 5.3: *Different distance categories for the typing error mal-rules: Left/Right, Top/Bottom, and Distant.*

three categories of key distances to avoid a non-linear optimization problem. As shown in Figure 5.3, we have the most error-prone category *Left/Right*, the keys to the top, on the bottom right and bottom left are combined to *Top/Bottom*, and all the other keys belong to the category *Distant*.

- *Technical (binary):* German umlauts cause problems to type on the keyboard. From handwriting, the children are used to write the vowel first and subsequently put additional dots on top of it. This causes a high confusion rate between umlauts and their corresponding unmutated vowels. To model this input device-specific difficulty, we introduce a binary mal-rule *Technical*.

Visual Dyslexic Confusion

Researchers investigated many approaches to model the visual similarity of letters, e.g., by geometric moments of letters [LKX+09] or based on empirically collected confusion probabilities [BC89]. However, they either studied difficulties in letter recognition at the acuity limit or used fonts unequal to those taught in Swiss schools. Our measure is based on the findings that simple difference measures in images are a good approximation to the letter similarity [Blo88].

- *VisualSimilarity (continuous):* We introduce a visual similarity mal-rule based on the cross-correlation between images of letters. This is computed using the CH3 Steinschrift font, which corresponds to the hand writing taught in Swiss schools (see Figure 5.4). Since most letters in the written language are lower case, the confusion

Figure 5.4: *CH3 Steinschrift font used for the visual similarity measure of letters.*

should be calculated on their lower case representation. However, due to the capitalized letters on the keyboard, we compute three visual distances. One between lower case letters VS(LowerCase), one between capitals VS(UpperCase), and additionally a distance between lower and upper case letters VS(L/UCase).

Empirical data of dyslexic children demonstrates that letters are also confused more frequently, if the feature a high visual similarity when mirrored horizontally, like the common *'d'-'b'* confusion. To evaluate the visual similarity of a letter pair, we therefore take the maximum of the two cross-correlation values of actual and horizontally mirrored image.

5.4.2 Phoneme Level

Auditory Dyslexic Confusion

The auditory similarity of phonemes can lead to confusions. A common example are the two auditory similar nasal phonemes /n/ and /m/. In the following, we first present how we estimate the error phoneme from the available input string. Second, we investigate two auditory distances between phonemes and describe how we finally model auditory similarity.

Error Phoneme Identification To determine the auditory distance between correct word and erroneous input, we need the correct and the error phoneme. The correct phoneme is available since we know the entire correct word. However, the identification of the error phoneme is not unambiguously possible. The non-bijectivity of the phoneme-grapheme correspondence in German inhibits the definite determination of the error phoneme. A letter can be a fragment of multiple graphemes, which themselves can belong to several phonemes. E.g. the letter *'d'* can represent the grapheme *'d'* or be

a part of the grapheme *'dt'*. The grapheme *'d'* again is a representation of the phonemes /d/ and /t/. This can be seen in the words *Bün**d**nis* [byn**t**nɪs], *La**d**ung* [la:**d**ʊŋ] and *Sta**dt*** [ʃta**t**].

To identify the error phoneme, we search for the closest possible grapheme with respect to a given auditory metric, in order to receive the most probable auditory error cause. If the error grapheme is a representation of the correct phoneme, the auditory distance is considered as zero. These cases are characterized more detailed by the phoneme-grapheme matching mal-rules.

Signal-based Metric In a first approach, we investigate the sound of phonemes on a signal-based level. The idea is to find a metric in the signal or frequency space, which corresponds to the perceived distance of phonemes. For that purpose, we extract sound snippets of phonemes from the prerecorded dictations of Dybuster. The raw signal of these spoken representations can be further processed to a frequency spectrum or to a parametric representation using the perceptual linear predictive (PLP) technique [Her90] or the mel-frequency cepstrum coefficients (MFCC) [ZZS01].

Based on a given parameterization we try to define a metric using heuristic distance measures or classification routines. For example, based on a linear discriminant analysis [Bis06] we are able to determine, how difficult it is for a statistical method to distinguish two given phonemes. However, there are two main difficulties with this approach:

1. It's a strong assumption to suggest an auditory similarity of phonemes in human hearing based on the classification difficulties of a statistical algorithm.
2. As described in Section 5.2, a phoneme is the smallest phonetic unit capable of conveying a distinction in meaning. This means, that phonemes can be pronounced in slightly different ways, as illustrated in the *ich* and *ach* example. However, this can strongly influence features extracted from the auditory signal.

For these reasons, we favor a articulation-based metric.

Articulation-based Metric In a second attempt we propose an auditory distance based on sound characteristics. Phonemes have articulation-based attributes, which allow to structure the phonemes in a hierarchical order (see Figure 5.5). Our approach is based on the hierarchical phoneme structure proposed by Dekel et al. [DKS05]. The assignment of phonemes to nodes

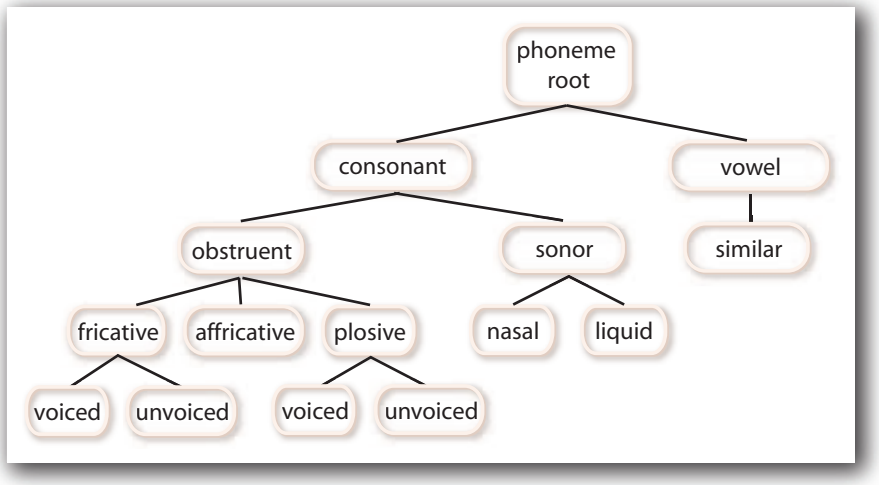

Figure 5.5: *Hierarchical phoneme structure: Every phoneme is assigned to one of the leaf nodes, each representing a phonetic attribute.*

can be found in Appendix B. We modified the structuring of vowels to better address our findings regarding vowel confusion probabilities in the user data.

The attributes *Front/Center/Back* introduced by Dekel et al. are not relevant vowel features regarding the collected user data. To assign an auditory distance to vowel pairs, we employ the so-called vowel triangle [Hal00]. In Figure 5.6 the vowels are positioned according to their first and second formant frequency. These are the main resonance frequencies of the vowels. The Mel scale [Ped65] is used for the frequencies to visualize the perceptual distance.

$\mathbf{f_1}$	$\mathbf{f_2}$	**vowel**	/o/	/a/	/ɘ/	/y/	/ɛ/	/e/	/i/
320 Hz	800 Hz	/u/	1.96	0.29	0.06	T	0.09	0.66	KD
500 Hz	1000 Hz	/o/		1.45	T	0.11	0.1	0.45	KD
1000 Hz	1400 Hz	/a/			0.08	0.25	T	2.19	0.31
500 Hz	1500 Hz	/ɘ/				T	T	0.08	0.17
320 Hz	1650 Hz	/y/					T	0.19	0.08
700 Hz	1800 Hz	/ɛ/						1.42	0.31
520 Hz	2300 Hz	/e/							1.19
320 Hz	3200 Hz	/i/							

Table 5.1: *Vowel confusion probabilities [‰]: The first and second formant frequency of every vowel are given on the left. The table on the right shows the empirical confusion probabilities in per mill averaged over all children. T and KD indicate the interference with the two features Technical and KeyDistance.*

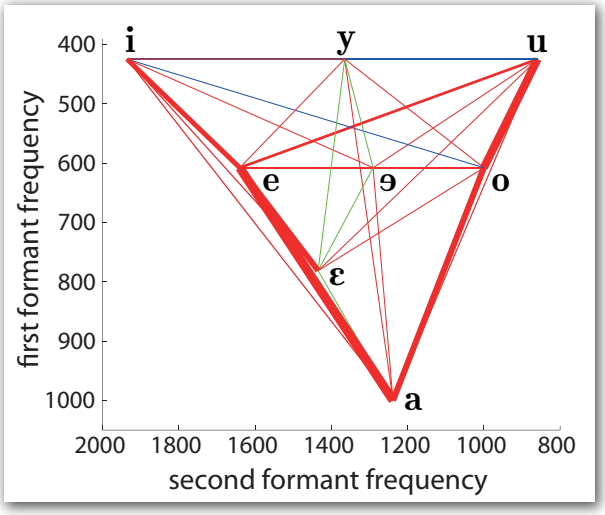

Figure 5.6: *Vowel triangle: Vowels are positioned with respect to their first and second formant frequencies on the Mel scale (Fant, 1960). Empirical confusion probabilities are represented by the thickness of the connecting red lines. Blue and green lines indicate a significant influence of other features (key distance and technical).*

The thickness of the vowel-connecting lines represents the confusion probabilities found in the user data given in Table 5.1. Confusion possibilities influenced by additional factors like key distance (green line) or other input device specific difficulties (blue line) are not taken into account. It can easily be seen, that confusions between nearby phonemes along the edges /i/-/a/ and /a/-/u/ of the vowel triangle are more likely to happen, and are thus labeled as similar. Between vowel pairs, connected across the vowel triangle, hardly any confusions occur.

The auditory distance of diphthongs requests a special treatment, since they represent transitions from one vowel to another. We check on which letter position the confusion occurred and threat the diphthong like the corresponding vowel.

- *Auditory Similarity (categorical):* We define the auditory similarity (AS) mal-rule as a categorical feature representing the nearest common ancestor node of the correct and the closest possible error phoneme.

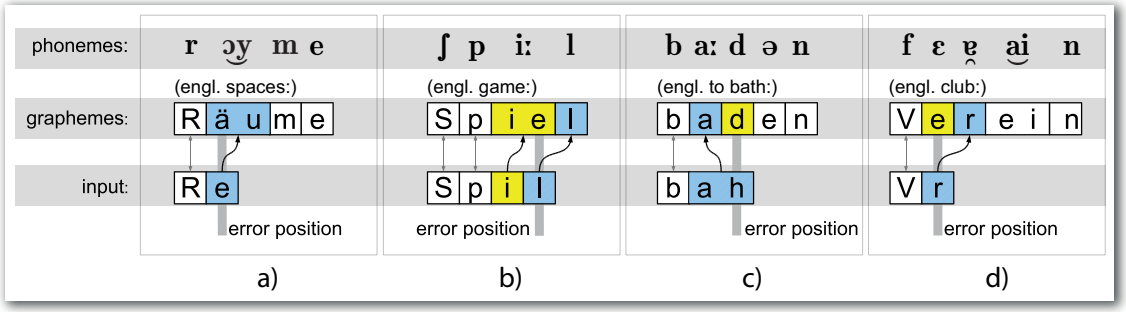

Figure 5.7: *Alignment of correct and input phonemes and resulting mal-rules: a) Phoneme matching b) Letter omission c) Letter addition d) Phoneme omission*

Phoneme-Grapheme Matching

To detect phoneme-grapheme matching (PGM) errors, we align the user input and the phonological structure of the correct word. Phoneme-based metrics between strings have been defined using graphonemes [Ver88] or based on phonological sting alignment [Kon03]. However, such methods request the entire string of input and correct word. To allow for an identification of PGM errors, only based on the input up the error letter, we use a novel, local alignment of the phonological structure of input and correct word. The algorithm compares the error phoneme with the current, the following, and the previous phoneme:

- *PhonemeMatching:* A false letter can be part of a wrong representation of the correct phoneme. E.g., in Figure 5.7.a the false letter *'e'* is the beginning of the grapheme *'eu'*, which is a representative of the correct phoneme /ɔ_y/.
- *LetterOmission:* If a false letter marks the beginning of the next phoneme and the current phoneme is falsely represented by the previous input grapheme, we face a *LetterOmission*, such as in Figure 5.7.b: the error letter *'l'* matches the following phoneme, and the current phoneme /iː/ is incorrectly represented by the grapheme *'i'*.
- *LetterAddition:* The previous input grapheme concatenated with the false letter can match the previous phoneme. In Figure 5.7.c the false letter *'h'* appended to the previous input grapheme *'a'* results in the grapheme *'ah'* - which is a representative of the previous phoneme /aː/.

To discriminate the errors in greater detail, we further subdivide the mal-rules presented above. In *PhonemeMatching*, we distinguish between *Vowel* and *Consonant* phonemes as well as between *Main* and *Special* graphemes. The attributes *Main* and *Special* are manually attached to every grapheme. They indicate whether a grapheme is the most common (*Main*) representative of the phoneme or an unusual (*Special*) one. *LetterOmission* and *LetterAddition* are both subdivided into *Elongation* and *Sharpening* based on the type of phoneme the error occurred in (*Vowel*/*Consonant*). These binary mal-rules are language specific and the phoneme-grapheme correspondence has to be adapted, if the mal-rules are applied in other languages.

Phoneme Omission

Similar to the PGM errors, we can align user input and phonological structure of the correct word.

- *PhonemeOmission (binary):* If a false letter marks the beginning of the next phoneme and the current phoneme is completely omitted, we detect a *PhonemeOmission*. As displayed in Figure 5.7.d), the incorrectly entered grapheme *'r'* matches the following phoneme /ɐ/. The current phoneme /ɛ/ has been omitted.

5.4.3 Feature Vector

The presented mal-rules allow for a detailed description of errors. Each provides error information in one of the three following forms:

- *Binary:* A mal-rule has been applied or not. E.g., writing a letter in false case (capitalization).
- *Categorical:* The described mal-rule comprises multiple states. E.g., *KeyDistance* (typing error) contains confusions with letter to the left or right (*Left/Right*), with letters to the top or bottom (*Top/Bottom*) or with distant keys (*Distant*).
- *Continuous:* Mal-rules, such as *VisualSimilarity*, provide a continuous value for a given error. This indicates to what extent a mal-rule is activated in an error.

For a given error *e*, each mal-rule returns one or several activation values. In the binary case, this is simply one or zero. Categorical mal-rules are partitioned into multiple, binary dummy variables. In the continuous case, we obtain a value between zero and one, indicating the activation level of the

Figure 5.8: *The confusion of the letter 'm' and 'n' could be due to a doubling of the letter 'n', due to a confusion of similar phonemes* /m/ *and* /n/*, or due to the small key distance of 'm' and 'n', which leads to an activation of the mal-rules Sh(Addition), AD(Nasal), and KD(Left/Right).*

mal-rule. All of the presented mal-rules $f_i(e)$ together describe a single error and are represented in a feature vector $\mathbf{f}(e)$. Notably, several mal-rules can be activated at the same time for one error e, indicating that all of them would generate the same error pattern, as illustrated in Figure 5.8. This example shows that some error categories exhibit identical symptoms and thus cannot be classified unambiguously, not even by a human.

5.5 Conclusion

In this chapter we introduced our error taxonomy for isolated word spelling and a set of mal-rules to describe errors. Due to the specific setting of spelling software, the taxonomy is structured with respect to the requested information to identify an error category. The design of the mal-rules is driven by the insights gained from the collected user data. They account for spelling difficulties on a letter as well as a phoneme level.

However, the ambiguity in the error description, i.e., the fact that multiple independent mal-rules can lead to the same error pattern, prevents a definite assignment of errors to categories. In the next chapter we will approach this ambiguity by a statistical model of spelling knowledge, which will provide a spelling error classification.

CHAPTER 6

Spelling Knowledge Representation

The set of mal-rules introduced in the previous chapter allows for a detailed description of errors. However, as seen in the *Unmut* example in Figure 5.8, several mal-rules can generate the same error, which makes an unambiguous assignment of errors to categories impossible.

In this chapter we address this problem by a statistical student knowledge representation for perturbation models. The inference algorithm is designed to estimate error rates based on unclassified input with multiple errors described by independent mal-rules. We identify an inference algorithm based on a Poisson regression with a linear link function as most suitable. This spelling knowledge representation models the student-specific spelling difficulties and allows for a classification and prediction of errors. The information provided by this knowledge representation will serve as the basis for a more detailed data analysis (see Chapter 8 and 10), and the development of an enhanced word selection controller described in Chapter 9.

6.1 Overview

In collaboration with elementary school teachers and psychologist, we identified the demand for information on two levels to adapt spelling training appropriately to students' needs:

- *Local information:* Error localization and classification to enable adequate remediation on erroneous inputs.
- *Global information:* Spelling knowledge of student to allow for optimized word selection based on further spelling performance prediction on the entire word database, and for feedback to human tutors on students' strengths and weaknesses.

In this chapter we present the student knowledge representation for spelling, which provides the requested local and global information about a student. We decided to design a perturbation model for spelling to avoid a specification of the strongly differing spelling process of children and rather rely on mal-rules, which describe errors.

In a perturbation model the student is typically represented by error probabilities for each of those mal-rules. There are two factors which brings us to take a different viewpoint on spelling errors: (1) the immediate feedback on committed errors and the subsequent correction by the student allows for multiple errors in one single task, i.e., it enables multiple entries of the same error key at one single letter position; (2) the assumption that student attributes are not either in a learned or unlearned state, but rather represent different gradation of spelling difficulties. Therefore, we regard errors as randomly occurring events and try to estimate their rate of occurrence dependent on the mal-rules involved in an error.

Randomly occurring events are best described by a Poisson distribution, which is characterized by the expected number of events that occur during a given time interval, or a given number of exposures to risk. In the case of spelling, every possible error in a prompted word is considered as an exposure to risk and we are interested in the corresponding error expectation values. By means of a Poisson regression we estimate the error rate of mal-rules for every student based on its currently available input data. The employment of a linear link function in the Poisson regression assures the independence of the presented mal-rules.

As illustrated in Figure 6.1, based on the estimated error rates we can predict the further spelling performance on the entire word database and provide a probabilistic classification of committed errors. This local and global information about a student can be used to adapt the training to the student needs and to choose appropriate remediation actions. In the following, we will first describe the data and how the student parameters are inferred from it. Then, we investigate the significance of individual mal-rules. At the end of the chapter, we describe the error classification and prediction, give an example of use and evaluate the model based on the data of the first user study.

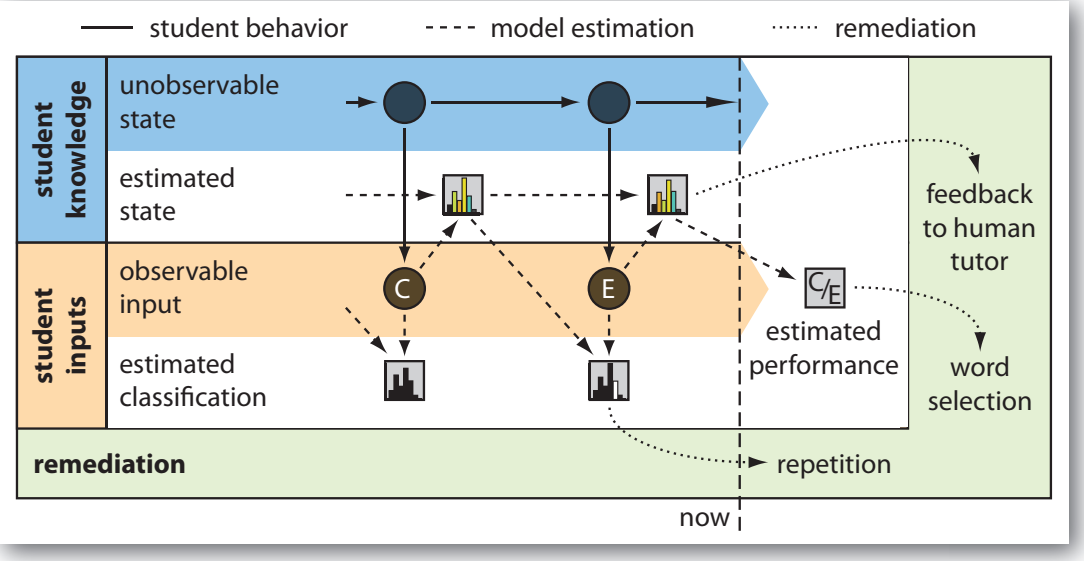

Figure 6.1: *Workflow of the student model: The dark circles at the top represent the unobservable knowledge state of a student progressing over time. The model can only observe the student inputs, which can either be correct (C) or erroneous (E). Each input allows for an update of the student knowledge representation, indicated by the colored bars in the rectangle. This enables a classification of subsequently committed errors, and a prediction of the further spelling performance. Appropriate remediation actions can be conducted on a local level (repetition), or a global level (word selection and feedback to human tutors).*

6.2 Data Collection

The student model, representing the student's difficulties with individual mal-rules, has to be estimated out of the available input data of a student. We analyze every input of a student and distinguish between *possible errors*, i.e., error inputs which could potentially be entered at the given word, and *committed errors*, i.e., error inputs which the student actually entered during the training. Each letter of a word contains 29 possible errors, namely one capitalization and 28 confusions with all other letters. The collection of possible and committed errors for the *Unmut* error example (see Figure 5.8) is illustrated in Figure 6.2.

Every possible error e of all words from the database could be considered as an item on which as student can be tested on. There is an average of 6.7 letters per word and a database of 1 500 words, which yields approximately 300 000 test cases. The presented mal-rules allow for a detailed description

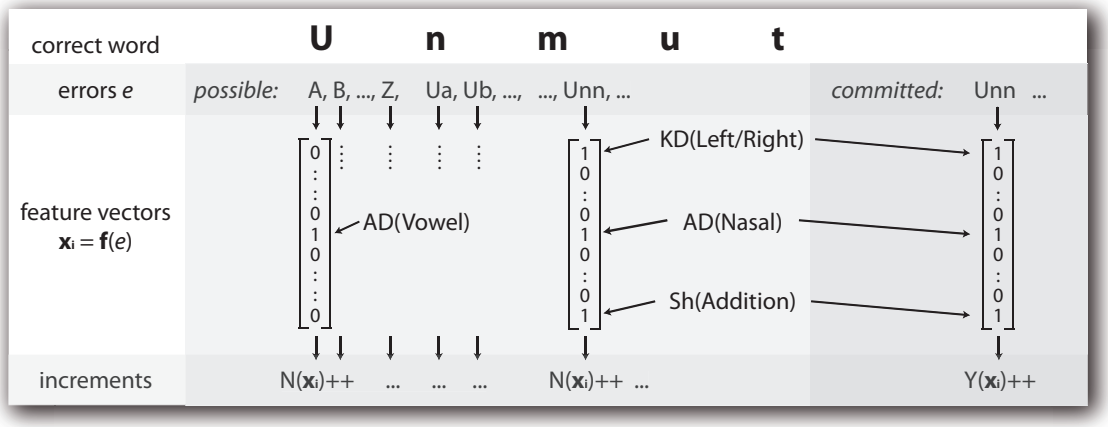

Figure 6.2: *Data collection for the prompted word Unmut: Every letter of the word is permuted with each letter from the alphabet, the feature vector* $\mathbf{x}_i = \mathbf{f}(e)$ *describing this possible error e is computed, and the corresponding occurrence count* $N(\mathbf{x}_i)$ *is incremented. For every error e the student actually committed in this word, we compute the feature vector* $\mathbf{x}_i = \mathbf{f}(e)$ *and increment the corresponding error count* $Y(\mathbf{x}_i)$.

of these items. The feature vector $\mathbf{f}(e)$ indicates the extent to which each mal-rule is activated in an error e. Tatsuoka proposed a **Q**-matrix, in which all the precomputed feature vectors for every item are assembled [Tat85]. To avoid the assembly of an approximately $300\,000 \times$number-of-mal-rules **Q**-matrix, we do not store the feature vector for every item, but continuously compute and assemble only the feature vectors $\mathbf{x}_i = \mathbf{f}(e)$ which were actually presented to the student. For each $\mathbf{x}_i$ we count the number of times an error possibility described by the feature vector $\mathbf{x}_i$ was encountered ($N(\mathbf{x}_i)$), and how often such an error was actually committed by the student ($Y(\mathbf{x}_i)$). This reduces the number of different feature vectors down to 363 for the input data of the first user study.

6.3 Inference Algorithm

The inference algorithm has to estimate the student's difficulties with each individual mal-rule, based on the observed error behavior described by $Y(\mathbf{x}_i)$ and $N(\mathbf{x}_i)$ for all $\mathbf{x}_i$. Due to the possibility of multiple errors, we consider errors as randomly occurring events, which are best described by a Poisson distribution.

Therefore, the probability distribution for the error count $Y(\mathbf{x}_i)$ is defined as:

$$P(Y(\mathbf{x}_i)) = \frac{e^{-\mu(x_i)N(\mathbf{x}_i)}(\mu(\mathbf{x}_i)N(\mathbf{x}_i))^{Y(\mathbf{x}_i)}}{Y(\mathbf{x}_i)!}$$

where $\mu(\mathbf{x}_i) > 0$ denotes the rate parameter and $N(\mathbf{x}_i)$ the number of exposure to risk. The expectation value of $Y(\mathbf{x}_i)$ in a Poisson distribution is given by $\mathbb{E}[Y(\mathbf{x}_i)] = \mu(\mathbf{x}_i)N(\mathbf{x}_i)$. The error rate $\mu(\mathbf{x}_i)$ has to be related to the unknown student parameters $\boldsymbol{\beta}$ and the feature vector $\mathbf{x}_i$ by a link function $g(.)$:

$$g(\mu(\mathbf{x}_i)) = \boldsymbol{\beta}\mathbf{x}_i$$

The Poisson regression to estimate the student parameters $\boldsymbol{\beta}$ is part of the family of generalized linear models [MN89]. McCullagh and Nelder propose the canonical log link function for the Poisson regression, which has the beneficial property of matching the domain of the link function with the range of the non-negative rate parameter. However, they state that, since the log link function induces a multiplicative effect on the mal-rules, the appropriateness of this selection has to be tested. To make allowance for the independence of the presented mal-rules, we request an additive behavior of the error rates of individual mal-rules. For example, if a given error activates a Typo and a PGM mal-rule, we expect the error rate on such an error possibility to be the sum of the error rates of the Typo and the PGM mal-rules. This is due to the fact that these two error types are independent and act on distinct sections of the spelling process (remember Figure 2.1). Therefore, we propose the usage of a linear link in the presence of independent mal-rules.

To compare the effect of the two link functions we employ the Tukey-Anscombe plot for residual analysis [AT63]. The plot shows the inter-relation of fitted values of the method used and its deviance residuals. The residuals should be normally distributed with expectation zero, and a constant variance on the entire scale of fitted values. The analysis is performed on the input data of all children of the first user study. As can be seen in Figure 6.3 (left), the non-zero error expectation value (red line) of the log link clearly depicts the violation of the independence assumption of individual mal-rules. Low error expectation values with single mal-rules activated are rather underestimated (positive residuals), and the combination of mal-rules results in overestimated error expectation values (negative residuals). As Figure 6.3 (right) shows the linear link provides a zero expectation value across the entire fitted value scale. One can see how reasonable estimates are assured even for the higher fitted values. This effect is additionally illustrated by the error example in Figure 6.4. The error has the two mal-rules *KD(Left/Right)* (typing error) and *PM(ConsSpec)* (PGM error) activated. Table 6.1 shows the estimated

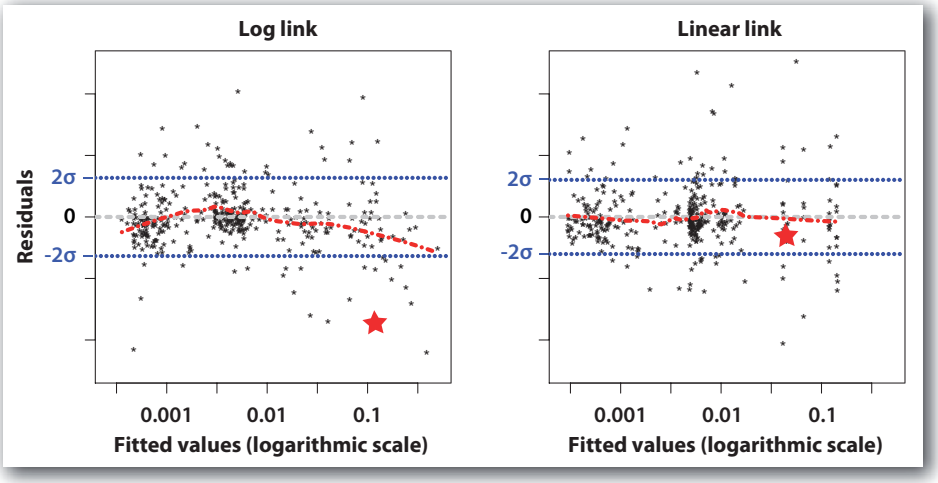

Figure 6.3: *Tukey-Anscombe plot for log (left) and linear (right) link function on a logarithmic scale. σ indicates the median absolute deviation of the residuals and the red line shows the LOWESS smoothed expectation value of the residuals [Cle79]. The log link method suffers from a non-zero error expectation (red line), due to the multiplicative interdependence of the mal-rules. This is depicted (red star) by the Netz - `Nez` error example (see Figure 6.4).*

error probabilities for an error with only *KD(Left/Right)* or *PM(ConsSpec)* activated and the combination of both, as it is the case in the presented example. The multiplicative interrelation of mal-rules of the log link leads to an overly pessimistic estimation of the error probability (0.116 compared to the measured 0.036). The residual of the estimation is visualized by a red star in Figure 6.3. In contrast, the error rates of the linear link function show an additive behavior. The estimated error expectation (0.045) for errors having both features activated corresponds more closely with the empirical expectation value (0.036) as shown in Table 6.1.

This error example indicates the importance of the additive interrelation of mal-rules. The remediation actions will focus on the most severe weaknesses

Method	BIC	KD(Left/Right)	PM(ConsSpec)	Both activated
Log link	−10295	0.002	0.028	0.119
Linear link	−10895	0.004	0.041	0.045
Empirical error expectation value				0.036

Table 6.1: *BIC score and estimated error expectation values for log and linear link. As a comparison, the empirically measured error expectation value for the Netz-`Nez` error example are given.*

Figure 6.4: *The omission of the letter 't' could be due to a phoneme-grapheme matching error ('tz' and 'z' are both representatives of the phoneme /t_s/) or due to the small key distance of 't' and 'z' (QWERTZ keyboard).*

of the student. Therefore, if the log link function leads to strongly overestimated error rates of errors with multiple mal-rules activated, then those errors will experience an undesired bias in the subsequent training. The appropriateness of the linear link function can additionally be evaluated by comparing the two regression models based on the Bayesian information criterion [Bis06]. The BIC score is an approximation to the model evidence and serves as a measure of the goodness of fit for statistical models. Lower scores indicate a better representation of the data. The log link and the linear link model yield an BIC score of -10295 and -10895 respectively, which shows the superiority of the linear link in the context of independent mal-rules.

6.4 Significance of Mal-Rules

To evaluate the significance of individual mal-rules, we employ the likelihood ratio (LR) test [CT98]. This test returns the log-ratio of the likelihood of the full model to the likelihood of a model with the given mal-rule left out. These values asymptotically follow a χ^2 distribution with one degree of freedom. The computation of the likelihood of a Poisson regression depends on the dispersion parameter ϕ [MN89]. The models fitted into the data of individual students show a mean over-dispersion of 3.1. The likelihood of each model is corrected with the respective dispersion ϕ.

Not every student struggles with all difficulties represented by the set of mal-rules. To justify the appropriateness of a mal-rule, we investigate the highest LR score across all $N = 54$ students of the first study, to determine whether the mal-rule is significant at the $\alpha = 5\%$ level for at least one of the students. Consequently, we have to apply a false discovery rate correction. The Šidàk correction ($\alpha_c = 1 - (1 - \alpha)1/N$, [Abd07]), well suited for the independent tests of individual students, yields a corrected significance level α_c of 0.095%.

As can be seen in Appendix C, most mal-rules are highly significant. This is especially true for all capitalization, phoneme-grapheme matching and phoneme omission features. In the dyslexic confusion error category, 4 out of 14 auditory confusion nodes from the hierarchical phoneme structure are not significant. This is mainly due to the fact that most confusions of fricative and fluid sounds are very sparsely sampled. We estimate the closest possible phoneme for the auditory confusion and many fricative and fluid phonemes are represented by the same graphemes. This often results in a detection of phoneme-grapheme matching errors rather than auditory confusions.

An interesting finding is the non-significance of the visual similarity mal-rules for both lower and upper case letters. This indicates that dyslexics rather suffer from auditory and phonological processing deficits, than from an impaired visual processing. This corresponds well with recent findings in dyslexia research [Ram03].

6.5 Error Classification and Prediction

The spelling knowledge representation provides an estimate of the student's difficulties on individual mal-rules. During the training, the representation of the student's mastery of the domain is continuously updated after each entered word. Based on these estimates we can compute a prediction of further spelling performance and a classification of committed errors for each individual student. This information is expressed by the following two values:

- $\mathrm{P}_S(k|\mathbf{f}(e))$: The probability that the k^{th} mal-rule is the source a committed error e.
- $\mathbb{E}[E|w]$: The expected number of errors a student will make on the spelling of the word w.

To provide the probability of the k^{th} mal-rule being the cause of an error described by $\mathbf{f}$, we employ Bayes' theorem. $\mathrm{P}(\mathbf{f}|k)$, the probability that an error occurs if the k^{th} mal-rule is causing an error, is always equal to 1.

$$\mathrm{P}_S(k|\mathbf{f}) = \frac{\mathrm{P}(\mathbf{f}|k)\mathrm{P}(k)}{\mathrm{P}(\mathbf{f})} = \frac{\beta_k f_k}{\boldsymbol{\beta}\mathbf{f}}$$

The estimated error rates $\boldsymbol{\beta}$ are used to specify the prior probability $\mathrm{P}(k)$ of the k^{th} mal-rule causing an error. The probability $\mathrm{P}_S(k|\mathbf{f})$ corresponds to the part of the k^{th} mal-rule on the total error expectation of a given error described by $\mathbf{f}$. In the following analyses we use the maximum a posteriori, i.e., the most likely category to classify errors.

The expected error count $\mathbb{E}[E|w]$ for the word w can be obtained by summing over the error expectation values of the errors contained in the set $\mathrm{C}(w)$ of all possible confusions in the word w:

$$\mathbb{E}[E|w] = \sum_{e \in \mathrm{C}(w)} \mu(\mathbf{f}(e)) = \sum_{e \in \mathrm{C}(w)} \boldsymbol{\beta}\mathbf{f}(e)$$

This allows for a prediction of the spelling performance on every word in the entire word database, even if the word has never been prompted so far. The given formulas indicate that the classification and prediction of errors is dependent on the individual student parameters $\boldsymbol{\beta}$. Varying student characteristics influence the determination of the most likely cause of an error and change the error expectation values, as illustrated in the following section.

6.6 Example of Use

The presented student model provides an error classification and a prediction of further spelling performance, dependent on the student characteristics. In this section, we present the spelling knowledge representation and its application for three selected students of the first user study, to show the influence of varying student parameters. Subject 1 is dyslexic and has strong difficulties with capitalization and dyslexic confusions. Subject 2 belongs to the control group and has the highest error rate for typing errors (*KD(Left/Right)*) of the three children. The main difficulties of the dyslexic subject 3 are the phoneme-grapheme correspondences. Especially elongation and sharpening are the cause of many errors committed by subject 3. The student parameters of the three selected subjects are illustrated in Figure 6.5. As the logarithmic scale indicates, the error rates of mal-rules vary by orders of magnitudes. For example, the *KD(Left/Right)* and *ToLowerCase* mal-rules have an error rate around 0.005 and 0.1 respectively. Nevertheless, the *KD(Left/Right)* mal-rule is still relevant, since it is activated for two confusions for every letter in a word. A word, such as *Männer* (engl. men), contains 12 *KD(Left/Right)* confusion possibilities, but only one for the capitalization mal-rule *ToLowerCase*.

Figure 6.6 (left) shows the probabilistic classification of the *Unmut* error example (remember Figure 5.8). Due to the high confusion rate of nasal phonemes (*AS(Nasal)*), the error is classified as an auditory dyslexic confusion for subject 1. Subject 2 shows less difficulties with auditory similarities and sharpening errors, but high typing errors rates. Therefore, the error is classified as typing error (*KD(Left/Right)*) for subject 2. The high error rate at *Sh(Addition)* makes a phoneme-grapheme matching error most likely

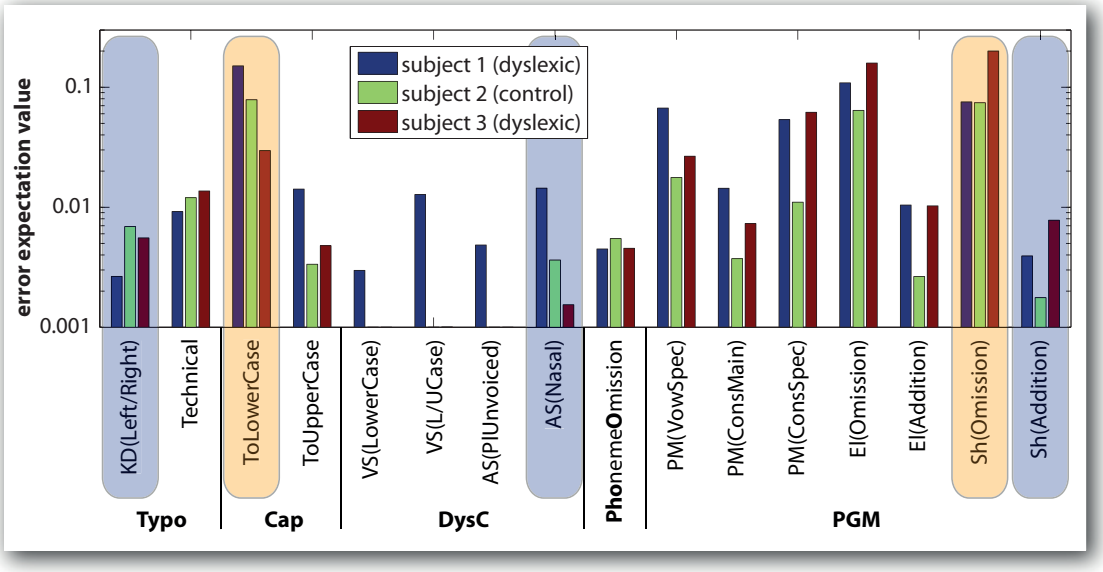

Figure 6.5: *Student characteristics for three subjects of the first user study. All estimated parameters β with a value above 0.001 for at least one of the three students are displayed on a logarithmic scale. Mal-rules relevant for the error classification (blue) and prediction (orange) examples are highlighted.*

for subject 3. Similarly, we obtain varying error expectations for the three students. Figure 6.6 (right) shows the error expectation for each letter of the word *Männer* (engl. men). The first letter contains a capitalization possibility (*ToLowerCase*). The strong difficulty of subject 1 on capitalization results in an error expectation of almost 0.2, in contrast to 0.1 and 0.05 for subject 2 and 3 respectively. However, the error expectation at the second *'n'* is estimated twice as high for subject 3 compared to subject 1 and 2, due to the high error rate on sharpening (*Sh(Omission)*). These examples show how different student characteristics influence the error classification and prediction. This information allows for an adaptation of the spelling training to individual students, as will be described in Chapter 9.

6.7 Validation

As illustrated above, the student model provides a classification and prediction of errors for individual students. To verify the determined cause and expected number of errors, we need the true underlying source and expected number of errors. However, as shown in the error example in Figure 5.8, some errors are not unambiguously classifiable, even by a human. Due to the lack of a ground truth of the error classification, we investigate the

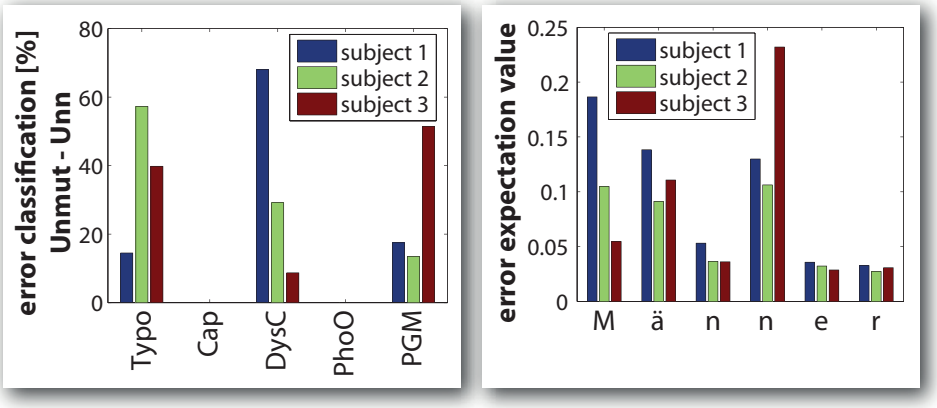

Figure 6.6: *Application of the student model for three students: (left) Probabilistic error classification of Unmut - Unn; (right) Estimated error expectation values for each letter of the word Männer.*

error repetition for the individual error categories, and inspect if they are in accordance with the expected behavior. For the verification of the spelling performance prediction, the error expectation numbers are compared with the empirical measures from the training of the first user study.

6.7.1 Error Classification

The time-dependence of the error repetition behavior provides information on how much learning has taking place at the point in time of error entry, as illustrated in Figure 6.7. The rate of forgetting over time depends on the presence of an unknown concept of spelling in an error that can be learned. If students learn word specific spelling concepts of committed errors, e.g., the correct phoneme-grapheme matching, then the probability of forgetting the just learned spelling will grow over time. This results in a time-dependent increase of the error repetition probability (ERP: $\mathrm{P}(R_1 = f)$) at the first repetition (R_1) of the word after the erroneous input. In contrast, input device dependent errors, which are not related to word specific difficulties, are not expected to show an increase in ERP over time.

We investigate the error repetition probability for all error (sub-)categories on the inputs of the 54 children of the first user study. The analysis is performed on the 39 600 classified errors on which a repetition has been recorded. Figure 6.8 shows the time dependence of the error repetition probability at R_1. The bars indicate the increase of the ERP from repetitions with less than 1 minute between erroneous input and R_1 to repetitions with more

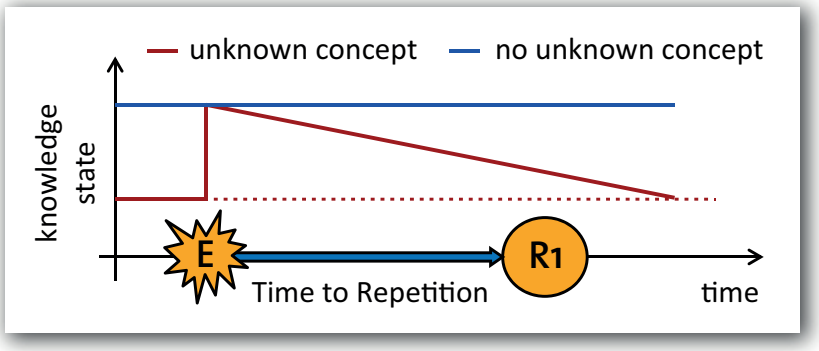

Figure 6.7: *Learning and subsequent forgetting can only be present, if an error is caused by unknown spelling concepts. The presence of a missing spelling concept can lead to a knowledge increase at error entry (learning) and subsequent decrease over time (forgetting), which becomes manifest in the time-dependence of R_1.*

than 1 minute. The significance of the increase is evaluated in a chi-squared congruency table test [Eve92].

In general, the observed ERP increase is in line with the expected behavior. The increase of dyslexic confusion, phoneme omission, and phoneme-grapheme matching is highly significant (all $p < 0.001$), which indicates that errors classified in one of these categories are based on missing spelling knowledge. An exception is the ERP increase of visual dyslexic confusions, which is not significant ($p = 0.38$) due to the low amount of observed error repetitions (184 observations only). Notably, the ERP increase for PGM errors is more than twice as large as for auditory dyslexic confusions and phoneme omissions, which have both of an auditory origin. This agrees with the assumption that lower level auditory deficits are more persistent and can not as easily trained and remediated as missing knowledge in the phoneme-grapheme correspondence.

Seventy-eight percent of Typos are *KeyDistance* related errors and show no significant increase ($p = 0.87$), which is in accordance with the common assumption about the error category. Likewise, upper to lower case confusions (*ToLowerCase*), which constitute the main part of capitalization errors, show no significant increase ($p = 0.44$). This indicates that the students actually know when to use upper case letters and the difficulty of capitalization is rather the unusual way of entering an upper case letter by using the shift key.

Interestingly, we observe a significant increase of the ERP for *Technical* ($p = 0.01$) and *ToUpperCase* ($p < 0.001$) errors of the typing and capitalization error category, respectively. The *Technical* errors are in fact related to word specific difficulties, namely the presence of umlauts. The significant increase

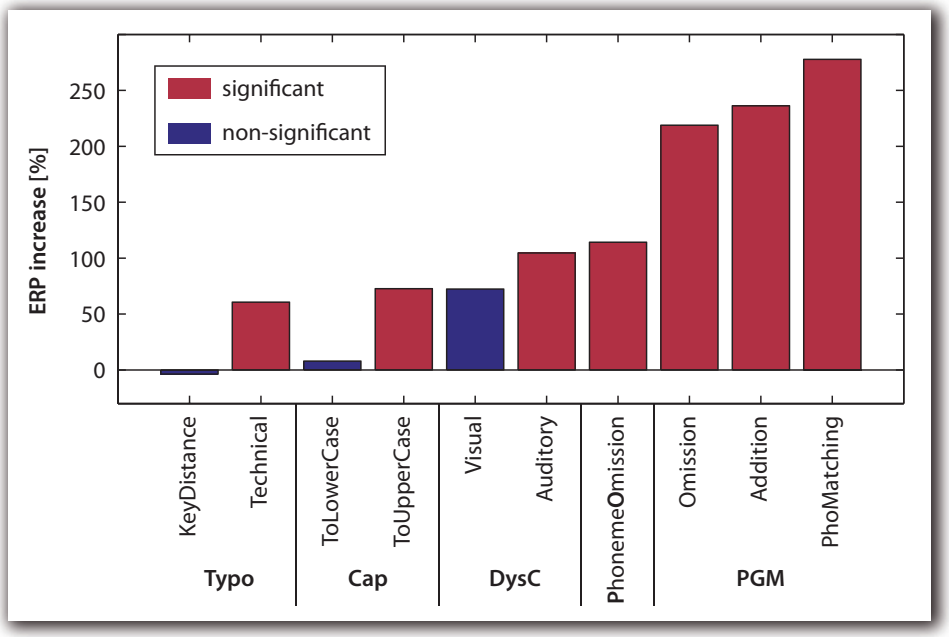

Figure 6.8: *Error repetition behavior: Relative ERP increase (in percent) from less than 60s to more than 60s between error and repetition for all (sub-)categories.*

($p = 0.01$) indicates that children have to learn the concept of entering umlauts on a keyboard and, due to their infrequent occurrence, forget this technique over time. The increase in ERP of *ToUpperCase* errors can not conclusively be explain and are assumed to be related to the misuse of the Caps Lock key. However, capitalization mal-rules are never activated in conjunction with other mal-rules, and their classification can be assumed to be correct.

6.7.2 Error Expectation

As the second part of the student model validation, we analyze the estimated error expectation by comparing it to the empirical error expectation values. As a benchmark, we consider the word difficulty measure based on the symbol confusion matrix (SCM) of the original Dybuster version [GV07], and a difficulty measure for spelling proposed by Spencer [Spe07]. The later one is based on the phonetic difference (difference between number of phonemes and letters), the phoneme transparency (probability of a phoneme being represented by a specific grapheme), and the frequency of a word, extracted from language corpora. We reconstructed Spencer's difficulty measure based on the CELEX2 word database [BPvR93]. The training of the three measures was performed on all inputs committed by the students during the first user study. Although, this leads to a slight bias to the student

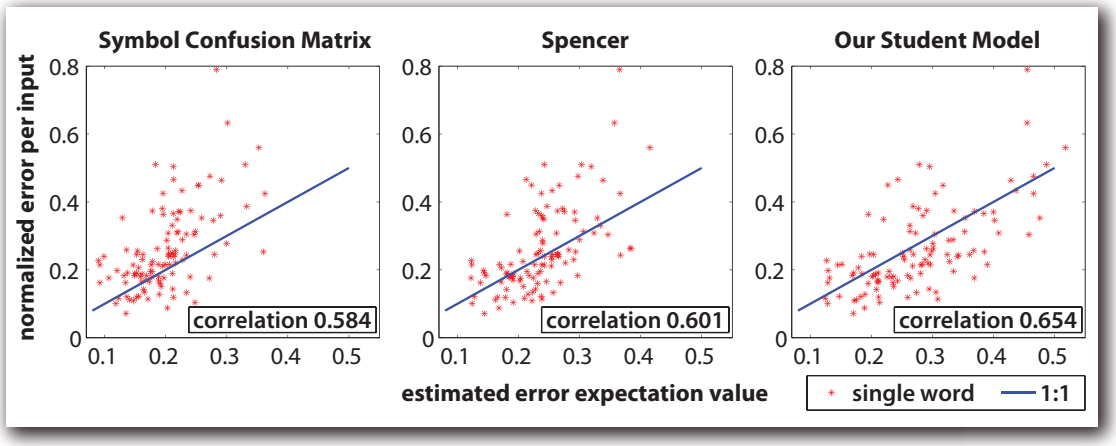

Figure 6.9: *Empirical error expectation values plotted against three word difficulty measures.*

characteristics of frequently training children, it allows for a comparison with Spencer's measure, which represents general spelling difficulties constant for all students. Subsequently, we investigated the predictive power on words entirely learned by every student, i.e. on words which have been entered twice correctly in a row.

In the first user study this yields a set of 111 words, of which the estimated error expectation values and the average error per input is displayed in Figure 6.9. The blue line represents the actual correct prediction of errors. As can be seen, all measures provide reasonable estimates for simple words (left). However, our model outperforms on very difficult words (right) and allows for an error estimation of more than 0.5 compared to a maximum of approximately 0.35 and 0.4 for the SCM and Spencer's approach respectively. The two related methods fail to represent specific difficulties of complicated words, which is indicated by the high density of data points above the correct prediction line for word with high empirical error expectation value. This is reflected in the correlation between expected word difficulty and empirical error per input. The presented student model exceeds the SCM and Spencer's measure by more than 0.05.

6.8 Error Distribution

In this section we will investigate the distribution of errors over the presented error categories. Table 6.2 shows the percentage of each category on the total amount of committed errors for the dyslexic, the control and all children, of

	1^{st} Study			2^{nd} Study			Subject		
Category	**Dys.**	**Con.**	**All**	**Dys.**	**Con.**	**All**	**1**	**2**	**3**
Typo [%]	32.5	34.7	33.3	33.0	34.0	33.4	22.8	43.6	36.9
Cap [%]	18.7	20.2	19.3	16.1	16.5	16.3	39.5	25.8	15.0
DysC [%]	9.0	9.3	9.1	9.2	9.7	9.4	9.1	7.1	6.2
PhoO [%]	7.0	9.0	7.7	9.1	10.6	9.7	5.1	8.7	7.4
PGM [%]	32.8	26.9	30.6	32.6	29.1	31.2	23.5	14.8	34.5
Total	30950	18727	49677	44111	28037	72148	749	876	840
Per input	0.338	0.279	0.313	0.345	0.276	0.315	0.445	0.294	0.334

Table 6.2: *Distribution of spelling errors (in percent) for the different groups of the first and second user study as well as the three example subjects. The last two rows display the total amount of committed errors and the average number of errors per input.*

the first and second user study. Additionally, the error distribution for the example students from Section 6.6 are given.

Although dyslexics show in general more errors per input than children of the control group, their average error distribution did not differ strongly. However, there are large within group differences between subjects of the dyslexic as well as of the control group. This indicates that the spelling training does not have to be adjusted to dyslexic or control children, but rather to the strengths and weaknesses of individual students.

In the further analyses we mainly employ the Typo and PGM error categories. Typing errors represent mechanical errors, which are for the most part not related to specific spelling difficulties of a word. In contrast, PGM errors occur due to missing knowledge in the phoneme-grapheme conversion. Therefore, this category serves as a measure of spelling knowledge for learning and forgetting comparisons. Additionally, together they account for approximately 2/3 of all committed errors.

6.9 Conclusion

This chapter addressed student knowledge representations in perturbation models on input data with multiple errors and independent mal-rules. We identified an inference algorithm based on a Poisson regression with a linear link function as most suitable to allow for an estimation of error rates based on unclassified student inputs. The appropriateness of the chosen approach is evaluated in the residual analysis of different link functions and manifests in

more reliable estimations. The model enables a classification and prediction or errors. We validated the estimated error classification and prediction on the input data of a first user study. Both classification and prediction showed the expected behavior and clearly outperformed related measures.

An interesting finding is the non-significance of visual dyslexic confusion mal-rules. Many typical visual confusions, such as *'d'-'b'*, also exhibit similarities on an auditory level (/d/-/b/). Our model represented these difficulties based on auditory mal-rules and found no additional significant influence of the visual similarity. This provides evidence for an auditory and phonological origin of dyslexia.

The investigation of the error distribution over all dyslexic as well as control children did not yield purely dyslexia specific error patterns. Although dyslexic children feature generally higher error rates, the dyslexic and control group show on average similar error distributions. However, the strong within group differences and unequal repetition behavior on errors of different categories request an extensive adaptation to individual students, independent of their indication of dyslexia. The presented model builds the foundation of the enhanced word selection controller (Chapter 9) as well as of further modeling approaches and analysis (Chapter 8 and 10).

CHAPTER 7

Affective Modeling

The student knowledge representation introduced in the previous chapter enables the long-term modeling of the actual spelling knowledge of a student. This information about the student's strengths and weaknesses in spelling builds the foundation of an effective, student-adaptive training. However, even the most effective tutoring system will fail if the student is not receptive to the material being presented. The student's current attitude toward the training, i.e., if he's interested, motivated and engaged, has been shown to be essential for the process of learning [KRP01]. Models of affect are therefore employed to represent these states in intelligent tutoring systems. These allow for an adaptation of the training to the short-term variation of the student's affect and have led to improved learning performances [AWRT09].

In this chapter we focus on the processing and selection of features relevant to affective modeling. We present a systematic approach to feature processing based on domain knowledge of affective dynamics. The presented method is implemented in a affective modeling framework, which will enable the development of a model of engagement presented in Chapter 8.

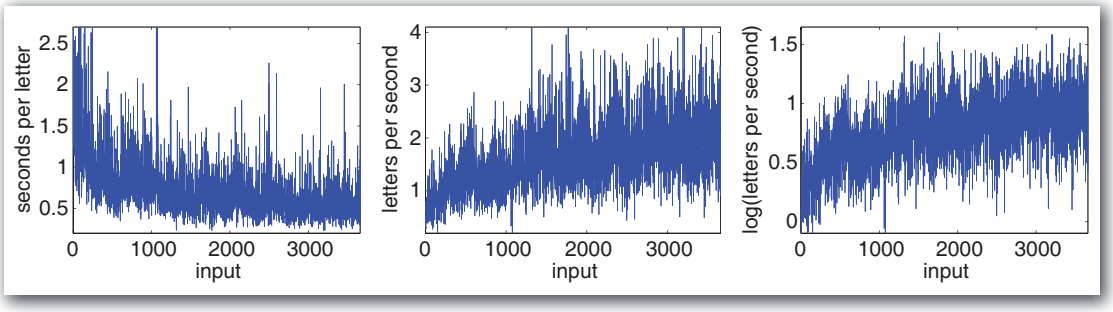

Figure 7.1: *The Input Rate of the 3600 inputs collected in the three month of training of one student extracted as: (left) seconds per letter, (center) letters per second, (right) logarithm of letters per second.*

7.1 Overview

Due to its recognized relevance in learning, affective modeling is receiving increased attention. Models of affect comprise various dimensions of emotion, motivation and engagement [dVP02]. For the representation of the affective states of a student, researchers employed models from Bayesian networks [AW05] to regression models [BCK04]. Models of affect are developed based on experimental readouts, which are assumed to be related to affect. These include observable student input data [JW06] as well as additional sensor measurements and camera data [CMA+10]. Measurements of student interaction with software tutoring systems provide the opportunity for data-driven affective modeling based on large and well-organized sample sets from a variety of experimental conditions.

However, there are two reasons why modeling affect is considered a particularly challenging task. First, ground truth is invariably approximated. Second, experimental readouts and state emissions often exhibit partial observability and significant noise levels. The latter is caused by the fact that observed student input behavior, e.g., the student input rate, is not influenced by affective states only. Various social, environmental and task dependent influences act simultaneously on the observed feature. Additionally, the input behavior is often subject to progress over time. Figure 7.1 illustrates the *Input Rate* over the 3600 inputs of one student, computed as seconds per letter, letters per second, and the logarithm of letters per second. The different specifications and scalings nicely illustrate the two main difficulties in affective modeling:

1. *Non-i.i.d. data:* Due to the progress in typing, the extracted data is

clearly not independent and identically distributed (i.i.d.). However, this property is a basic assumption for many statistical inference algorithms.

2. *Scaling:* The exact specification of the observed behavior and its scaling will strongly influences the predictive power of a feature.

Feature processing can significantly improve the predictive power of features for affective modeling and avoid the need for an extension of the affective model by progress components for each observed input behavior. However, this processing has mostly been neglected so far. Researchers used features completely unprocessed [CMA+10], or relied on expert knowledge and manual tuning for the feature processing [BCK04].

In this chapter we will present a systematic approach to feature processing for affective modeling. First, we describe the general concept of the method. In Section 7.3 we then present the affective modeling framework, which allows for an intuitive and fast processing and selection of relevant features. In Section 7.4 we give more detail on the automated identification of an optimal processing and illustrate the presented method in an example optimization of the *Input Rate* feature.

7.2 Feature Processing

In this section we describe the general concept of our feature processing method for affective modeling. It employs domain knowledge about affective dynamics and learning behavior to identify an optimal processing of continuous features. The main difficulties for affective modeling identified in the previous section are addressed as follows:

1. *Non-i.i.d. data:* Our approach to cope with the non-i.i.d. data is based upon the following central assumptions: emotional and motivational states come in spurts [JW06], and they affect the observed features on a short-to-medium time scale. The long-term progress in the observed input behavior acts on a different time scale and can be removed based on a time scale separation.

2. *Scaling of data:* To identify an optimal scaling of the data we require a measure to rate a given processing. Our approach relies on the assumption that the observed feature is simultaneously influenced by several independent factors. The central limit theorem states that the sum of many independent random variables converges to a normal distribution [Kal97]. Therefore, the processing, which results

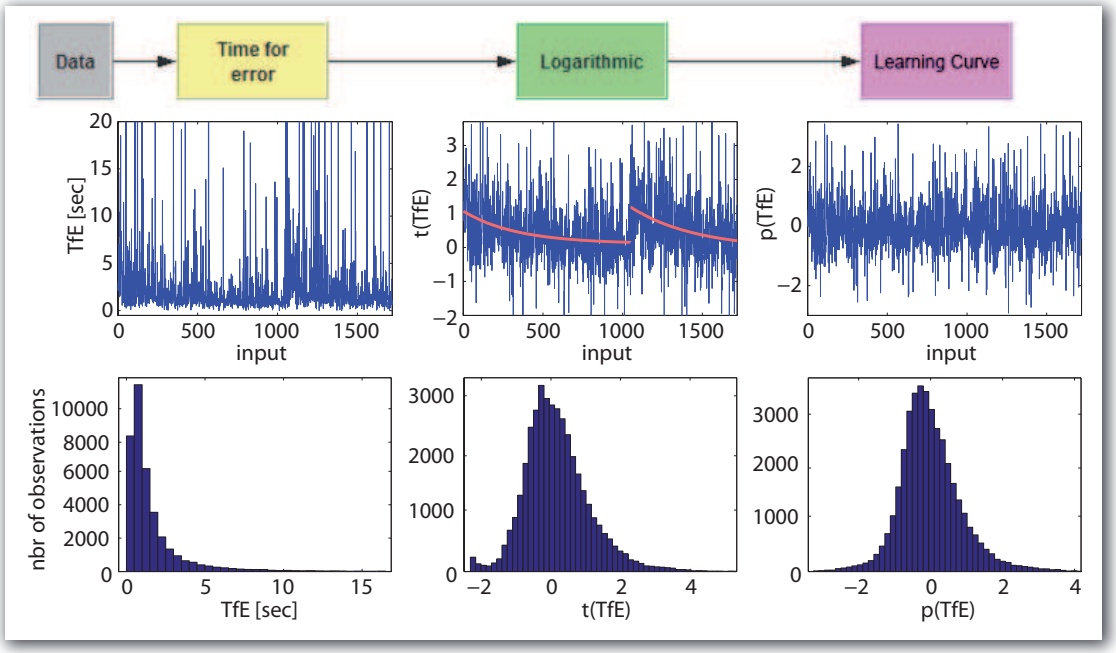

Figure 7.2: *General concept of feature processing for affective modeling exemplified by means of the processing pipeline for the TfE feature. On the* 2nd *and* 3rd *row, signal (for two learners) and histogram plots (of all children from* 1st *study) show the processing steps: extracted feature (left), after scaling (center), and after time scale separation (right).*

in a feature distribution with maximal normality, is optimal in the following sense: the different factors, including affective states, act additively and independently on the observed feature.

The workflow of our approach to feature processing is illustrated in Figure 7.2. The original feature representing the *Time for Error* (TfE) for two example students is displayed at the very left. The feature is clearly not normally distributed, as depicted in the histogram. The *Logarithmic* scaling results in an improved normality, however, the long-term progress for each student is still evident. After the time scale separation by means of a *Learning Curve* regression, the feature approaches the desired i.i.d. and normality properties. It's important to note, that the time scale separation is not performed until the feature is scaled such that the different influences act additively and independently.

7.3 Affective Modeling Framework

In this section we describe the affective modeling framework, which provides a set of tools required for the feature processing. It enables a intuitive and modular design of processing pipelines for individual features. This allows for dealing with large sets of features and the possibility of incorporating and investigating new features with very little effort.

Extracted features are considered as signals, which are stored in data packages and passed through the processing pipelines. A pipeline consists of one or several modules, which can be manually connected in the framework. The input of each module is a data package $\mathbf{x}$, which becomes subject to the module specific processing $p_{\mathbf{y}}(\mathbf{x})$ with module parameters $\mathbf{y}$. This new data package then forms the output of the module.

Individual data points x_i store a feature value and a reliability flag, indicating if the stored feature value should be included for further processing. A data point can be set to non-reliable, due to missing information in the feature extraction part or due to invalid processing requests, such as the logarithmic scaling of negative values.

In the following, we introduce the individual processing modules of the affective model framework, which are employed by the processing optimization method described in Section 7.4. We present the module specific parameters and give the range of values commonly observed in the parameter optimization for the model of engagement described in Chapter 8. The abbreviations of the processing modules are depicted in bold letters.

7.3.1 Features

At the beginning of every pipeline stands a feature module. It extracts information from the provided student data, such as the input rate of a student. Features are application-dependent and need to be implemented according to requested information and available student data. The features employed for the engagement model are described in detail in Chapter 8. These are the only modules, which have to be adjusted for affective modeling based on the data of a different learning software.

Each feature has to be manually labeled as dependent or independent variable for the later model building process. Features are evaluated for every input and marked with the corresponding student ID. All data points x_i of one feature form a signal, which is stored in a data package and sent into the pipeline.

7.3.2 Scaling

The affective modeling framework provides a set of scaling modules. The scaling operation $p_{\mathbf{y}}(\mathbf{x})$ is evaluated point-wise. The processing $p_{\mathbf{y}}(\mathbf{x})$ and the corresponding parameters $\mathbf{y}$ of the different scaling modules are described in the following:

- ***Log**arithmic:* The logarithmic scaling

  $$p_s(x_i) = \log(s + x_i)$$

  depends on the parameter s. If $s + x_i \leq 0$, the data point x_i becomes non-reliable. The parameter s is strongly dependent on the scale of the processed feature.

- ***Exp**onential:* The exponential scaling

  $$p_b(x_i) = \exp(-\frac{x_i}{b})$$

  is specified by the parameter b. If $b > 0$, the orientation of the features is changed, i.e., large feature values are scaled toward zero and low feature values result in high values. The parameter b is strongly dependent on the scale of the processed feature.

- ***Split**ting:* The processing of the splitting module is represented by the indicator function

  $$p_s(x_i) = \mathrm{I}_{x_i > s}$$

  with parameter s for the splitting level. The splitting module is commonly used to separate between zero and one or more observations ($s = 0$), e.g., to indicate if a help call occurred or not.

7.3.3 Time Scale Separation

Time scale separation enables a distinction between sustainable progress in the observed feature ($f(i)$) and other local effects ($p(x_i)$), such as the influence of affective states. The separation of long-term variation $f(i)$ depends on the temporal input position i in the student's input history. In the framework, different modules for the modeling of the long-term progress are provided:

- ***Lin**ear **Reg**ression:* For a fast approximation to long-term variations, a linear regression is provided:

  $$f(i) = ai + b$$

  This approximation is valuable for exploratory optimizations with many cycles.

- ***Learn**ing **C**urve:* Learning curves (see Section 10.2) are employed to describe practice effects of serial trials. The progress is modeled by means of an exponential function:

  $$f(i) = a\exp(-bi) + c$$

  This model of the long-term progress is employed to describe the progress in the observed input behavior of the model of engagement (see Chapter 8).

- ***Savitzky-Golay:*** If the long-term process is not assumed to be a learning progress, but just an arbitrary, slow variation over time, a non-parametric model can be used. The implemented Savitzky-Golay filter [SG64] performs a local polynomial regression f on a series of values, which is evaluated at the position i.

The actual processing of the time scale separation modules consists of an estimation of the long-term variation $f(i)$ for every student and a subsequent removal from the signal:

$$p(x_i) = x_i - f(i)$$

7.3.4 Outlier Handling

The affective modeling framework provides modules for the handling of outlier and non-reliable values. Data points detected as outlier can either be marked as non-reliable or be constrained by given bounds. Data points marked as non-reliable are generally not considered by processing modules. However, specific modules allow for an estimation of a feature value. This avoids loss of data, if subsequent machine learning techniques are not able to deal with missing data and becomes prominent for features, such as time after help request, which only sparsely yield reliable values.

- ***Deviation Cut:*** The deviation cut module assumes a normally distributed signal $\mathbf{x}$ and constrains the data points inside the intervals of $\pm a$ times the standard deviation around the mean:

  $$p_a(x_i) = \min(\mu + a\sigma, \max(\mu - a\sigma, x_i))$$

  with $\mu = \text{mean}(\mathbf{x})$ and $\sigma = \text{std}(\mathbf{x})$. The optimal constraint of the data range is commonly found to be three to four times the standard deviation.

- ***Non-reliable Remover:*** The non-reliable remover allows to replace non-reliable values in a predefined way, if subsequent processing steps are not able to deal with missing data. The options for the computation of replacement values include mean and median of the data points **x** over one or all subjects, and all the regression models introduced in the time scale separation module.

7.3.5 Filtering

The filter modules remove unwanted components from the signal **x**. To finally enable the affective models to run in real-time, all filters are implemented with the option to use only present data, i.e., $p(x_i)$ depends only on x_k for $k \leq i$. The following filters are provided in framework:

- ***Low-Pass:*** The low-pass module performs a Gaussian low-pass filter

$$p_n(x_i) = \sum_{j=0}^{n} x_{i-j} G(j,n),$$

  where $G(j,n)$ corresponds to the sampled Gaussian kernel

$$G(j,n) = \frac{1}{\sqrt{2\pi n}} e^{-\frac{j^2}{2n}}$$

  and n is the parameter for the filter size.

- ***Variance:*** The variance filter returns the variance over the last n inputs:

$$p_n(x_i) = \mathrm{var}([x_{i-n}, ..., x_i])$$

7.3.6 Utility

The utility modules of the framework provide help in constructing the processing pipelines. These range from visualization of the data, to machine learning methods, which indicate the appropriateness of selected parameters.

- *Visualization:* The visualization module displays all incoming data packages. It enables an illustration of the actual signal **x** as well as its histogram (see Figure 7.2, center and bottom row). Linked into an optimization cycle, the module allows for a continuous visualization of the transformation of signal and histogram.

- *LASSO Regression:* The LASSO regression module enables an investigation of the relation between independent (R) and dependent variables ($\mathbf{x}$) contained in the incoming data packages. The LASSO method bounds the L^1-norm of the parameter vector $\mathbf{b}$ by a parameter t to inhibit overfitting [Bis06]. For binary dependent variables ($R \in \{-1,1\}$), e.g., in the model of engagement (see Chapter 8), a logistic regression model is used:

$$\hat{\mathbf{b}} = \underset{\mathbf{b}}{\operatorname{argmax}} \prod_{i=1}^{N} \frac{1}{1+\exp(-\mathbf{b}^T \mathbf{x}_i R_i)} \quad \text{subject to} \sum_{j>0} |b_j| \leq t.$$

  Based on a 10-fold cross-validation, an optimal bounding parameter t and the corresponding features can be selected. The LASSO regression module allows not only to compare different features but also various parameter settings of the processing modules. However, the memory requirement of an extensive comparison of many features and several parameter settings can exceed the available resources.

- *Regression Visualization:* The regression visualization module performs a 10-fold cross-validation based on a regression model according to the present dependent variable and visualizes the result. The user has to select which features from the incoming data packages should be used for the regression. For binary dependent variables a logistic regression model is used, as illustrated in Figure 8.2.

- *Parameter Optimizer:* The parameter optimizer module can be used for continuous features to find appropriate parameter settings of the previously presented modules. The parameter optimizer module has to be linked at the beginning of the pipeline section, which has to be optimized. A *Buffer* module is placed at the end of the section, as illustrated in Figure 7.3. The parameter optimizer searches for parameters of the modules to minimize a given cost function, as described in more detail in the next section. Additionally, the module contains the option to visualize the topology of the cost function (see Figure 7.5), to offer an assessment of the attained solution.

7.4 Normality Maximizing Processing

The affective modeling framework presented in the previous section allows for a manual composition of processing pipelines. The LASSO regression facilitates the selection of appropriate module combinations and parameter settings. However, for large user data volume and extensive feature sets

the simultaneous evaluation of different processing is very limited in terms of memory consumption. If these memory constraints are reached and a simultaneous evaluation of all processing and features by a LASSO regression is not feasible anymore, we propose the normality maximizing preprocessing of the features. The feature selection can then be performed on the reduced set of processed features only.

The domain knowledge on affective dynamics and learning behavior described in Section 7.2 enables the identification of an optimal processing $p_{\mathbf{y}}(\mathbf{x})$ on the data of an individual feature $\mathbf{x}$ alone. The idea is to search in the set of processing modules and in the corresponding parameter space for a processing, which results in a distribution of the feature with maximal normality. However, it's important to bear in mind that the central limit theorem underlying the optimality criterion is based on an i.i.d. assumption on data. Therefore, the optimality of a processing can only be assessed after the time scale separation.

The *Parameter Optimizer* and *Buffer* module pair are employed to select a section of the processing pipeline of continuous features $\mathbf{x}$ for optimization. All combinations of modules from the set of modules $\mathbf{M}$ contained in the selected subpart of the pipeline are then tested for optimality. In the following we describe the employed cost function and optimization method, and its adaptation to our specific setting.

7.4.1 Cost Function

The cost function of the optimization has to feature the normal distribution as the limiting distribution. Two options are implemented in the parameter optimizer module. The Jarque-Bera normality test [JB80] and a differential entropy measure $h(X)$. The differential entropy of the discrete signal $h(p_{\mathbf{y}}(\mathbf{x}))$ is approximated by a histogram approach [GvdM87] and has the normal distribution as the limiting distribution of standardized $p_{\mathbf{y}}(\mathbf{x})$ (with $\mu = 0$ and $\sigma = 1$) [Bis06].

The Jarque-Bera test yielded many numerical instabilities for very non-normally distributed $p(\mathbf{x})$. The differential entropy approach performed clearly superior and hence was used in the further analyses.

7.4.2 Nelder-Mead Optimization

The optimization method was chosen with respect to the specific characteristics of the cost function and the optimization problem. The optimization of

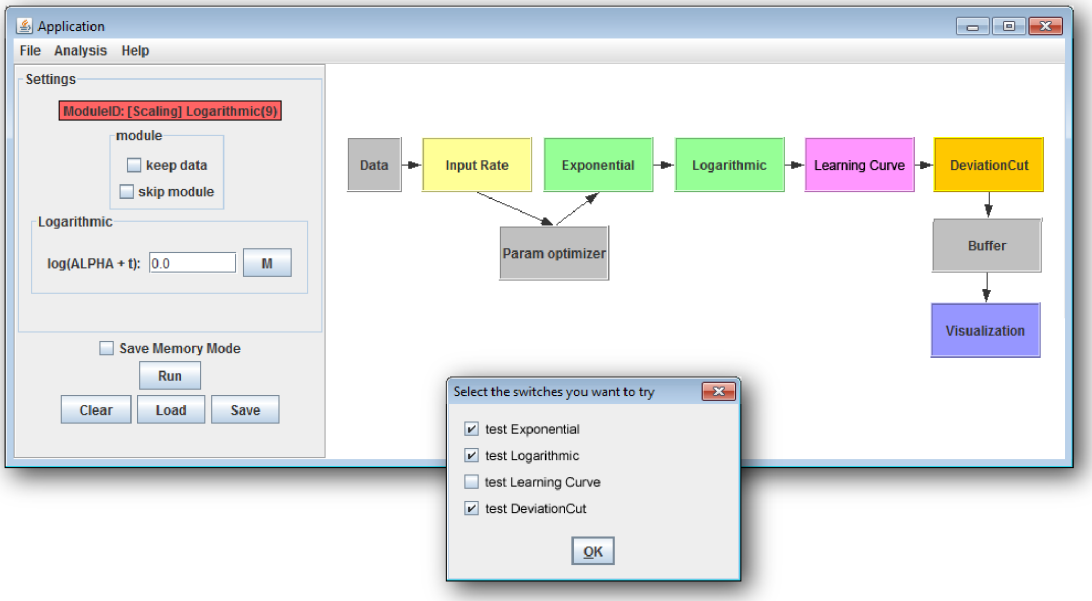

Figure 7.3: *Affective modeling framework: The Parameter Optimizer and Buffer module enclose the set of modules, which should be searched for an optimal processing. In the dialog we select the two scalings and the outlier detection as optional.*

the method has to:

- optimize multiple parameters $\mathbf{y}$,
- work with a nonlinear cost function $f(\mathbf{y})$,
- run without any derivatives with respect to the parameters.

The Nelder-Mead method [NM65] fulfills these requirements. It is a heuristic minimization technique for smooth functions, based on the concept of a $N+1$ simplex for the optimization of N parameters. Each vertex i of the simplex represents a parameter setting $\mathbf{y}_i$. The iterative method identifies the worst parameter setting and generates a new test position by extrapolating the behavior of the cost function measured at the vertices of the simplex. The algorithm consists of a reflection, expansion, contraction, and reduction step.

The modification of the parameters $\mathbf{y}$ can lead to an increase of the number of non-reliable values $n_r(p_{\mathbf{y}}(\mathbf{x}))$ in the data. For example, the decrease of the parameter s of the logarithmic scaling $\log(s + x_i)$ will render all data points $x_i \leq s$ non-reliable. To penalize non-reliable data points, the minimization method has to run on the ordered pair $f(\mathbf{y}) = (n_r(p_{\mathbf{y}}(\mathbf{x})), -h(p_{\mathbf{y}}(\mathbf{x})))$. To take account of the non-continuity of the objective function, the Nelder-Mead

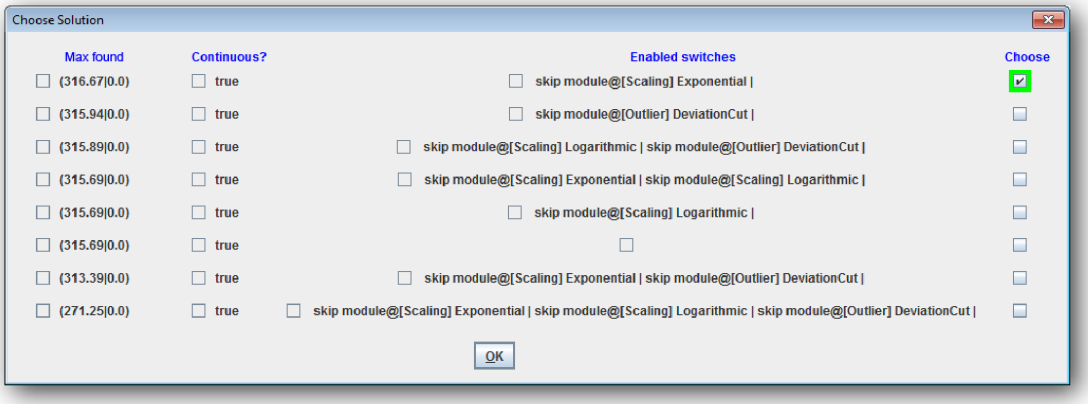

Figure 7.4: *Selection of the feature processing. The optimal processing was found for the Logarithmic - Learning Curve - Deviation Cut pipeline.*

method has to be adapted. The original convergence criterion relies on the variance of the function values

$$\sum_{i=1}^{N+1} (f(\mathbf{y}_i) - \bar{f})^2 < \epsilon_v$$

only. To enable a convergence at non-continuities in the objective function, we introduced an additional convergence criterion based on the distance of the best $\mathbf{y}_1$ and the worst parameter settings $\mathbf{y}_{N+1}$:

$$|\mathbf{y}_1 - \mathbf{y}_{N+1}| < \epsilon_d$$

7.4.3 Optimization of Input Rate

In this section we will illustrate the identification of the optimal processing of the *Input Rate* feature (letters per second). Figure 7.3 shows the affective modeling framework with a processing pipeline for the *Input Rate* feature. The subsection of the pipeline between *Parameter Optimizer* and *Buffer* contains the four modules *Exponential* and *Logarithmic* scaling, *Learning Curve*, and *Deviation Cut*. The *Parameter Optimizer* evaluates all possible combinations of the modules selected by the user in a message dialog. In the *Input Rate* example, we select all modules as optional except the *Learning Curve*, to ensure the time scale separation. As listed in Figure 7.4, the *Logarithmic* scaling and *Deviation Cut* combination achieved the distribution with maximal normality from the eight evaluated pipelines.

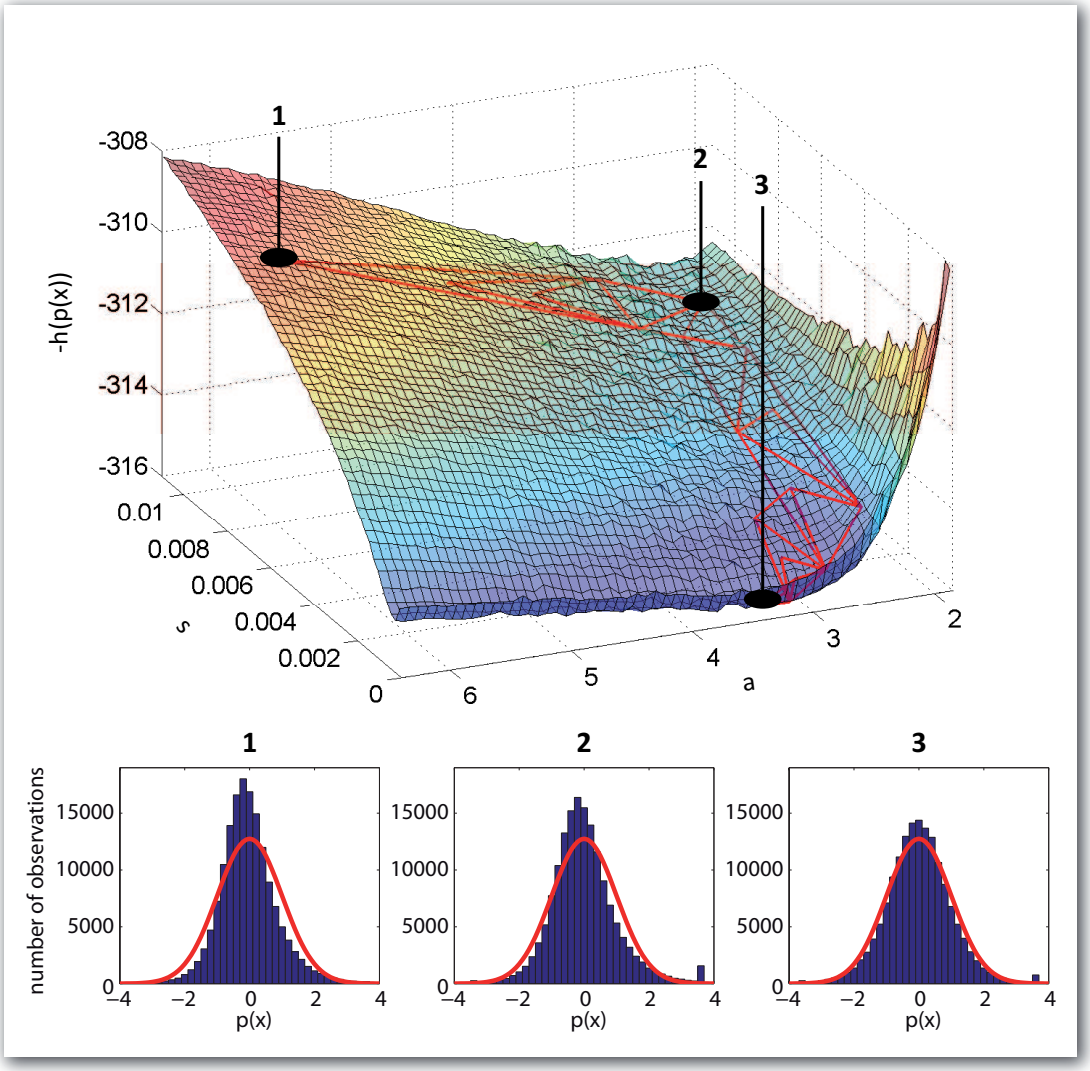

Figure 7.5: *Visualization of the optimization process of the Input Rate feature. Top: Illustration of the cost function in parameter space for the logarithmic scaling (s) - learning curve - deviation cut (a) pipeline; the red triangle sequence depicts the optimization path of the Nelder-Mead method. Bottom: Histogram of three sample distributions of processing based on parameter settings extracted from the indicated positions in the parameter space; the convergence toward the normal distribution can be observed.*

Figure 7.5 (top) illustrates the cost function $h(p_{\mathbf{y}}(\mathbf{x}))$ in the parameter space of the optimal pipeline *Logarithmic - Learning Curve - Deviation Cut*. The triangles depict the optimization trajectory of the Nelder-Mead method. The signal and the histogram of three selected vertexes of the simplex can be investigated in Figure 7.5 (bottom).

7.5 Conclusion

In this chapter we introduced a novel approach to feature processing for affective modeling. We showed that domain knowledge on affective dynamics and learning behavior can be systematically incorporated into data preprocessing. It addresses the appropriateness of scalings and removes the progress component from the observed input behavior. The proposed algorithm is implemented in a modular framework for affective modeling, which provides a rich toolbox for feature processing and allows for an intuitive design of processing pipelines.

In the next section we describe the employment of the presented method to develop a model of engagement based on the collected user data. This application will demonstrate the advantages of feature processing for affective modeling.

CHAPTER 8

Model of Engagement

This chapter entertains the idea that intelligent tutoring systems can adapt the training to individual students based on data-driven identification of engagement states from student inputs. We develop a model of engagement dynamics in spelling learning by quantitatively relating input behavior to learning. The method for feature processing in affective modeling presented in the previous chapter is employed to increase the predictive power of the observed input behavior. The model structure is the dynamic Bayesian network inferred from the input data collected in the first user study. *Focused* and *Receptive* states are identified on the basis of input and error behaviors alone.

8.1 Overview

In recent years, researchers developed various models of affective determinants, i.e., of motives, attitudes, moods, and emotions. For example, de Vicente and Pain describe a 9-dimensional representation of the student's motivation, comprising variables such as confidence, effort, and satisfaction [dVP02]. The development of fine grained models of affect requires precisely labeled data based on student self reports or expert evaluation of additional sensory and camera data. Due to the lack of a reliable assessment

of the student's affective states in the Dybuster user studies, we inferred the model of engagement on the basis of input and error behaviors. However, this inhibits a detailed modeling of the underlying affective dimensions. The developed model represents rather higher level factors, namely *Focused* and *Receptive* states, as well as *Forgetting*.

The employment of learning as an indication of engagement is described in more detail in Section 8.2. In Section 8.3, we specify the set of features, which are assumed to be related to engagement. The features are implemented in the affective modeling framework and underwent the processing presented in the previous chapter. Based on the LASSO logistic regression, the features for the model of engagement are selected. The regression analysis in Section 8.4 clearly demonstrates the advantages of feature processing for affective modeling.

Based on the processed features selected by the LASSO method we developed a causal model of engagement dynamics and forgetting. Section 8.5 describes how the *Focused* and *Receptive* states are identified by quantitatively relating input behavior and learning. These states needs to be treated as a dynamic variable, since empirical evidence suggests that a student's motivation level tends to go in spurts [JW06]. The structure of the developed model is a dynamic Bayesian network and the mutual dependence of the engagement states is inferred from the input data. In Section 8.6 we describe and investigate the final model of engagement and evaluated the stability of the model.

8.2 Indication of Engagement

Engagement states are inferred from the repetition behavior of committed errors and without external direct assessments. We subscribe to the validated hypothesis of interplay between human learning and affective dynamics [KRP01]. Therefore, we investigate the presence of learning to draw conclusions about the current states of engagement.

As in Section 6.7, committed errors and the knowledge state at subsequent spelling requests of the same word are jointly analyzed. Error repetition R_1 acts as a noisy indicator for learning (see Figure 8.1). R_1 is influenced, on the one hand, by the initial knowledge state indicating the severity of an error, i.e., if an error has been committed due to missing spelling knowledge or due to lack of concentration. On the other hand, error repetition is affected by the increase of knowledge (learning) at the point in time of error entry. Both effects are strongly related to engagement.

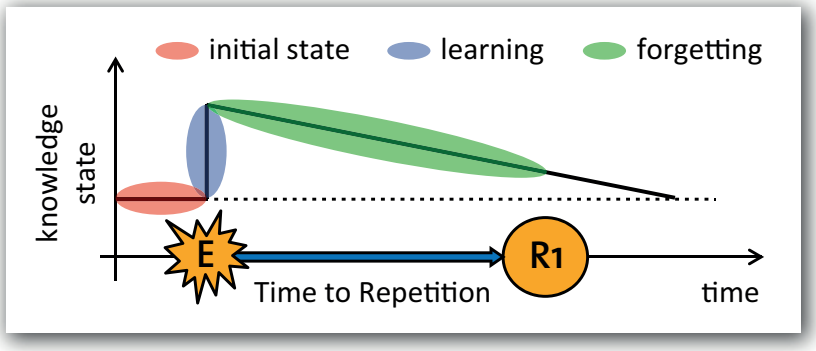

Figure 8.1: *Indicator of engagement: The knowledge state after an error is determined by the initial knowledge (severity of error) and the amount of learning at error entry, which are influenced by engagement states. Subsequently, we expect a decay in spelling knowledge by forgetting.*

Since the assessment of the knowledge state after the error, i.e., the subsequent spelling request of the same word, is delayed, R_1 is also influenced by forgetting of the learned spelling knowledge. Therefore, our model of engagement has to comprise the process of forgetting. We restrict the analysis on phoneme-grapheme matching (PGM) errors, which represent missing knowledge in spelling, in contrast to, e.g., typing errors. We extracted 14892 observations of PGM errors with recorded word repetitions from the log files of the first study.

8.3 Extracted Features

In collaboration with the Institute of Neuropsychology of the University of Zürich, we identified a set of features which are assumed to be related to engagement or the process of forgetting. The recorded features are consistent with previous work [BCK04, JW06, AW05, OL08]. The set contains measures of input and error behavior, timing, and variations of the learning setting induced by the word selection controller. Each feature is evaluated for every input i and returns the initial value for the data point x_i. We extracted 159704 data points for every feature from the log files of the first study, which resulted in a data package size of approximately 4 MB.

In the following we describe the extracted features in detail. Their abbreviations are indicated in bold letters. The very first is the indication of dyslexia:

- ***Dys**lexia:* This binary feature stays constant for all inputs of an individual student. It represents the diagnosis of dyslexia.

8.3.1 Timing

Measures of timing have shown to be valuable predictors for engagement [Bec05]. Features extracting specific error behavior are averaged per input, if multiple errors occur in one input and result in non-reliable data points, if no error occurred at all.

- ***I**nput **R**ate:* The input rate is computed based on the time t to complete the input and the number of keystrokes k of the input.

$$value = \frac{k}{t}$$

- ***I**nput **R**ate **V**ariance:* This features represents the variance of the input rate over all k keystrokes l_j of an input. $\bar{t}$ is the average time between two keystrokes $t(l_j, l_{j+1})$.

$$value = \frac{1}{k-1} \sum_{j=1}^{k-1} \left(t(l_j, l_{j+1}) - \bar{t} \right)^2$$

- ***T**hink **T**ime:* This feature extracts the time between the program has finished playing the audio sample of the prompted word (t_A) and the user starts typing (t_U).

$$value = t_U - t_A$$

- ***T**ime for **E**rror:* The time for error feature extracts the time between the last correct keystroke (t_L) and the entry of the error letter (t_E).

$$value = t_E - t_L$$

- ***T**ime **t**o **N**otice **E**rror:* This features measures the time between the entry of an error letter (t_E) and the first corrective action (t_{FC}).

$$value = t_{FC} - t_E$$

- ***T**ime for **C**orrection:* The time for correction feature extracts the time between the first corrective action after an error (t_{FC}) and the entry of the next letter (t_N).

$$value = t_N - t_{FC}$$

- ***T**ime **a**fter **C**orrection:* Similar to time for correction, this feature extracts the time between the last corrective action (t_{LC}) and the entry of the next letter (t_N). This represents the time a student is thinking about the spelling after an error until he continues typing. It equals the time for correction feature, if only one corrective action is performed.

$$value = t_N - t_{LC}$$

- ***O**ff **T**ime:* The off time feature measures the greatest time span t between two consecutive keystrokes l_j and l_{j+1}.

$$value = \max_j(t(l_j, l_{j+1}))$$

8.3.2 Input & Error Behavior

Measures of the input and error behavior have commonly been used for affective models [BCK04, AW05]. The Dybuster-specific set of input and error features consists of:

- ***H**elp **C**alls:* The help call feature counts the number of help requests by the student. These include repetition of the dictation, replay of the computed auditory code and short display of the correct spelling of the prompted word.
- ***F**inished **C**orrectly:* This feature indicates, if all errors have been corrected and the word is finally spelled correctly.
- ***S**ame **P**osition **E**rror:* This feature counts how many times two or more errors have been committed at the same letter position in a word.
- ***S**illy **I**nput:* The silly input feature is a heuristic measure for the seriousness of an input, which developed during the log file analysis. If the input
  - is 4 times longer than the prompted word,
  - contains 4 times the appearance of a letter not being part of the prompted word, or
  - contains a letter appearing more often than the length of the prompted word (this criterion applies only to words, which have more than 4 characters),

  the input is classified as silly.

- ***R**epetition **E**rror:* This repetition error feature parses the input history of the prompted word for a given sequence of correct and erroneous inputs, and returns true if the sequence was found and false otherwise. For our analysis we only consider a correct and a erroneous previous input, resulting in the features **REc** and **REe**.
- ***E**rror **F**requency:* The spelling knowledge representation, presented in Chapter 6, provides an estimate of the error expectation value for every word. The occurrence of errors is assumed to be Poisson distributed. Based on the observed (λ_O) and expected number of errors (λ_E) over the last ten inputs, the Kullback-Leibler divergence [KL51] (D_{KL}) between observed (P_O) and expected error distribution (P_E) is employed as a measure for the relative error frequency.

$$D_{\mathrm{KL}}(P_O||P_E) = \sum_i P_O(i) \log \frac{P_O(i)}{P_E(i)}$$

  This distance between observed and expected distribution is taken positive if more errors than expected occurred and taken negative if less errors occurred.

8.3.3 Controller Induced

The features described in this section are controlled by the training software. They are not related to affect. The measures enable the modeling of forgetting over time or by interference [OL08].

- ***T**ime **t**o **R**epetition:* This feature measures the time between the erroneous input and the next repetition of the respective word.
- ***L**etters **t**o **R**epetition:* The letters to repetition feature return the number of entered letters between the erroneous input the next repetition of the same word.

8.4 Feature Selection

The features described in the previous section are implemented in the affective modeling framework. The relation between a feature with given processing $p(\mathbf{x})$ and error repetition R_1 is estimated via the LASSO regression module. However, the comparison of all reasonable possibilities of processing module combinations (~ 20) and different parameter settings ($\sim 5^3$; 5 settings for 3 parameters) for every feature (18) would exceed the available hardware

resources. Already this rather sparse sampling of the parameter space would lead to 50 000 data packages, which request approximately 200 GB of memory. Therefore, the optimal processing $p(\mathbf{x})$ of all continuous features is identified by the parameter optimization described in Chapter 7.

An exception are the controller induced features, which are not related to affect and hence cannot assumed to be normally distributed. The optimal processing of the controller induced features and of the discrete feature *Help Calls* are selected using the LASSO logistic regression. As a matter of course, binary features are not in the need of any processing.

The LASSO method allows for an evaluation of different filter and filter settings for all features. The bounding parameter t is continuously increased

Feature	**Processing Pipeline**	**b**	**sig.**
Dys		x	
Timing			
IR	Log - LearnC - DevC - Var	-0.12	4e-6
IRV	Log - LearnC - DevC	-0.22	2e-11
TT	Log - LearnC - DevC	x	
TfE	Log - LearnC - DevC - LowP	-0.50	1e-9
TtNE	Exp - LearnC - DevC	-0.18	1e-5
TfC	Log - LearnC - DevC - LowP	x	
TaC	Log - LearnC - DevC - Var	x	
OT	Log - LearnC - DevC - LowP	0.27	1e-9
Input & Error Behavior			
HC	Split(zero/non-zero)	0.29	2e-4
FC		-0.49	1e-7
SPE		x	
Silly		x	
REc		-0.28	8e-8
REe	LowP	0.20	1e-9
EF	Exp	0.06	2e-4
Controller Induced			
TtR	Exp	-0.29	2e-8
LtR	Log	0.34	1e-9

Table 8.1: *Optimal processing pipeline, estimated parameter b and significance for features selected by the LASSO logistic regression. Note that the Exponential scaling inverts the orientation of a feature. Feature not selected by the LASSO method are marked with a 'x'.*

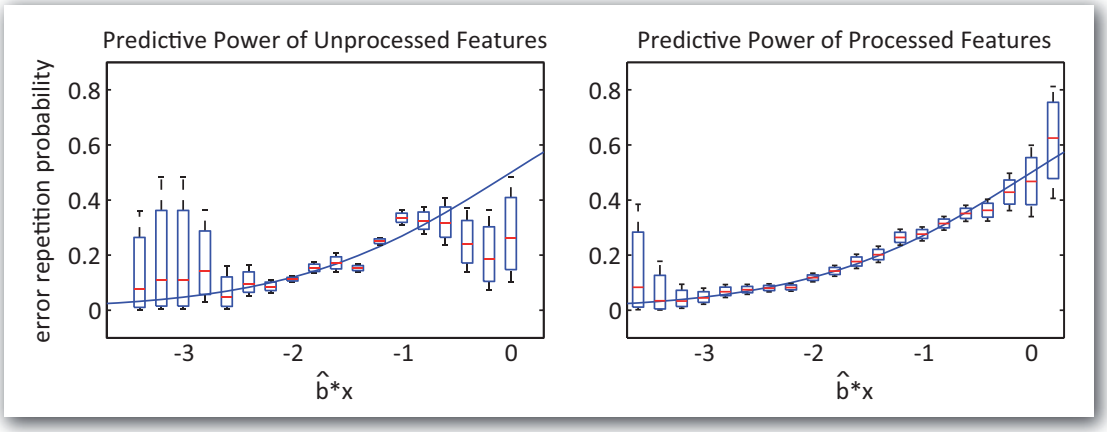

Figure 8.2: *ERP prediction (10-fold cross-validation) from unprocessed (left) and processed features (right). Predictions are plotted as blue curve and accompanied by mean (red stroke), 68% (box), and 95% confidence intervals (whisker) of the observed repetitions for bins containing at least 10 observations.*

until the processed features $p(\mathbf{x})$, which enter the regression model, is not increasing the BIC score any further. The selected features with corresponding processing pipeline are listed in Table 8.1. The regression parameters are denoted by b_i.

The benefit of feature processing is demonstrated in an evaluation of the predictive power of the feature set. Figure 8.2 illustrates the comparison between error repetition probability (ERP) predictions obtained from unprocessed and processed features. The predictions displayed in the charts are computed based on a 10-fold cross-validation. The observed repetition behavior is collected in bins and is displayed for bins containing at least 10 observations. Notably, the model with unprocessed features leads to extreme prediction values (from $\hat{b}x = -16.6$ and ERP = 5e-9 to $\hat{b}x = 4.4$ and ERP = 98.8) not illustrated in the chart. This effect is owing to outliers and inappropriate scaling of the unprocessed features.

Beside the evident superiority of the processed features, the benefit of the feature processing is additionally reflected in the estimated model evidence. The model based on processed features exhibits a better BIC score (-6369) compared to unprocessed regression (-6742).

8.5 Model Building

The LASSO logistic regression allowed for a selection of features, which are related to our indicator of engagement. The aim of this section is to describe the development of a causal model of engagement based on these features. The design of the model relies on the assumption that motivational and emotional states of students come in spurts and should be modeled by dynamic variables [JW06]. We introduce a graphical model which relates input behavior to learning, and explains the dynamics of engagement states in spelling training. The structure of the dynamic Bayesian network is inferred from the user input data, as described in the following.

The investigation of the selected features listed in Table 8.1 and the consideration of the desired input behavior of children enables the identification of three factors influencing the knowledge state at the next repetition:

- ***F**ocused **S**tate:* indicates focused or distracted state of the student. In non-focused state more non-serious errors due to lapse of concentration occur, which corresponds to a higher initial knowledge state in Figure 8.1. If errors are committed only because the student was distracted and not because the actual spelling was unknown, the error is less likely to be committed again at the next repetition. This results in a lower ERP.
- ***R**eceptive **S**tate:* indicates the receptiveness of the student (receptive state or beyond attention span). In a non-receptive state, children are not able to learn the correct spelling of committed errors, which becomes manifest in a smaller knowledge increase in Figure 8.1 and causes a higher ERP.
- *Forgetting:* The time (decay) and number of inputs (interference) between error and repetition induce forgetting of learned spelling knowledge and increase the ERP. This corresponds to a later repetition or a steeper decrease in Figure 8.1.

The parameters of the logistic regression indicate how features are related to the ERP. We infer the affiliation of features to engagement states based on the relations extracted from the regression analysis and expert knowledge about desired input behavior, as illustrated in the following examples: (1) The parameter $b = 0.06$ of EF demonstrates that a higher than expected *Error Frequency* is related to a lower ERP. This indicates that a student is *non-focused* and commits more but rather non-serious errors. On contrary, (2) if a student does not finish an input correctly (FC $= 0$), the ERP increases ($b = -0.49$).

This indicates that students, which are not correcting their spelling errors, are less likely to pick up the correct spelling, and the *Finished Correctly* feature should be assigned to the *Receptive* node. Notably, (3) the number of help calls, commonly considered as an indicator of strong engagement in active cognitive processes that are thought to promote deeper understanding and long-term retention [Ano07], are related to a higher ERP ($b = 0.29$). The effect can be explained by the nature of the Dybuster help options. The most frequent help call is the repetition of the dictation, indicating that the prompted word has already left the student's memory. Therefore, the *Help Calls* feature is associated with receptiveness.

The edges between engagement states and observed nodes are directed to the extracted features, since the observed input behavior is assumed to be influenced by the engagement states. However, the *Time to Repetition* and *Letter to Repetition* features are not observations of the student behavior. These features are controlled by the spelling software and induce forgetting; hence, they point toward the *Forgetting* node.

In the following we investigate the mutual dependence of the two engagement states, which are considered as dynamic nodes. We compare three models: (1) based on a mutual independence assumption (FS $\nleftrightarrow$ RS); (2) with dependence of *Focused* on *Receptive* state (FS $\leftarrow$ RS); (3) with dependence of *Receptive* on *Focused* state (FS $\rightarrow$ RS). The mutual dependence of the engagement states is inferred based on the estimated model evidence (BIC).

The connections between engagement states and *Forgetting* are omitted from the analysis. Although these two components of the model both influence the knowledge state at the next repetition, they describe different periods of the modeled process and are assumed to be mutually independent. Whereas the engagement states represent the processes at the point in time of error entry, the *Forgetting* describes the loss of learned spelling knowledge in the time between error and repetition.

The conditional probability distributions of observed input behavior and latent variables are modeled either by tabular nodes (for binary and categorical features with discrete parents), by Gaussian nodes (for continuous features), or by Softmax nodes (logistic function for binary features with continuous parents). The parameters of the distributions of the DBN are estimated by means of the expectation maximization (EM) algorithm [Mur01].

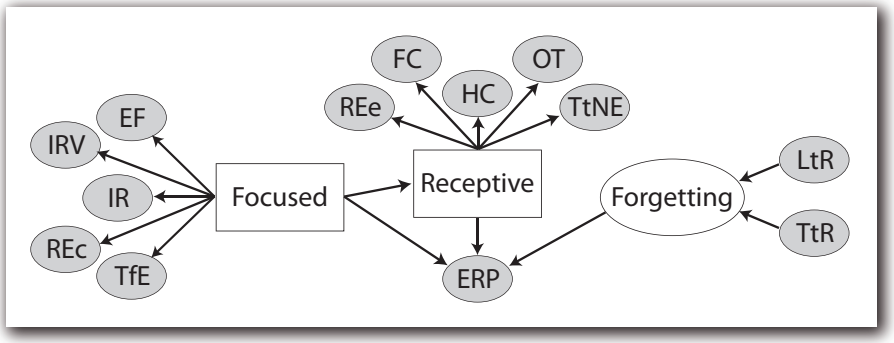

Figure 8.3: *The selected dynamic Bayesian net representation. Rectangle nodes denote dynamic states. Shaded nodes are observed.*

8.6 Results

In this section we will present the final Bayesian net representation of engagement dynamics and forgetting. The graphical model enables an investigation and interpretation of the dependencies in the selected model. Finally, we will evaluate the stability of the engagement model and how strongly it depends on the number of features selected by the LASSO method.

8.6.1 Engagement Model

Figure 8.3 shows the graphical model (FS $\rightarrow$ RS) best representing the data with a BIC of -718577, compared to -724111 (FS $\nleftrightarrow$ RS) and -718654 (FS $\leftarrow$ RS). The selected model comprises a conditional dependence of the *Receptive* on the *Focused* state. The relation between the *Focused* and *Receptive* state is illustrated by their joint probability distribution in Figure 8.4. As one can see, in a fully *focused* state, students are never found completely *non-receptive*. In contrast, students can be distracted (*non-focused*) despite being in a *receptive* state.

The ERP conditioned on the two states is presented in Figure 8.5. One can observe that the offset between top plane (*forgetting*) and bottom plane (*no forgetting*) is greater in the *focused* compared to the *non-focused* state. This underpins the assumption that in the *non-focused* state more non-serious errors are committed, of which the correct spelling is actually already known by the student. Therefore, the *Forgetting* has a lower impact on their ERP. As expected, the *non-receptive* state generally causes a higher ERP. Again, this effect on learning is reduced for non-serious errors in the *non-focused* state.

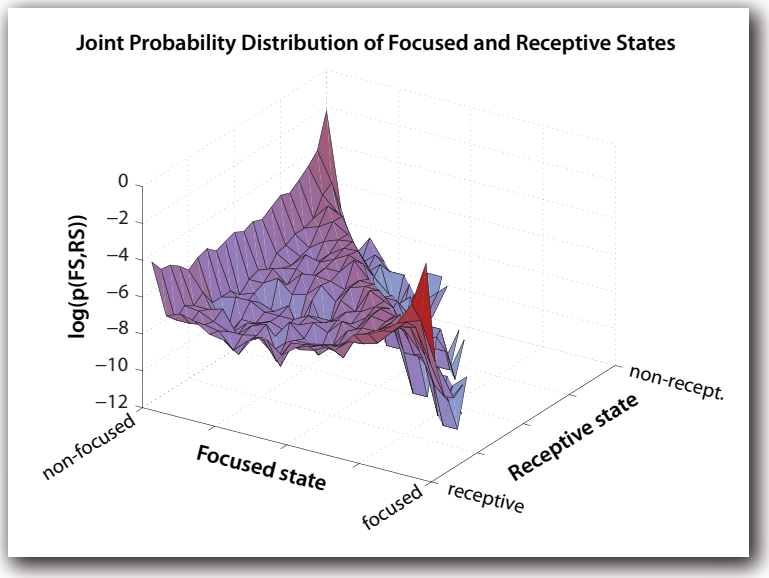

Figure 8.4: *Joint probability distribution of Focused and Receptive states.*

The estimated parameters of the conditional probability distributions for all observed nodes are presented in Table 8.2 (two right most columns). They demonstrate how the different states influence the mean or probability of the observed input measure. For example, the mean of the *Error Frequency* feature is lower in the *focused* than in the *non-focused* state (mind the *Exponential* scaling of the EF feature). The variance of the input rate is in general lower in the *focused* state. This observation holds true both inside (IRV) and between inputs (IR). In the *receptive* state we observe more correctly finished inputs and a lower probability for help requests by the student.

Another interesting finding yields the investigation of the age-dependence of engagement states. It shows that students below the median of 10.34 years exhibit a significantly ($p < 0.001$) higher probability of being classified as *non-receptive* (24.2%) and *non-focused* (32.5%) compared to those above the median (20.0% and 27.0%, respectively). This indicates that younger students tend to fall significantly more frequently into *non-focused* and *non-receptive* states.

8.6.2 Stability

To evaluate the stability of the graphical model, we investigate the influence of the stop criterion, employed in the feature selection, on the final Bayesian net representation. We stop the feature selection process of the LASSO method

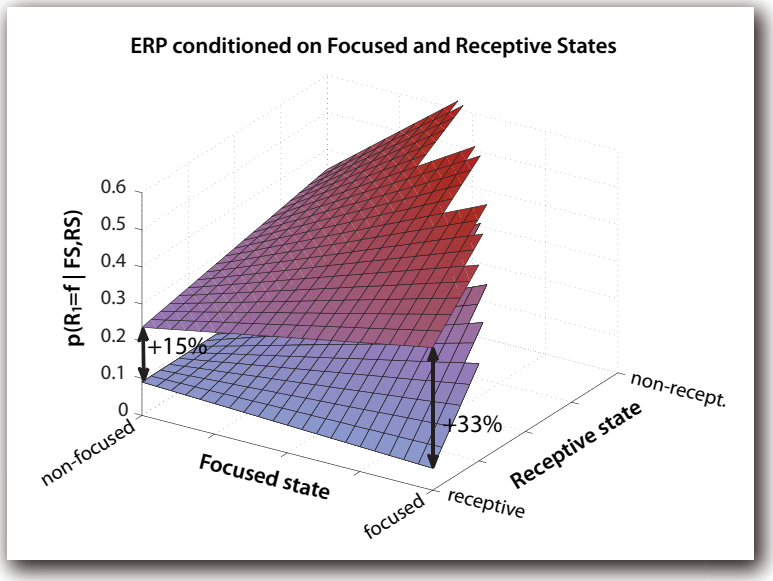

Figure 8.5: *ERP conditioned on engagement states for forgetting (top) and no forgetting (bottom plane). The ERP is plotted for all observed combinations of engagement states only.*

two and four features earlier and construct a model based on the reduced feature set. As illustrated in the bottom row of Table 8.2, the full feature set results in a logistic regression model with the highest BIC score and is therefore employed for the final model of engagement.

The main finding of the analysis is that the reduction of the model did not cause a severe alteration of the model. As illustrated in Table 8.2, the remaining features of the reduced models yield very similar distribution parameters as in the full model. Also the joint probability distribution of the ERP conserves the characteristics of having a greater difference between *forgetting* and *no forgetting* in the *focused* state and of having a generally higher repetition probability in the *non-receptive* state.

8.7 Conclusion

In this chapter we described the development of a model of engagement dynamics. The model is based on features extracted from the input data collected in the first user study. The regression analysis demonstrated that the systematic approach to feature processing for affective modeling increased the predictive power of the employed features. Especially all timing features benefited strongly from the appropriate scaling.

	4 Reduced		2 Reduced		Full	
Feature	$\mathbf{p_1}$[%]/**m**		$\mathbf{p_1}$[%]/**m**		$\mathbf{p_1}$[%]/**m**	
Focused State	focused	non-f.	focused	non-f.	focused	non-f.
EF	x	x	x	x	0.16	-0.34
IR	-0.45	1.00	-0.41	0.88	-0.41	0.87
IRV	x	x	-0.37	0.79	-0.36	0.78
REc	44%	33%	44%	32%	45%	32%
TfE	-0.14	0.30	-0.13	0.28	-0.13	0.28
Receptive State	receptive	non-r.	receptive	non-r.	receptive	non-r.
FC	x	x	96%	87%	95%	88%
HC	x	x	x	x	4%	28%
OT	-0.38	0.96	-0.35	0.81	-0.35	1.2
REe	0.15	-0.37	0.07	-0.15	0.07	-0.24
TtNE	0.15	-0.38	0.11	-0.26	0.11	-0.36
ERP						
Forgetting	focused	non-f.	focused	non-f.	focused	non-f.
receptive	0.68	0.72	0.67	0.89	0.62	0.76
non-r.	0.59	0.71	0.18	0.69	0.04	0.65
No Forgetting	focused	non-f.	focused	non-f.	focused	non-f.
receptive	1.00	0.97	1.00	0.92	0.94	0.91
non-r.	0.94	0.98	0.35	0.97	0.13	0.91
BIC of logistic regression						
BIC	−6426		−6388		−6369	

Table 8.2: *Optimal processing pipeline, estimated parameter b and significance for features selected by the LASSO logistic regression. Note that the exponential scaling inverts the orientation of a feature. The selected features are arranged according to their observed influence.*

The structure of the dynamic Bayesian network was identified on the basis of input and error behaviors alone. The model jointly represents the influences of *Focused* and *Receptive* states on learning, as well as the decay of spelling knowledge due to *Forgetting*. The presented causal model can be investigated and exhibits coherent conclusions and stability.

This core model can be extended with assessments of engagement of a different nature, such as sensor, camera or questionnaire data. This information allows us to verify the annotation of the hidden nodes and to relate the identified states to the underlying fundamental affective dimensions.

Part III

Evaluation

CHAPTER 9

Word Selection Controller

The goal of the student models presented in Part II is to enable an adaptation of the training software to the student's individual needs. In Part III of this thesis we aim at evaluating the developed models in a second user study.

For that purpose, we enhance the original software version by incorporating the gained insights and developed models. One main element of the phoneme-based enhancements is an adaptive word selection controller based on the student knowledge representation introduced in Chapter 6. In this chapter, we describe the concept and design of the novel controller in detail. The controller is implemented in the improved Dybuster version, which is employed in the second user study. The collected user data enables an evaluation of the gain in efficacy induced by the phoneme-based enhancements as described in Chapter 10.

9.1 Overview

In this chapter we will introduce our improved word selection controller. The controller relies on error prediction and classification, provided by the spelling knowledge representation, as well as on findings from log file analyses. The insights gained from the model of engagement have not been

incorporated, since the second user study started before the completion of the model.

As described in Chapter 3, the original word selection controller of Dybuster selects words from a module in an error entropy minimizing way. The error entropy is computed based on the global symbol confusion matrix and a local word error history. However, this original approach entails two problems:

1. The assignment of words to modules is precomputed and static. This makes a strong adaptation to the student's needs impossible.
2. The selection of new words from the modules and the scheduling of the training of these words are addressed by the same concept of error entropy. Despite featuring a certain mathematical elegance, this approach completely ignores any information about the timing of repetition. This lack often results in an immediate repetition of erroneously spelling words, due to a strong error entropy increase based on the committed errors.

The word selection controller presented in this chapter addresses the two drawbacks identified in the original version. It allows for a more extensive adaptation to the student and relies on the insights gained from the investigations of learning and forgetting. The function of the word selection controller in Dybuster can be separated into the following tasks, which will be described in more detail in the next two sections:

1. Select unprompted words from database for training.
2. Schedule the training of the present word set.

Notably, the controller design is chosen according to the specific setting of the application, i.e., a fixed three months training period and an available database of 1500 words. In the first user study, the children work on average on 800 of the 1500 words. Therefore, we focus on a selection of words from the database, which optimizes the training gains during the three months of training. The design of the controller has to be adapted, if it is employed for a long-term Dybuster therapy with an extended word database.

9.2 Word Selection from Database

One of the main tasks of the word selection controller is to decide which words from the database should enter the training process. Similar to the original approach where the error entropy is minimized, the enhanced word selection controller adapts the training to efficiently minimize the expected

number of errors in everyday writing. It takes account of the number of errors $\mathbb{E}[E|w]$ expected in the spelling of a word and its frequency of occurrence $F(w)$ in language corpora. This possible learning gain of a word is divided by the number of letters $L(w)$ to obtain the efficiency index $K(w)$ of the word.

$$K(w) = \frac{\mathbb{E}[E|w]F(w)}{L(w)}$$

The normalization by the number of letters $L(w)$ prevents a favor of long words, which take longer to listen to and to spell. The efficiency index returns the expected training gain in spelling per letter and consequently per time.

The expected number of errors $\mathbb{E}[E|w]$ considers all error categories except the typing errors. Generally, Dybuster is not a typing but a spelling training software and focuses on spelling rather than typing difficulties. For example, the fact that the key *'q'* has less surrounding keys accepted by Dybuster as input, thus featuring less typing error possibilities, does not decrease the training gain of words involving the letter *'q'*.

If new words are requested for training, the word selection controller selects words from the database with the highest expected learning gain $K(w)$. The controller does not have to follow a predefined course based on word modules, but can select from the entire word database. This results in a word set for training, optimized for a given student. However, the constant confrontation with the specific spelling difficulties can lead to high error rates and a frustration of the student. Therefore, we intersperse words with low expected number of errors $\mathbb{E}[E|w]$, if too many errors are committed by the student.

9.3 Training of Words

After having selected the words for training, there arise two main questions in the design of the word selection controller:

1. How often should a word be trained until it is considered as learned?
2. How should this training be scheduled?

The original word selection controller considers only the first question: every word has to be spelled twice correctly in a row. The word then enters a recap module and gets prompted once more approximately one month later. In the recap module, only one correct entry is required to end the training process of a word, independent of previous erroneous inputs in the recap module.

The issue of scheduling the training is not sufficiently addressed by the original controller. As the selection of words from the modules, the scheduling of the training is based on the error entropy minimization of the word selection and neglects any timing information. Actually, no distinction is made between already prompted and unprompted words, i.e., all words from the current module are considered as part of the training set. However, this error entropy approach leads to immediate repetition of words as well as very long periods between errors and repetition.

The wide distribution of repetition timings after erroneously spelled words in the first study enables an analysis of the effect of different scheduling options. In the next section we investigate the influence of the time to the first repetition after committed errors on learning. Following, we describe how these findings are incorporated into the novel word selection controller.

9.3.1 Optimal Point in Time for Repetition

Based on the collected user data we are able to investigate the impact of the time to repetition (TtR) onto the long-term learning of spelling. The goal is to find the optimal point in time for the first repetition R_1 after an erroneous input. As has been shown in Figure 6.8, the earlier an erroneous input is repeated, the lower is the corresponding ERP. At first appearance, that demands a repetition of erroneously entered words as early as possible, to avoid subsequent errors on the same word. However, the goal of repetition prompts is the long-term learning of the correct spelling. This effect is measured by means of the error repetition probability (ERP2: $P(R_2 = f)$) at the second repetition of the word (R_2), as illustrated in Figure 9.1. We consider only inputs with more than 12 hours between first and second repetition to exclude correct spelling in the second repetition by retrieval from the short-term memory. This ERP2 value provides a measure for learning efficacy, and is used to determine the optimal point in time for repetition. We expect the long-term learning to be influenced by two opposing effects:

1. A low TtR enables a correct R_1 due to retrieval from the short-term memory. If R_1 is correct after a longer time span, the probability is higher that the correct spelling was actually stored in long-term memory. Therefore, a large TtR will decrease the error repetition probability at R_2, if R_1 was correct (ERP2c: $P(R_2 = f|R_1 = c)$).
2. An erroneous R_1 after a short TtR does not strengthen a false representation in long-term memory as much as an erroneous R_1 after a

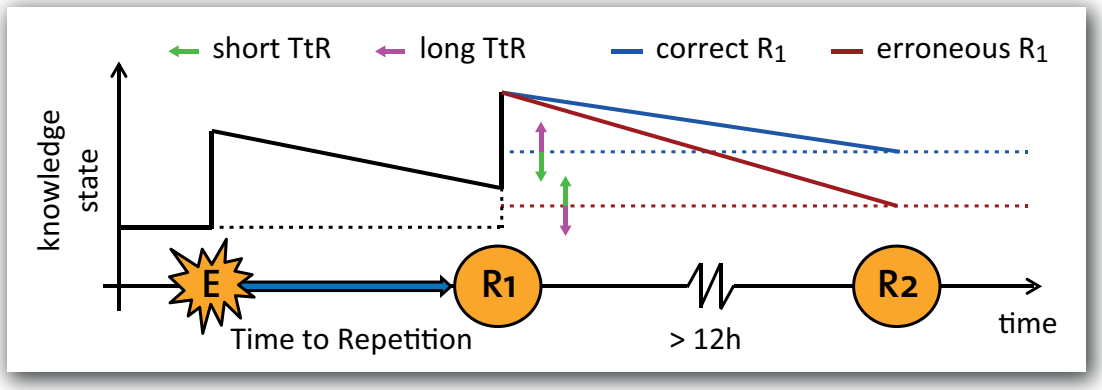

Figure 9.1: *Influences on second repetition R_2: (1) Short TtR lowers the value of R_1, i.e, positive effect of correct R_1 and negative effect of erroneous R_1 are weaker. (2) After long TtR the correct or erroneous R_1 have a strong influence on final knowledge state.*

long time period. Therefore, a large TtR will increase the error repetition probability at R_2, if R_1 was false (ERP2f: $P(R_2 = f|R_1 = f)$).

For PGM errors, these two effects are illustrated in Figure 9.2. The front and back row bars indicate the dependence of ERP2c and ERP2f on the TtR. The requested ERP2 value can now be computed by marginalizing out the first repetition (measured ERP is depicted by plane in Figure 9.2):

$$P(R_2 = f) = P(R_2 = f|R_1 = c)P(R_1 = c) + P(R_2 = f|R_1 = f)P(R_1 = f)$$

As can be seen in Figure 9.2, the ERP2 (middle row bars) for PGM errors is high for a short time span, due to the high ERP2c value. The ERP2 decreases with the ERP2c value, reaches a minimum at 3 to 6 minutes and rises again on account of the increasing influence of the ERP2f. This identifies a repetition between 3 and 6 minutes after a PGM error as most effective. Similar effects are shown by the phoneme omission and dyslexic confusion error categories. However, due to the random distribution of most typing and capitalization errors, a distinct point in time optimal for repetition cannot be found. We assigned a large optimal TtR to these two categories to lower their significance in the repetition scheduling.

The rather small ERP2 increase after the 3 to 6 minutes interval is not significant at the 5% level. This behavior is due to the fact that the original word selection controller does not distribute the repetition timing completely randomly, but is influenced by the change in error entropy after committed errors. Therefore, certain repetition patterns occur very rarely. For example, if R_1 is erroneous, the word is most likely repeated immediately. The case

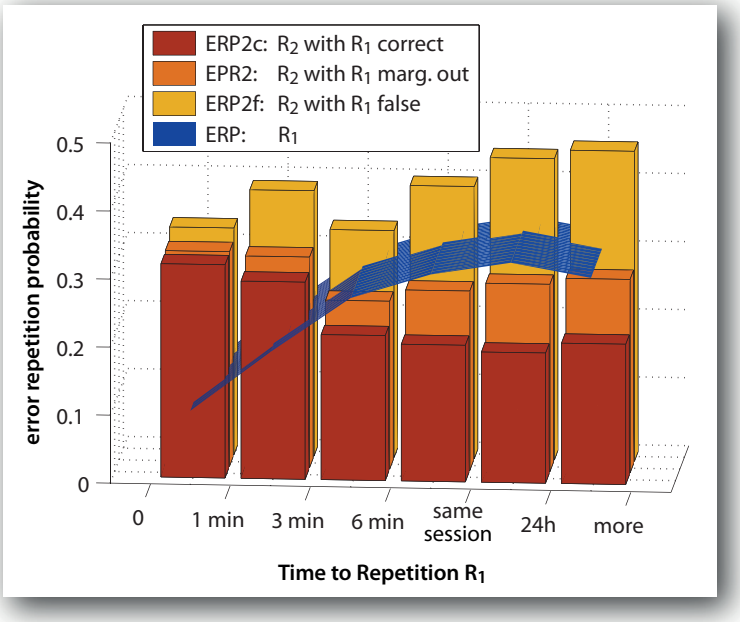

Figure 9.2: *Error probabilities at the first and second repetition of a PGM error dependent on the time between error and first repetition.*

of a time span of more than 12 hours between first and second repetition, requested to consider the word in the analysis, is only achieved, if the first repetition was conducted at the very end of a training session. This sparse sampling of certain repetition pattern inhibits significant results. Nevertheless, the analysis provides an indication for the optimal point in time and the gained insights are used in the design of the enhanced word selection controller.

9.3.2 Training Scheduling

In the novel word selection controller the structure of the original training process has largely been maintained. Generally, words have to be entered twice correctly and then recapitulated once more after a month. Two adjustments to this training process were made: first, words, which are entered correctly directly at the first prompt, immediately leave the training cycle and enter the recap cycle; second, it is possible for words to re-enter the training cycle after erroneous inputs in the recap cycle. These two changes enable, on the one hand, a speed up in processing of words whose spelling is already known. On the other hand, it renders an intensified training possible, if

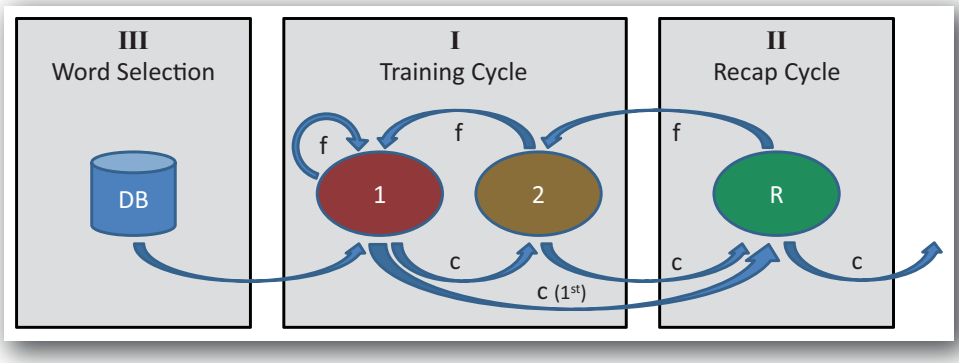

Figure 9.3: *Pool structure of the training process: After the selection from the database, the word passes through the training and the recap cycle until it leaves the training process. The priority of the word selection controller is: 1^{st} training cycle (1^{st} and 2^{nd} pool); 2^{nd} recap cycle; 3^{rd} selection from database.*

difficulties in spelling are still present in the recap cycle. The training process is illustrated in Figure 9.3.

One main improvement of the new word selection controller is the adjusted scheduling of the training. The estimated efficiency index $K(w)$ of a word w is only used for the word selection from the database. As soon as the word enters the training process, its training scheduling is solely dependent on empirically observed spelling difficulties and time. If words are spelled erroneously in the training cycle, the word selection controller determines the optimal point in time for repetition R_1, as presented in the previous section. After a correct R_1, the R_2 is scheduled for the next training session (24 hours later). The additional recapitulation of learned words is administered after one month, as by the original controller. The scheduling of R_2 and recapitulation is chosen based on expert knowledge of therapist, since a detailed analysis of the effect of different timings was not possible from the collected user data. The parameter space for the different timings of R_2 and the recapitulation is even larger compared to the analysis of R_1, which leads to a very sparsely sampled parameter space.

9.4 Implementation

In this section we describe how the enhanced word selection and training scheduling are implemented in the improved Dybuster version. The different cycles of the training process are represented by word pools, as illustrated in Figure 9.3. First, words are selected from the database and enter the training

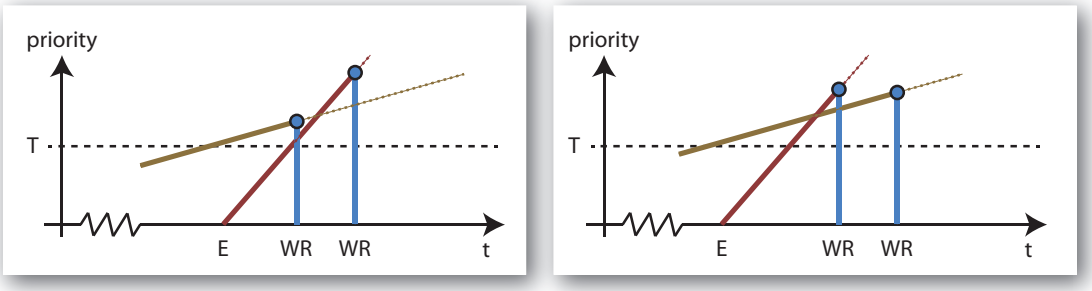

Figure 9.4: *Word selection controller: Repetition priority for words of 1^{st} training pool (red) increases faster than for words from 2^{nd} training pool (beige). Words are only selected if priority is above threshold T. In the left example the word from the 2^{nd} pool is selected at the first word request (WR). In the right example the priority for the word of the 1^{st} pool is already higher at the first WR and hence repeated first.*

process, traverses the different pools according to the amount of committed errors and leave the pipeline as learned and recapitulated. Whenever the word learn game requests the next word for training, the different pools are inquired for upcoming repetition requests. If none are available, new words are selected from the database, assigned to the 1^{st} pool and prompted for the first time.

However, the training scheduling of words from the different pools can interfere. For example, a word - transfered to the recap pool one month ago - needs a repetition at the same time as an error - committed 3 minutes ago - requests a repetition prompt. If the request originate from different cycles of the pipeline, the ordering is performed according to the time sensitivity, namely training before recapitulation. The ordering inside the training cycle is administered with respect to the repetition priority of a word. This priority value is computed based on the time since the last prompt and the desired point in time for repetition. The concept is illustrated in Figure 9.4. The desired point in time defines the slope of the priority increase. As soon as a word exceeds the threshold T, it can be selected at the next word request of the word learn game. If several words are above the threshold, the one with the highest priority at the given point in time is selected.

9.5 Conclusion

In this chapter we presented the novel word selection controller implemented in the improved Dybuster version. It is based on the information provided by

the phoneme-based spelling knowledge representation and the investigations of learning and forgetting on the collected user data. The controller allows for an adaptation to individual students by means of a word selection with highest expected learning gain. Additionally, it incorporates the insights from the analysis of error repetition into the scheduling of the training.

The enhanced software version is employed in the second Dybuster user study. The design of the controller is chosen with respect to this specific setting of a three months training period and a limited word database. Since the second study started before the completion of the model of engagement, the model has not been incorporated into the controller. However, this offers the opportunity for investigations of the learning gain induced by the phoneme-based enhancements alone, which are presented in the following chapter.

CHAPTER 10

Learning Progress

In this thesis we introduced a textural code representing phonological information and a phoneme-based student knowledge representation, which led to an adaptive word selection controller. These phoneme-based enhancements are implemented in an improved Dybuster version and evaluated in the second user study. In this chapter we will investigate the learning progress of the children from the two user studies. This analysis will allow us to draw conclusions about the gain in learning progress induced by the phoneme-based enhancements as well as about the influence of different cognitive factors on the learning process.

10.1 Overview

In this chapter we present the investigation of the spelling progress of children working with Dybuster. We analyze the spelling behavior by means of learning curves based on the collected log file data of the two user studies.

First, we compare the learning progress from the first and the second study. This allows for an evaluation of the phoneme-based enhancements, consisting of a new phonological code (see Chapter 3) and an adaptive word selection controller (see Chapter 9), implemented in the learning software. We expect

the dyslexic children who work with the enhanced software version to improve their spelling behavior significantly faster than dyslexic individuals who work with the original version.

Second, based on data collected in our second study, we investigate the influence of different cognitive factors on the learning progress. These factors include: the indication of dyslexia, attention functions, and memory performances. Comparing children with and without dyslexia allows us to explore whether both groups benefit to the same extent from the training or if children with dyslexia, irrespective of the method used, generally experience more problems acquiring spelling knowledge.

We decided to evaluate attention and memory functions because, on the one hand, it has been suggested that reading problems are associated with impaired memory functions [SKD+04], which in turn cause reduced phonological representations. On the other hand, attention functions build the general basis for learning, since attention processes control all functions of our cognitive system, provided that tasks are not over-learned and automated [ZGF02]. Attention helps people focus on the relevant information [PP87]. Therefore, we aim to examine the influence of memory and attention functions on the spelling progress acquired in a structured environment.

In the following, we will describe the concept of learning curves, which is employed for the comparison of the learning progress between different groups. Section 10.3 specifies the investigated error categories and the compared groups in detail. The final results of the learning progress evaluation are presented in Section 10.4.

10.2 Learning Curves

We investigate the learning progress of both studies by means of learning curves. The concept of describing practice effects by simple nonlinear functions in a broad range of tasks is presented in Newell and Rosenbloom's "Mechanisms of Skill Acquisition and the Law of Practice" [NR81]. It has become a well-established procedure in the psychology of learning to analyze learning behavior based on such learning curves. However there is an ongoing debate regarding which decay function best fits the relation between proficiency and number of practice trials. Based on the findings of Heathcote et al., we decided to rely on an exponential law of practice [HBM00]. This exponential law of practice describes the process of learning by an exponential decay function

$$\mathbb{E}[E(t)] = a'e^{-bt} + c$$

where $\mathbb{E}[E(t)]$ is the error expectation value at time t. For the comparison of the spelling progress of two groups we are interested in the initial error expectation ($a = a' + c$: error expectation value at time $t = 0$), the learning progress (b) and the asymptotic error expectation (c : error expectation value for time $t \to \infty$). Therefore we perform the variable transformation $a = a' + c$ and obtain the exponential decay function

$$\mathbb{E}[E(t)] = (a - c)e^{-bt} + c$$

The error expectation values $\mathbb{E}[E(t)]$ are collected for training days only, i.e., we count the days the children were really working with the training software. Additionally, we only consider the first prompt of each word for every student. This protocol enables us to exclude repetition effects and supports an investigation of the general spelling performance. The error expectation value $\mathbb{E}[E(t)]$ at day t is computed by dividing the number of committed errors $Y(t)$ by the number of error possibilities $N(t)$ of all students of a given group. A weighted nonlinear least squares method is employed to estimate the parameters for the exponential fit to a dataset. The number of error possibilities ($N(t)$) are used as weights for the estimation.

To compare two groups of students and evaluate the significance of the difference between the two regressions we run a combined estimation. Every parameter p is replaced by a term $p(1 + d_p g)$, consisting of an absolute parameter p for the group $g = 0$ and a relative parameter d_p, denoting the relative difference of the parameter p between the first ($g = 0$) and the second ($g = 1$) group. This results in an estimation of the following form:

$$\mathbb{E}[E(t,g)] = (a(1 + d_a g) - c(1 + d_c g))e^{-b(1+d_b g)t} + c(1 + d_c g)$$

where g equals zero for the first group and one for the second; d_a, d_b and d_c indicate the percentage difference between the corresponding parameters of the two groups and their t-tests return a measure for the significance of the difference. The introduction of the additional three parameters to the regression model can lead to overfitting. This is especially the case if the data contains no differences in initial error expectation value, learning progress and asymptotic error expectation value between the groups. To avoid overfitting and to reduce the model for estimation, we run a backward model selection based on the BIC score. Removed features will be marked

with an ”R”, to indicate that the data provides more evidence for a model without the parameter.

To take account for the within group variation of parameters, we considered the employment of non-linear mixed effects (NLME) models. They enable a joint representation of inner and inter group differences by introducing four additional parameters [PB00]. However, the investigation of NLME models on a control sample of the comparisons yielded equal conclusions. Therefore, we forwent the employment of NLME models for simplicity reasons.

10.3 Data Analysis

In this section we specify the learning progress comparisons in detail. This includes the description of the investigated error categories as well as the compared groups.

Error Categories

Based on the student knowledge representation presented in Chapter 6 we are able to investigate the spelling progress on individual error categories. The progress analyses are performed on phoneme-grapheme matching (PGM) and typing errors. These categories comprise approximately two third of all errors and represent very different difficulties in spelling as described below. This allows for an investigation of the progress on distinct categories with large enough data set to provide significant results.

As described in Chapter 5, PGM errors reflect difficulties in the phoneme to grapheme mapping process. These are mostly additions or omissions of silent letters or doubling of letters and are a major difficulty for children with dyslexia. PGM errors account for approximately 30% of all committed errors during both studies (see Section 6.8). Since the different grapheme representations of a phoneme all sound the same, the correct matching has to be learned by heart or by acquiring rules. Therefore, the progress in the PGM error expectation value is an appropriate measure for the learning behavior.

In contrast to PGM, typing errors are mostly accidentally committed errors that are not related to specific spelling difficulties of words (see Chapter 5). Typos account for approximately one third of all committed errors. Due to the fact that typing errors are unsystematic and independent of general spelling difficulties, we expect the progress over time to be independent of the prompted words, and hence, to be equal for both software versions.

		Dyslexic		Control		t-Test	
	Cognitive Function	**Mean**	**SD**	**Mean**	**SD**	**T**	**p**
A	Alertness	46.5	25.9	54.4	24.2	-1.22	0.23
	Flexibility	46.5	32.7	52.6	33.8	-0.72	0.48
	Impulse Control	41.2	29.5	49.2	29.5	-0.94	0.35
M	Learning Performance	46.1	30.9	51.4	30.2	-0.67	0.50
	Short-term Memory	55.8	29.8	58.1	28.5	-0.30	0.77
	Long-term Memory	59.7	26.5	61.1	25.8	-0.20	0.84

Table 10.1: *Cognitive function comparison (A: attention and M: memory) for children with and without dyslexia. No significant confounding effects of dyslexia with the cognitive functions were found.*

Comparisons

First, we compare the log file data from dyslexic children of the first and the second study. The learning progress is evaluated by PGM errors as well as typing errors. Since not all children achieved the maximal amount of training sessions, the analysis is performed on the first 30 training days. The training times (see Section 4.7) during the analyzed first 30 days do not significantly differ between the dyslexic children of the first and second user study; thus, enabling us to investigate whether the children can benefit from the phoneme-based enhancements of the spelling training software, and in which error categories the benefit becomes manifest.

Second, we run investigations on the log file data collected during the second study alone. We present the comparisons of children with and without dyslexia, as well as the comparison of different groups based on attention functions and verbal memory skills. In this analysis we investigate the progress on phoneme-grapheme matching errors, to observe the increase in actual spelling knowledge. For the attention function and memory performance analysis, we classify all children based on their performance in the standardized neuropsychological tests described in Section 4.3, independent of their indication of dyslexia. In all testings the subjects are classified as low or high scorers, if they performed below or above the norm interval ($30\% - 70\%$), respectively. To examine confounding effects between cognitive functions and the absence or the presence of the diagnosis of dyslexia, we apply a t-test for independent samples. The outcome of this analysis is presented in Table 10.1 and demonstrates that there are no significant differences in the cognitive functions between the two groups (dyslexic vs. control). The

number of dyslexic and control children in each group is presented in the corresponding tables in the Results section.

To further investigate how the subtests of the attention and memory functions are related to each other we apply a parametric correlation analysis. The outcome of the parametric correlation analysis of the subtests of attention functions yields that alertness and flexibility correlate significantly ($r = 0.277$, $p < 0.05$; all two tailed). Additionally, the analysis evidences that all subtests of the memory functions correlate significantly with each other, such as learning performance with short-term memory ($r = 0.652$, $p < 0.01$), learning performance with long-term memory ($r = 0.595$, $p < 0.01$), and short-term memory with long-term memory ($r = 0.761$, $p < 0.01$).

However, the main finding of this analysis is that attention functions are orthogonal to memory functions. None of the subtest belonging to the attention function (alertness, flexibility, and impulse control) correlates significantly with any subtest of the memory skills (learning performance, short-term memory, and long-term memory). Since, our dyslexic sample is in addition not confounded with attention functions or memory skills, we are able to independently investigate the influence of individual cognitive functions on the acquisition of spelling skills.

10.4 Results

In this section we present the evaluation of the phoneme-based enhancements and the comparisons of high and low scorers in different cognitive functions. The results of the learning curve estimations are illustrated in the figures and tables below. If not stated otherwise, the black and red lines illustrate the fitted learning curves for both groups. The red and black points show the measured error expectation values at a given day for the two groups. The plotted error bars denote the 95% confidence intervals for the expectation value measure of the analyzed error category at this day.

In the tables, the parameters of the first group are given in absolute values. The difference to the second group is displayed by the relative change. p shows the significance of each parameter. The initial error expectation value a describes the expectation value of errors at the beginning of the study, which corresponds to the axis intercept. Additionally, d_a represents the relative difference between the first and the second group. The learning progress b demonstrates the slope of the learning curve and depicts the speed at which children improve during training. The relative difference of the slope between groups is denoted d_b. The asymptotic error expectation value c

indicates the limit of the children's training performance and d_c characterizes the relative differences of this factor.

10.4.1 Evaluation of Phoneme-based Enhancements

The amelioration of the learning progress for dyslexic children from the original to the enhanced Dybuster version is evaluated based on phoneme-grapheme matching errors (PGM) and typing errors (Typo). Figure 10.1 illustrates the expectation values of PGM and typing errors for children with dyslexia of the first and second user study.

Phoneme-Grapheme Matching

As expected, both groups with dyslexia start with the same error expectation value ($d_a = \mathrm{R}$) and show no difference in the asymptotic error expectation value ($d_c = \mathrm{R}$) for PGM errors (see Table 10.2). The fact that the children with dyslexia in the first and second study show comparable initial and asymptotic error expectation values on PGM underpins the notion that the two groups do not differ from each other a-priori.

The main finding of this analysis is that the learning progress of the group who undergoes spelling training with the new phoneme-enhanced software version is 154% higher, than the progress of individuals who experience training with the old spelling program. This result evidences that dyslexic children working with the new software version benefit significantly more from the training (d_b: $p =$ 2e-7).

Table 10.2 additionally shows the estimated parameters for the comparison of all children (dyslexic and control) from the first and second study. The learning progress in spelling over all children is also significantly increased with the new software version (d_b:, $p =$ 2e-16), but only by 104%. This indicates that the phoneme-based enhancements of the software version primarily supports the dyslexic children.

Discussion The results evidence that the new word selection controller is able to adapt the training to individual students and increase the training gain of the analyzed period of 30 training days. Although we can not identify a common spelling difficulty of all dyslexic children, the individual subjects suffer from very specific spelling difficulties. The old word selection method of Dybuster constrained the word selection by a devision of the word database into separate modules and relies on a letter-based analysis of

errors. In contrast, the novel controller accounts for spelling difficulties on a phonological level, selects words from the entire database, and hence enables an adaptation to these individual spelling difficulties. Therefore, the children repeatedly work on their individual spelling problems. Consequently, the children learn the linguistic spelling rules based on the German language and generalize them to other words after training.

Additionally, the phoneme-based textural code supports children in their learning behavior. The code is implemented based on the notion that the core problem of dyslexia is a phonological processing deficit. This deficit becomes manifest in reduced phoneme to grapheme mapping skills [RRD+03]. The additional textural code supplies easily extractable information about the phonological word structure. This results in a segmentation of the word in phonemes and supports the association with their graphemes. The visualization of the association between phonemes and graphemes strengthens the phonological awareness and mainly supports the dyslexic subjects in their spelling training.

Typing Errors

In a supplementary analysis, we investigate the error behavior on the Typo category. Table 10.2 illustrates that the initial expectation value of typing errors ($a = 0.0006$) is orders of magnitudes lower than for PGM errors ($a = 0.031$). However, since the possibilities for PGM errors occur much less frequently

	Initial error exp. val.		**Learning progress**		**Asymptotic error exp. val.**	
	a	**p**	**b**	**p**	**c**	**p**
	d_a	**p**	**d_b**	**p**	**d_c**	**p**
PGM						
1^{st} Study All (abs.)	0.024	2e-16	0.046	4e-07	0.0071	2e-08
2^{st} Study All (rel.)	R		+104%	2e-16	R	
1^{st} Study Dys. (abs.)	0.031	2e-16	0.050	7e-09	0.0091	7e-12
2^{st} Study Dys. (rel.)	R		+154%	2e-07	R	
Typing error						
1^{st} study dys. (abs.)	0.0006	2e-16	0.011	0.0039	R	
2^{st} study dys. (rel.)	R		R		R	

Table 10.2: *Estimated parameters of the comparison between participants from the first and second study. Learning curves are computed for PGM and typing errors.*

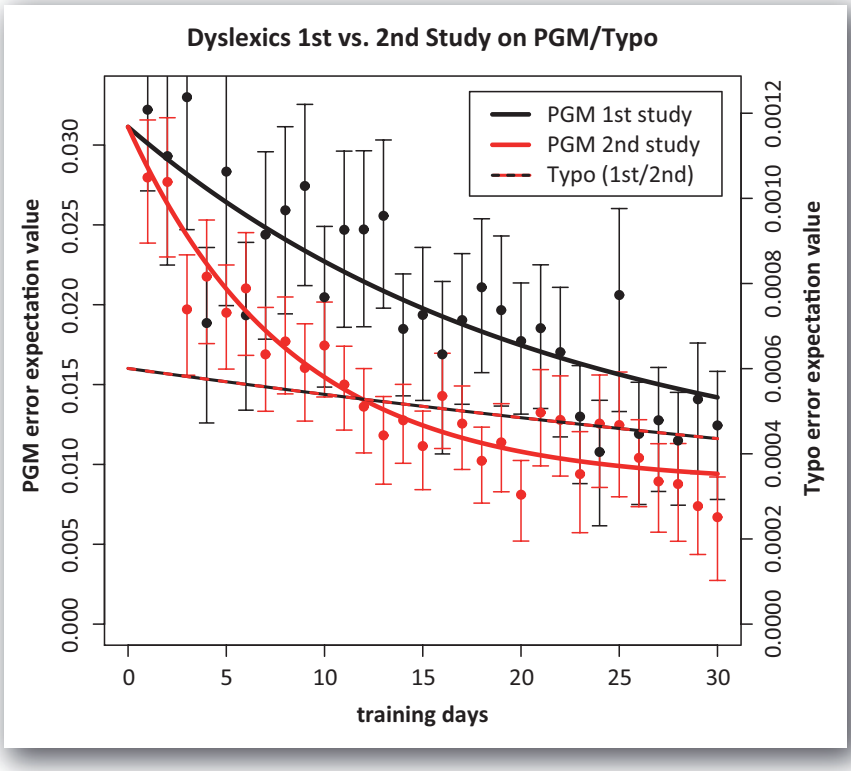

Figure 10.1: *Learning curves of PGM (bold lines; left y-axis) and Typo (thin line, right y-axis) errors for the children with dyslexia from the first (black) and second (red) studies. The points and error bars illustrate the PGM error probability estimate for a given day and its 95% confidence intervals. The spelling improvement of individuals with dyslexia from the second study (vs. first study) on PGM was significantly higher; however, in both groups the same typing error behavior was observed.*

than for typing errors, both categories account for one third of all committed errors each. In contrast to the group differences in PGM errors, the analysis of the typing errors reveals that both groups commit approximately the same number of typing errors at the beginning ($d_a = \mathrm{R}$) and at the end ($d_c = \mathrm{R}$) of the training. Moreover, the two groups reduce their typing errors to the same extent ($d_b = \mathrm{R}$); however, the learning progress on PGM ($b = 0.05$) as compared to Typo ($b = 0.011$) is substantially higher.

Discussion The slight improvement on typing errors can be explained by the lack of experience of 8- to 12-year-old children in working at a keyboard. We assume that the children gain knowledge about the key distribution on

the keyboard through training, which results in a slight reduction of the typing errors expectation value.

10.4.2 Influence of Dyslexia, Attention and Memory Functions

The investigation of the influence of cognitive functions are based on the log file data collected in the second user study. Since no confounding effects between the different cognitive functions were found, the respective groups can be compared independently.

Dyslexia

Figure 10.2 presents the learning curves of PGM errors for dyslexic and control children participating in the second study. As Table 10.3 depicts, controls (as compared to dyslexics) show 21.8% fewer spelling errors at the beginning of the training (d_a: $p =$ 5e-08). Both groups are able to significantly improve their spelling proficiency during the training (b: $p =$ 3e-08). The most important effect yielded from this analysis is that both children with and without dyslexia exhibit the same learning progress ($d_b =$ R). This result evidences that both groups benefit from the training to the same extent.

Furthermore, children without dyslexia as compared to children with dyslexia show a slightly lower asymptotic error expectation value (d_c: $p = 0.04$). This indicates that despite a similar training progress, dyslexic subjects are not expected to attain the same spelling skills even after a long period of training.

Discussion The results demonstrate that the multi-modal training induces a significant decrease in spelling errors, particularly phoneme-grapheme matching errors, in both children with and without dyslexia. This progress

		Initial error exp. val.		**Learning progress**		**Asymptotic error exp. val.**	
		a	**p**	**b**	**p**	**c**	**p**
		d_a	**p**	**d_b**	**p**	**d_c**	**p**
Dyslexic	(abs.)	0.029	2e-16	0.100	3e-08	0.0080	7e-10
Control	(rel.)	-21.8%	5e-08	R		-18.3%	0.040

Table 10.3: *Estimated parameters of the PGM error comparison between dyslexic and control children of the second study.*

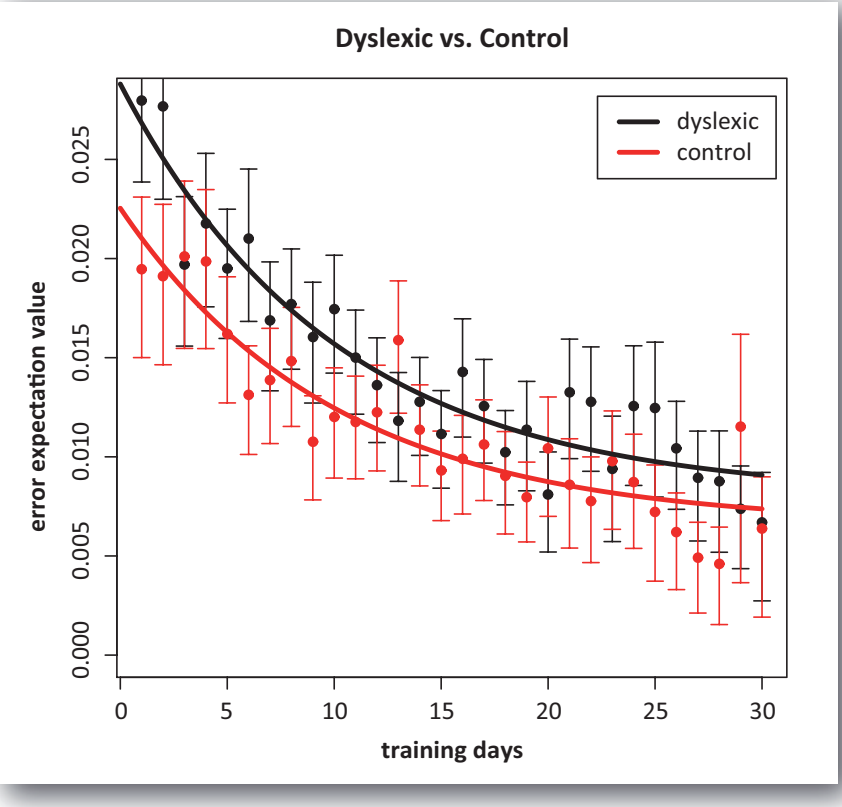

Figure 10.2: *Learning curves for PGM errors of dyslexic and control children from the second study. Both groups were able to improve their spelling skills to the same extent.*

is found for words that are presented for the first time. Therefore, children with as well as without dyslexia show that they not only memorize the word-form of the target words, i.e., the correct spelling, but that they are able to generalize concepts and adopt rules based on the German language.

Additionally, the analysis evidences that both groups benefit from the training to the same extent. Children with dyslexia are characterized by poor phonological awareness, which is attributed to difficulties in memorizing the phoneme-grapheme associations. Since it is known that dyslexic individuals use a non-phonological, visual coding strategy for memorizing information [MK09], the association between phoneme and grapheme were linked with a non-verbal textural code. Hence, dyslexic children are faced with a naturally occurring visual coding strategy that facilitates the memorization of the word form. Therefore, the multi-modal, nonverbal cues implemented in the training software allows dyslexic and control children to improve their phoneme-grapheme conversion knowledge in a similar way.

Attention

In a further step, we analyze how the attention functions influence the phoneme-grapheme matching progress. This involves comparing children with low attention functions to children with high attention functions, based on the data of the second study. We compare the children based on the attention functions impulse control, flexibility and alertness, described in detail in Section 4.3. As displayed in Table 10.4, our data shows that children with high compared to low impulse control ($d_a = -47.0\%$, $p =$ 1e-14), flexibility ($d_a = -46.4\%$, $p =$ 2e-16), and alertness scores ($d_a = -12.3\%$, $p =$ 0.0038) commit significantly fewer spelling errors at the beginning of the training. These findings indicate that low scorers do not benefit as much from traditional teaching and schooling in orthography as high scorers.

Notably, children with low attention functions (i.e., impulse control ($d_b =$ R), flexibility ($d_b =$ R), and alertness ($d_b =$ R)) show a similar training progress of the first 30 training days as the corresponding high attention score group. This shows that both groups (low and high attention scores) benefit from the computer-based training to the same extent.

Additionally, the two groups do not differ in their asymptotic error expectation value in all attention functions ($d_c =$ R). Therefore, it can be expected that children with low attention functions will be able to attain the same spelling

	Initial error exp. val.		**Learning progress**		**Asymptotic error exp. val.**	
	a / **d_a**	**p** / **p**	**b** / **d_b**	**p** / **p**	**c** / **d_c**	**p** / **p**
Impulse Control	(below: 11 dys./4 con. - above: 6 dys./6 con.)					
Below Norm (abs.)	0.032	2e-16	0.074	2e-06	0.0066	2e-05
Above Norm (rel.)	-47.0%	1e-14	R		R	
Flexibility	(below: 14 dys./5 con. - above: 13 dys./8 con.)					
Below Norm (abs.)	0.042	2e-16	0.127	5e-11	0.0085	3-16
Above Norm (rel.)	-46.4%	2e-16	R		R	
Alertness	(below: 11 dys./5 con. - above: 7 dys./8 con.)					
Below Norm (abs.)	0.028	2e-16	0.044	2e-16	R	
Above Norm (rel.)	-12.3%	0.0038	R		R	

Table 10.4: *Estimated parameters of the PGM error comparison between high and low scorers in the attention functions impulse control, flexibility and alertness.*

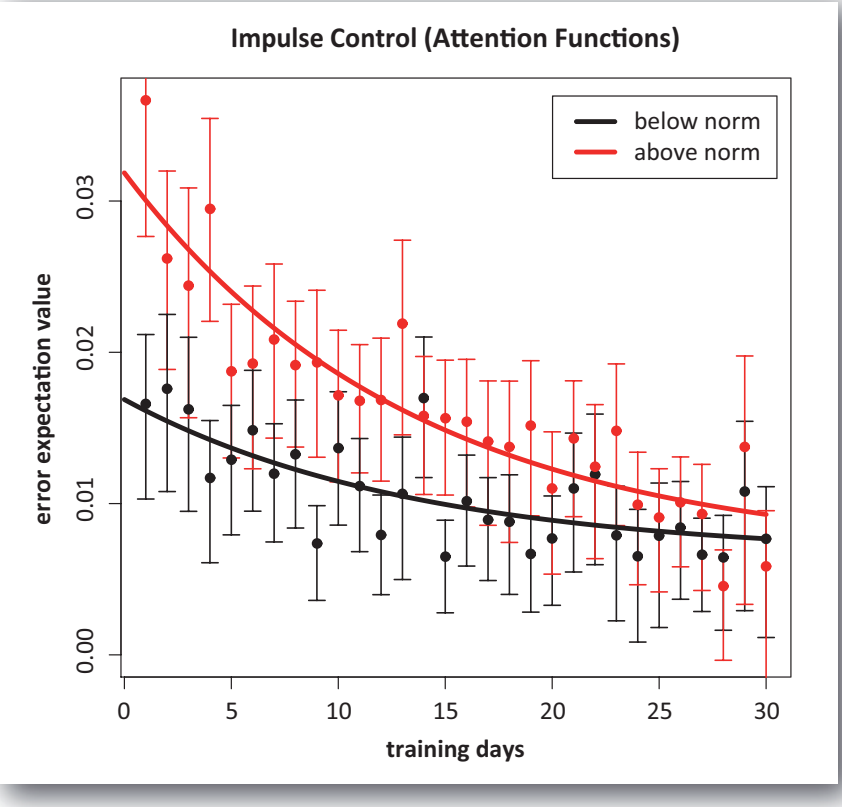

Figure 10.3: *Learning curves for PGM errors of children with high and low impulse control scores from the second study. Low and high scorers can benefit similarly from the structured environment and the implemented audiovisual codes of the learning software.*

level as children with high attention functions. As an example, Figure 10.3 illustrates the learning curves for the comparison of the groups with high and low impulse control.

Discussion We suggest that working on the computer facilitates children to structure their working strategy and supports them with focusing on the relevant task. The structural guidance is enforced with the interface of the topological code, which assists the users in their serial behavior of putting the correct letters in the right position. The support in focusing the attention on the relevant stimulus might be beneficial for children with reduced attention functions. Our findings are in line with previous evidence that children with ADHD can also improve their spelling skills when a clear strategy is taught [RCC08].

Memory

In this last analysis of spelling curves we aim to examine the influence of memory performances on spelling skills. Our data indicates that the initial error expectation value does not significantly differ between children with high and low learning performance, short-term memory, as well as long-term memory scores (see Table 10.5 for details). The respective groups also do not feature any differences in the asymptotic error expectation. However, compared to low scorers, children with high scores in learning performance ($d_b = +126\%$, $p = 0.0016$), short-term memory functions ($d_b = +175\%$, $p = 0.0015$), and long-term memory functions ($d_b = +226\%$, $p = 8\text{e-}05$) benefit significantly more from the computer based training. Figure 10.4 displays the difference between high and low learning performance.

Discussion These results are consistent with the notion that children's abilities to store and manipulate information in complex memory may have strong influence on learning [GAWA06]. Our data evidences that children with high memory performance benefit greatly from the information provided by the multi-modal learning software as it strengthens the retrieval of letters and phonemes stored in memory structures. However, children with low memory performance show difficulties to cope with the amount of

	Initial error exp. val.		**Learning progress**		**Asymptotic error exp. val.**	
	a / $\mathbf{d_a}$	**p** / **p**	**b** / $\mathbf{d_b}$	**p** / **p**	**c** / $\mathbf{d_c}$	**p** / **p**
Learn. Performance	(below: 14 dys./7 con. - above: 9 dys./7 con.)					
Below Norm (abs.)	0.026	2e-16	0.079	2e-8	0.0072	2e-12
Above Norm (rel.)	+12.7%	0.212	+126%	0.0016	R	
Short-term	(below: 10 dys./6 con. - above: 12 dys./10 con.)					
Below Norm (abs.)	0.022	2e-16	0.036	2e-05	0.0051	0.0017
Above Norm (rel.)	+15.9%	0.090	+175%	0.0015	R	
Long-term	(below: 7 dys./5 con. - above: 12 dys./9 con.)					
Below Norm (abs.)	0.025	2e-16	0.039	5e-05	0.0064	3e-06
Above Norm (rel.)	R		+226%	8e-05	R	

Table 10.5: *Estimated parameters of the PGM error comparison between high and low scorers in the memory functions learning performance, short-term and long-term memory.*

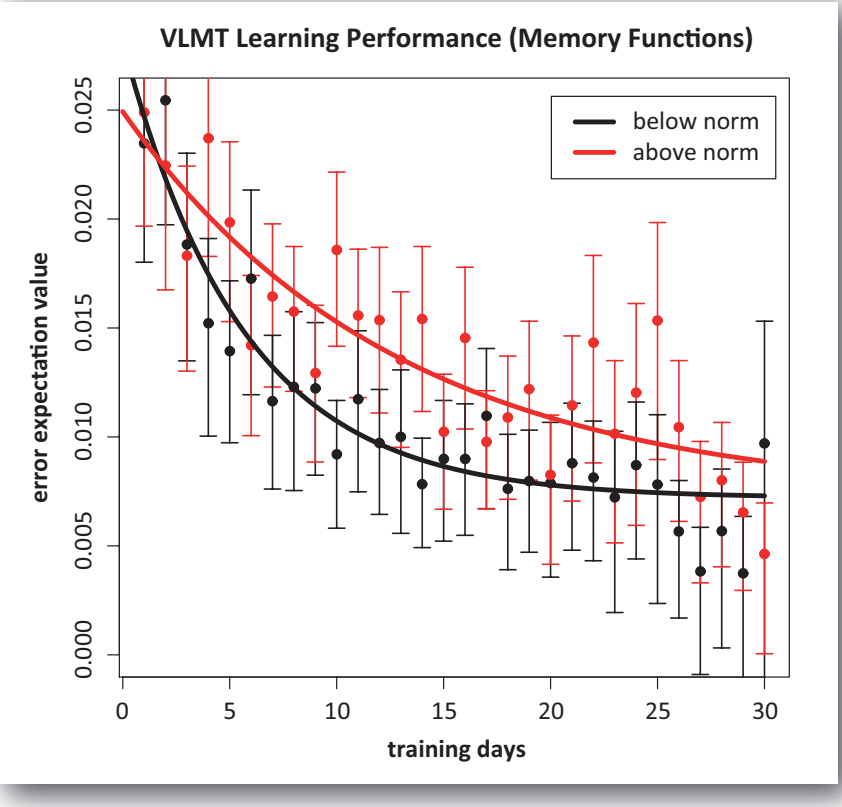

Figure 10.4: *Learning curves for PGM errors of children with high and low verbal learning performance (VLMT) scores from the second study. Children with high scores improved their spelling skills significantly faster than children with low scores.*

provided information. They are not able to extract the spelling information from the different representations as much as children with high memory performance.

10.5 Conclusion

Our results demonstrate that the phoneme-based enhancements implemented in the Dybuster spelling software positively influence the spelling performance of dyslexic children. The participants working with the student-adaptive software version show a significantly increased learning progress. Additionally, there is evidence that both children with and without dyslexia profit from the computer-based training in a similar way. Both groups were able to use the visual and auditory coding system implemented in the learn-

ing software to acquire spelling skills and facilitate the memorization of phonological information.

Children with low (vs. high) attentional performances could benefit equally from the structured computer-based learning software. This finding implicates that children with low attention resources need clear guidance and may benefit from a structured methodological approach. Moreover, we were able to show that memory functions correlate positively with learning progress irrespective of dyslexia. This indicates that memory functions are important cognitive sources for acquiring spelling skills in such a multi-modal learning environment.

CHAPTER 11

Conclusion

In this thesis we presented an entire loop in the data-driven development of an intelligent tutoring system. The work is based on the multi-modal spelling software Dybuster. We started our investigations on the basis of the data collected in a large-scale user study in 2006. This data led to the development of a novel spelling knowledge representation and a model of engagement. Based on the gained insights and the developed models, we extended the original software with phoneme-based enhancements, consisting of an adaptive word selection controller and a textural code. Finally, the improved software version was evaluated in a second user study.

In the following, we will review the principle contributions of the thesis and discuss its limitations and further work.

11.1 Review of Principle Contributions

We developed an error taxonomy for the specific setting of isolated word spelling, structured according to the requested information to describe errors. We presented a corresponding set of error generating, letter and phoneme level mal-rules. These mal-rules build the foundation for the development of a perturbation model representing the student knowledge in spelling. We identified an inference algorithm based on a Poisson regression with a linear

link function as most suitable to allow for student model estimations on unclassified student inputs. The appropriateness of the chosen approach has been demonstrated by a residual analysis of different link functions and manifests in more reliable estimations. The estimated student characteristics enable an intelligent tutoring system to compute local (error classification) and global (error prediction) information about the student. Interestingly, we couldn't identify purely dyslexia specific error patterns in the input data. Although, children with dyslexia show generally higher error rates, the dyslexic and control group feature on average similar error distributions. However, there are strong within group differences, which request an adaptation to the individual student.

Based on the student knowledge representation and insights gained in the analysis of learning and forgetting, we designed an improved word selection controller. The controller selects words from the entire word database and allows for an adaptation to the student's strengths and weakness without following prescribed tracks. It enables children to work independently according to their own learning pace. In addition, we presented a textural code representing the phonological structure of a word. In conjunction, these modifications build the phoneme-based enhancements implemented in the improved software version.

The improved Dybuster version was evaluated in a second user study. The learning progress comparison between first and second study evidence that dyslexic children benefit significantly from the phoneme-based enhancements. Children with dyslexia even improve their spelling skills to the same extent as children without dyslexia, and were able to memorize phoneme to grapheme correspondence when given the correct support and adequate training. The learning curve analysis additionally demonstrated that children with low attention functions benefit from the structured learning environment. In general, our data showed that memory sources are supportive cognitive functions for acquiring spelling skills and for using the information cues of a multi-modal learning environment.

In addition to the modeling of the actual spelling knowledge, we addressed the question of short-term variation in the students affective states. We introduced a systematic approach to incorporate domain knowledge in feature processing for affective modeling. The presented method employs time scale separation and normality-maximizing scaling and is implemented in a modular affective modeling framework. We demonstrated the advantages of feature processing for affective modeling in the development of a model of engagement. It enabled the identification of the dynamic Bayesian network model directly from spelling software logs. The model jointly represents the

influences of focused and receptive states on learning, as well as the decay of spelling knowledge due to forgetting.

11.2 Limitations and Further Work

The presented work covers many areas of interest, from psychology over linguistics to student modeling. In this chapter we will investigate the limitations of this interdisciplinary thesis and discuss the potential of further work in the respective areas.

The core of the presented models is the error taxonomy for isolated word spelling. Errors are analyzed on a letter and phoneme level. However, research on frequency effects in language acquisition proposes a different viewpoint on spelling [Ell02]. The frequency of occurrence of letter sequences can be employed for an explanatory model of spelling errors. By introducing *Main* and *Special* graphemes in our mal-rules we incorporated basic elements of this modeling concept. However, the extension of the set of mal-rules with more sophisticated representations of frequent spelling patterns has the potential to capture spelling difficulties not accounted for in our model.

The student knowledge representation and corresponding inference algorithm was designed according to the nature of the collected user data, i.e., that the students trained for a period of maximum three months. Due to the slowly changing spelling characteristics the entire input history of a student was used for spelling knowledge estimation. However, in the application outside of the laboratory setting, children might train over much longer periods and the student characteristics might vary strongly. Aging-schemes of input data or the incorporation of learning curves into the spelling knowledge representation, as described in more detail below, could be used to account for these long-term variations.

A similar question arises in the word selection controller design. The concept of selecting words with the highest expected learning gain from a limited word database might not be directly transferable to a long-term application of the software. The incorporation of a structured organization of the learning material would be essential. Moreover, the information provided by the spelling knowledge representation enables the design of specialized training sessions focusing on specific spelling difficulties of a student.

The evaluation of the learning progress by means of learning curves provided a comparison of different software versions or subgroups on individual error categories. A further subdivision of the error categories have been avoided due to the limited amount of committed errors per child in each category.

However, it would be interesting to explore further approaches to measure progress not only based on absolute error counts, but with respect to the relative change in individual spelling difficulties of a child. These thoughts lead to the questions of how else progress could be defined and how it still can be compared between different subjects. Additionally, findings of the learning progress evaluation based on error behavior in the training software do not allow to directly draw conclusions about improvement in every day writing. In the learning curve analysis we jointly measure learning in spelling as well as the increase in utilizing the presented learning aids. However, also the prevention of error entries by the use of learning aids and the consequential reduction of visualizing erroneously spelled words is desired and beneficial for the process of learning.

The presented model of engagement was inferred from student input data. We identified *Focused* and *Receptive* states on the basis of input and error behaviors alone. However, the lack of ground truth inhibits an evaluation of the developed model. Additional assessments of affect of a different nature, such as sensor, camera or questionnaire data, would on the one hand enable a verification of the labeling of the latent variables. On the other hand, it would allow to embed the model of engagement into an more rich affective context and to relate the identified states to the underlying fundamental affective dimensions (e.g., boredom, flow, confusion and frustration) of a student.

In this thesis we first presented a knowledge representation based on the assumption of relatively stable spelling knowledge; second, we investigated the long-term progress in spelling by means of learning curves based on the assumption that the observed performance of a student remains constant during a training session; finally, we modeled the short-term variation by means of a model of engagement. One of the key goals of further work is the incorporation of these three components into one student model. The representation of the student's spelling knowledge should account for the process of learning as well as for short-term variations in affective states. It is subject of further work to investigate models, which allow for a joint representation of these factors without being overly complex and too slow in convergence.

APPENDIX A

Behavioral Test Data

On the following pages we give the results of the psychological testings. Table A.1 lists the data of the first, Table A.2 the data of the second user study. The individual tests are described in more detail in Section 4.3.

	Dyslexic		**Non-dyslexic**		**Mann-Whitney**
Measures	**Mean**	**S.D.**	**Mean**	**S.D.**	**p**
Age (years)	10.36	0.87	10.36	0.81	0.922
School grade	3.96	0.84	3.88	0.89	0.726
IQ	106.04	12.26	112.94	10.22	0.062
Verbal IQ	108.04	12.32	115.13	9.71	0.034
Performance IQ	102.93	12.52	107.38	12.48	0.164
Wordlist reading error	-2.68	2.83	-0.38	1.11	1e-04
Wordlist reading time	-4.23	5.43	-0.34	1.09	1e-05
Text reading error	-2.93	3.48	-0.44	1.58	2e-04
Text reading time	-4.42	5.13	-0.18	0.67	1e-06
Writing performance	-1.20	0.67	0.19	1.05	4e-05
	Frequency		**Frequency**		
Gender (m/f)	10/18		13/13		
Handedness (r/l)	24/4		20/6		

Table A.1: *Test results for subjects of first study.*

	Dyslexic		Non-dyslexic		Mann-Whitney
Measures	**Mean**	**S.D.**	**Mean**	**S.D.**	**p**
Age (years)	10.89	0.94	10.29	1.00	0.535
Grade of school	4.68	0.85	4.32	0.90	0.168
IQ	113.03	10.99	117.92	12.31	0.236
Verbal IQ	114.34	16.07	122.80	12.59	0.046
Performance IQ	105.89	17.81	109.64	14.17	0.453
Wordlist reading error	-1.59	1.27	0.11	1.45	3e-05
Wordlist reading time	-1.96	0.99	0.06	1.31	3e-07
Text reading error	-1.81	0.99	0.06	0.83	1e-08
Text reading time	-1.88	0.99	-0.17	0.84	1e-07
Reading word similar: Pseudowords time	-1.02	0.80	0.23	0.76	2e-15
Reading word dissimilar: Pseudowords time	-0.87	0.86	0.29	0.95	2e-08
Spelling performance	-1.48	0.56	-0.16	0.68	1e-09
	Frequency		**Frequency**		
Gender (m/f)	27/10		13/12		
Handedness (r/l)	30/7		23/2		

Table A.2: *Test results for subjects of second study.*

APPENDIX B

Phoneme-Grapheme Correspondence

In this appendix we provide the German specific phoneme-grapheme correspondences found in the 1500 words employed in the user studies. The phonemes are structured according to their affiliation in the phoneme tree (see Section 5.4). Each phoneme-grapheme pair builds a so-called graphoneme.

Hierarchy			Phoneme	Grapheme
vowel (similar)			/a/	*a*
			/aː/	*a, aa, ah*
			/~aː/	*ant*
			/a_i/	*ai, ei, eih*
			/a_u/	*au*
			/e/	*e*
			/eː/	*a, e, ee, eh*
			/ɛ/	*ä, e*
			/ɛː/	*ä, äh*
			/ə/	*e*
			/i/	*i, ie, y*
			/iː/	*i, ie, ieh, ih*
			/ˆi/	*i*
			/ɪ/	*i, ie*
			/o/	*o, au*
			/oː/	*o, oh, oo*
			/ɔ/	*o*
			/ɔ_y/	*äu, eu, oi*
			/ʌː/	*ö, öh*
			/ɘ/	*ö*
			/u/	*u, ou*
			/uː/	*u, uh*
			/ˆu/	*u*
			/ʊ/	*u, ou*
			/y/	*ü, y*
			/yː/	*ü, üh, y*
			/ʏ/	*ü, y*
consonant	obstruent	fricative voiced	/j/	*j*
			/v/	*v, w*
			/z/	*s*
			/ʒ/	*g, j*
		fricative unv.	/f/	*f, ff, ph, v*
			/ç/	*ch, g*
			/x/	*ch*
			/s/	*s, ss*
			/ʃ/	*ch, s, sch, sk*
		plosive voiced	/g/	*g*
			/b/	*b, bb*
			/d/	*d*
		plosive unv.	/k/	*c, ch, ck, g, k*
			/p/	*b, p, pp*
			/t/	*d, dt, t, tt, th*
		affricated	/t_s/	*t, ts, tz, z*
			/k_v/	*qu*
			/k_s/	*ch, x*
			/p_f/	*pf*
			/h/	*h*
	sonor	nasal	/m/	*m, mm*
			/n/	*n, nn*
			/ŋ/	*n, ng*
		liquid	/l/	*l, ll*
			/ɒ/	*er*
			/ˆɒ/	*r, rr*
			/r/	*r, rh, rr*

Table B.1: *All graphonemes found in the* 1500 *words listed according to their affiliation in the phoneme tree. The phonemes are depicted in the ETHPA notation.*

APPENDIX C

Mal-Rules

Table C.1 depicts the significance of individual mal-rules. Since not all students show difficulties with every mal-rule, we investigate the highest likelihood ration (LR) score across all students of the first study. The **p** value denotes the significance for the student with the highest LR. To determine whether the mal-rule is significant at a given α-level for at least one of the students, we have to apply a false discovery rate correction. The **Sig.** illustrates the significance for the false discovery rate corrected $\alpha = 5\%$ (*), 1% (**), and 0.1% (***).

Error category	Mal-rule	p	Sig.
Typo	*KD(Left/Right)*	2e-16	***
	KD(Top/Bottom)	0.012	
	Technical	2e-16	***
Capitalization	*ToLowerCase*	2e-16	***
	ToUpperCase	2e-16	***
Dyslexic confusion	*VD(LowerCase)*	0.013	
	VD(UpperCase)	0.018	
	VD(L/UCase)	8e-08	***
	AD(Vowel)	0.018	
	AD(Similar)	5e-12	***
	AD(Consonant)	2e-04	**
	AD(Obstruent)	4e-06	***
	AD(Fricative)	0.005	
	AD(FrVoiced)	0.002	
	AD(FrUnvoiced)	0.002	
	AD(Affricative)	1e-06	***
	AD(Plosive)	4e-08	***
	AD(PlVoiced)	4e-04	*
	AD(PlUnvoiced)	3e-04	*
	AD(Sonor)	5e-07	***
	AD(Nasal)	7e-08	***
	AD(Fluid)	0.024	
Phoneme omission	*PhonemeOmission*	2e-16	***
Phoneme-grapheme matching	*PM(VowMain)*	3e-06	***
	PM(VowSpec)	2e-16	***
	PM(ConsMain)	2e-16	***
	PM(ConsSpec)	2e-16	***
	El(Omission)	2e-16	***
	El(Addition)	4e-14	***
	Sh(Omission)	2e-16	***
	Sh(Addition)	2e-16	***

Table C.1: *Significance of individual mal-rules evaluated by means of a likelihood ratio test. Significance code according to false discovery rate corrected* $\alpha = 5\%$ *(*), 1% (**), and 0.1% (***).*

Curriculum Vitae

Gian-Marco Baschera

Personal Data

April 22, 1982	Born in Tulsa, Oklahoma (USA)
Nationality	Swiss / Italian

Education

May 2007 –	Research assistant and Ph. D. student at the Computer Graphics Laboratory of the Swiss Federal Institute of Technology (ETH) Zurich, Prof. Markus Gross.
Oct. 2006	Diploma degree in Computational Science and Engineering.
Oct. 2003 – Oct. 2006	Diploma Studies of Computational Science and Engineering, ETH Zurich, Switzerland. Specialization: Robotics.
Oct. 2001 – Oct. 2003	Basic Studies of Mathematics, ETH Zurich, Switzerland.

Awards

Oct. 2006	Willi-Studer-Award for "Best Diploma"
Jul. 2011	Best Student Paper Award "Modeling Engagement Dynamics in Spelling Learning" at AIED 2011.

Scientific Publications

G.M. BASCHERA and M. GROSS. A Phoneme-Based Student Model for Adaptive Spelling Training. *Proceedings of the 14th International Conference on Artificial Intelligence in Education*, Brighton, UK, 2009.

A. MÜLLER, G. CANDRIAN, J. KROPOTOV, V. PONOMAREV, and G.M. BASCHERA. Classification of ADHD patients on the basis of independent ERP components using a machine learning system. *Nonlinear Biomedical Physics*, 2010.

G.M. BASCHERA and M. GROSS. Dybuster - Ein adaptives, multi-modales Therapiespiel für Legastheniker. *Spielend Lernen*, Rostock, DE, 2010.

G.M. BASCHERA and M. GROSS. Poisson-Based Inference for Perturbation Models in Adaptive Spelling Training. *International Journal of Artificial Intelligence in Education*, 2010.

M. KAST, G.M. BASCHERA, M. GROSS, L. JÄNCKE and M. MEYER. Computer-based Learning of Spelling Skills in Children with and without Dyslexia. *Annals of Dyslexia*, 2011.

G.M. BASCHERA, A.G. BUSETTO, S. KLINGLER, J.M. BUHMANN and M. GROSS. Modeling Engagement Dynamics in Spelling Learning. *Proceedings of the 15th International Conference on Artificial Intelligence in Education*, Auckland, NZ, 2011.

A. MÜLLER, G. CANDRIAN, V.A. GRANE, J. KROPOTOV, V. PONOMAREV, and G.M. BASCHERA. Discriminating between ADHD adults and controls using independent ERP components and a support vector machine: a validation study. *Nonlinear Biomedical Physics*, 2011.

Employment

May 2007 – May 2011	Research assistant at ETH Zurich, Zurich, Switzerland.
Jul. 2000 – Aug. 2000	Internship at Hilti AG in Web Technology development, Buchs (Switzerland).
Jul. 1999 – Aug. 1999	Internship at Hilti AG in e-Business development, Tulsa (USA).

Bibliography

[ABCL90] J. R. Anderson, A. T. Boyle, A. Corbett, and M. Lewis. Cognitive modeling and intelligent tutoring. *Artificial Intelligence*, 42:7–51, 1990.

[Abd07] H. Abdi. Bonferroni and sidak corrections for multiple comparisons. In N. J. Salkind, editor, *Encyclopedia of Measurement and Statistics*. Thousand Oaks, CA, 2007.

[AH02] M. Ahissar and S. Hochstein. The role of attention in learning simple visual tasks. In *Perceptual learning*, pages 253–272. MIT Press, 2002.

[AHD00] The american heritage dictionary of the english language. Houghton Mifflin, 2000.

[Ano07] A. Anohina. Advances in intelligent tutoring systems: Problem-solving modes and model of hints. *International Journal of Computer, Communications & Control*, 2(1):48–55, 2007.

[AT63] F. J. Anscombe and J. W. Tukey. The examination and analysis of residuals. *Technometrics*, 5(2):141–160, 1963.

[Atk06] K. Atkinson. Gnu aspell 0.60.4. `http://aspell.sourceforge.net/`, 2006.

[Aug85] G. Augst. Kommentar zum internationalen vorschlag der gross- und kleinschreibung. *Kommission für Rechtschreibfragen: Die Rechtschreibung des Deutschen und ihre Neuregelung*, pages 114–142, 1985.

[AW05] I. Arroyo and B. Woolf. Inferring learning and attitudes from a bayesian network of log file data. In *Proceedings of Artificial Intelligence in Education*, pages 33–40, 2005.

[AWRT09] I. Arroyo, B. P. Woolf, J. M. Royer, and M. Tai. Affective gendered learning companions. In *Proceeding of Artificial Intelligence in Education*, pages 41–48. IOS Press, 2009.

[BAZ+99] V. W. Berninger, R. D. Abbott, D. Zook, S. Ogier, Z. Lemos-Britton, and R. Brooksher. Early intervention for reading disabilities. *Journal of Learning Disabilities*, 32(6):491–503, 1999.

[BB83] L. Bradley and P. E. Bryant. Categorizing sounds and learning to read-a causal connection. *Nature*, 301:419–421, 1983.

[BB07] M. Bodén and M. Bodén. Evolving spelling exercises to suit individual student needs. *Applied Soft Computing*, 7(1):126–135, 2007.

[BBA76] A. Barr, M. Beard, and R. C. Atkinson. The computer as a tutorial laboratory: The stanford bip project. *International Journal of Man-Machine Studies*, 8:567–596, 1976.

[BC89] D. Boles and J. Clifford. An upper- and lowercase alphabetic similarity matrix, with derived generation similarity values. *Behavior Research Methods*, 21(6):579–586, 1989.

[BCK04] R. S. Baker, A. T. Corbett, and K. R. Koedinger. Detecting student misuse of intelligent tutoring systems. In *Proceedings of Intelligent Tutoring Systems*, pages 531–540, 2004.

[Bec05] J. E. Beck. Engagement tracing: Using response times to model student disengagement. In *Proceedings of Artificial Intelligence in Education*, pages 88–95. IOS Press, 2005.

[Bis06] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[Blo88] F. J. Blommaert. Early-visual factors in letter confusions. *Spatial Vision*, 3(3):199–224, 1988.

[BNP07] A. Bader-Natal and J. Pollack. Evaluating problem difficulty rankings using sparse student response data. In *Supplementary Proceedings of Artificial Intelligence in Education*. IOS Press, 2007.

[BOB09] A. G. Busetto, C. S. Ong, and J. M. Buhmann. Optimized expected information gain for nonlinear dynamical systems. In *Proceedings of International Conference on Machine Learning*, pages 97–104, 2009.

[BPC01] D. A. Balota, M. Pilotti, and M. J. Cortese. Subjective frequency estimates for 2'938 monosyllabic words. *Memory & Cognition*, 29(4):639–647, 2001.

[BPvR93] R. H. Baayen, R. Piepenbrock, and H. van Rijn. The celex lexical database. Philadelphia, PA, University of Pennsylvania, Linguistic Data Consortium, 1993.

[Bro90] A. S. Brown. A review of recent research on spelling. *Educational Psychology Review*, 2(4):365–397, 1990.

[BRW+99] T. Baldeweg, A. Richardson, S. Watkins, C. Foale, and J. Gruzelier. Impaired auditory frequency discrimination in dyslexia detected with mismatch evoked potentials. *Annals of neurology*, 45(4):495–503, 1999.

[Bur82] R. R. Burton. Diagnosing bugs in a simple procedural skill. In D. H. Sleeman and J. S. Brown, editors, *Intelligent Tutoring Systems*, pages 157–184. Academic Press, London, 1982.

[CA95] A. T. Corbet and J. R. Anderson. Knowledge tracing: Modelling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4:253–278, 1995.

[CCAH93] M. Coltheart, B. Curtis, P. Atkins, and M. Haller. Models of reading aloud: Dual-route and parallel-distributed-processing approaches. *Psychological Review*, 100(4):589–608, 1993.

[Cle79] W.S. Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368):829–836, 1979.

[CMA+10] D. G. Cooper, K. Muldner, I. Arroyo, B. P. Woolf, and W. Burleson. Ranking feature sets for emotion models used in classroom based intelligent tutoring systems. In *Proceedings of User Modeling, Adaptation and Personalization*, pages 135–146, 2010.

[Con02] C. Conati. Probabilistic assessment of user's emotions in educational games. *Applied Artificial Intelligence*, 16:555–575, 2002.

[CT98] A. C. Cameron and P. K. Trivedi. *Regression Analysis of Count Data*. Cambridge University Press, 1998.

[Dam64] F. J. Damerau. A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7(3):171–176, 1964.

[Dav85] R. D. Davis. *Anatomy of a Learning Disability*. Davis Dyslexia Association International, 1985.

[DKR99] E. Deci, R. Koestner, and R. Rayn. A meta-analytic review of experiments examining the effects of extrinsic rewards on intrinsic motivation. *Psychological Bulletin*, 125(6):627–668, 1999.

[DKS05] O. Dekel, J. Keshet, and Y. Singer. An online algorithm for hierarchical phoneme classification. *Lecture Notes in Computer Science*, 3361:146–158, 2005.

[dlTD04] J. de la Torre and J. A. Douglas. Higher-order latent trait models for cognitive diagnosis. *Psychometrika*, 69(3):333–353, 2004.

[Dre75] E. A. Drewe. Go-nogo learning after frontal lobe lesions in humans. *Cortex*, 11:8–16, 1975.

[dVP02] A. de Vicente and H. Pain. Informing the detection of the students' motivational state: an empirical study. In S. A. Cerri, G. Gouarderes, and F. Paraguacu, editors, *Proceedings of Intelligent Tutoring Systems*, pages 933–943, 2002.

[ECI94] (ECI/MCI) European Corpus Initiative Multilingual Corpus I. Website. `www.elsnet.org/resources/eciCorpus.html`, 1994.

[Ell02] N. C. Ellis. Frequency effects in language processing. *Studies in Second Language Acquisition*, 24:143–188, 2002.

[EMBG09] J. Ecalle, A. Magnan, H. Bouchafa, and J. E. Gombert. Computer-based training with ortho-phonological units in dyslexic children: new investigations. *Dyslexia*, 15(3):218–238, 2009.

[Eve92] B. Everitt. *The analysis of contingency tables*. Chapman & Hall, 1992.

[Fis73] G. H. Fischer. The linear logistic test model as an instrument in educational research. *Acta Psychologica*, 37:359–374, 1973.

[Fri85] U. Frith. Beneath the surface of developmental dyslexia surface dyslexia. In K. E. Patterson, J. C. Marshall, and M. Coltheart, editors, *Neurophysiological and Cognitive Studies of Phonological Reading*, pages 301–327. Erlbaum, London, 1985.

[GAWA06] S. E. Gathercole, T. P. Alloway, C. Willis, and A. M. Adams. Working memory in children with reading disabilities. *Journal of Experimental Child Psychology*, 93(3):265–281, 2006.

[GHNW95] M. Grund, G. Haug, C. L. Naumann, and B. V. Weinheim. *Diagnostischer Rechtschreibtest 5. Klasse*. Beltz-Verlag, Weinheim, 1995.

[GKG08] R. M. C. Garcia, C. D. Kloos, and M. C. Gil. Game based spelling learning. In *Frontiers in Education Conference*, pages S3B–11 – S3B–15, 2008.

[GV07] M. Gross and C. Vögeli. A multimedia framework for effective language training. *Computers & Graphics*, 31(5):761–777, 2007.

[GvdM87] L. Györfi and E. C. van der Meulen. Density-free convergence properties of various estimators of entropy. *Computational Statistics & Data Analysis*, 5(4):425–436, 1987.

[Hal00] T. A. Hall. *Phonologie: Eine Einführung*. Gruyter, 2000.

[HBM00] A. Heathcote, S. Brown, and J. D. Mewhort. The power law repealed: the case for an exponential law of practice. *Psychonomic Bulletin & Review*, 7(2):185–207, 2000.

[Her90] H. Hermansky. Perceptual linear predictive (plp) analysis of speech. *Journal of the Acoustical Society of America*, 87(4):1738–1752, 1990.

[HF09] A. Heray and C. Frasson. Predicting learner answers correctness through brainwaves assessment and emotional dimensions. In *Proceedings of Artificial Intelligence in Education*, pages 49–56, 2009.

[Hoc60] C. F. Hockett. The origin of speech. *Scientific American*, 203:88–96, 1960.

[HR06] M. Hilte and P. Reitsma. Spelling pronunciation and visual preview both facilitate learning to spell irregular words. *Annals of Dyslexia*, 56(2):301–318, 2006.

[ISW] Voelker Software. ispellwell. `http://www.ispellwell.com/`.

[Jam98] C. James. *Errors in Language Learning and Use: Exploring Error Analysis*. Harlow, Pearson Education, 1998.

[JB80] C. M. Jarque and A. K. Bera. Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economics Letters*, 6(3):255–259, 1980.

[JD04] A. James and E. Draffan. The accuracy of electronic spell checkers for dyslexic learners. *PATOSS bulletin*, August 2004.

[JW06] J. Johns and B. Woolf. A dynamic mixture model to detect student motivation and proficiency. In *Proceedings of User Modeling, Adaptation and Personalization*, pages 163–168, 2006.

[Kal97] O. Kallenberg. *Foundations of Modern Probability*. Springer-Verlag, New York, 1997.

[Kas11] M. Kast. *Neurocognition of Developmental Dyslexia: The Role of Multisensory Processing During Reading and Spelling*. PhD thesis, University of Zürich, 2011.

[KKC+01] T. Kujala, K. Karma, R. Ceponiene, S. Belitz, P. Turkkila, M. Tervaniemi, and R. Naatanen. Plastic neural changes and reading improvement caused by audiovisual training in reading-impaired children. *Proceedings of the National Academy of Sciences of the USA*, 98(18):10509–10514, 2001.

[KL51] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[KMV+07] M. Kast, M. Meyer, C. Vogeli, M. Gross, and L. Jancke. Computer-based multisensory learning in children with developmental dyslexia. *Restorative Neurology and Neuroscience*, 25(3-4):355–369, 2007.

[Kon03] G. Kondrak. Phonetic alignment and similarity. *Computers and the Humanities*, 37:273–291, 2003.

[KRP01] B. Kort, R. Reilly, and R. W. Picard. An affective model of interplay between emotions and learning: Reengineering educational pedagogy - building a learning companion. *Advanced Learning Technologies*, pages 43–46, 2001.

[Lez95] M. Lezak. *Neuropsychological Assessment*. Oxford University Press, New York, 1995.

[LG00] M. Linder and H. Grissemann. *Zürcher Lesetest*. Hogrefe-Verlag, Bern-Göttingen, 2000.

[LH88] M. Livingstone and D. Hubel. Segregation of form, color, movement, and depth: anatomy, physiology, and perception. *Science*, 240(4853):740–749, 1988.

[LHE99] S. Lux, C. Helmstaedter, and C. E. Elger. Normative study on the "verbaler lern- und merkfähigkeitstest" (vlmt). *Diagnostica*, 45(4):205–211, 1999.

[LKX+09] L. Liu, S. A. Klein, F. Xue, J. Y. Zhang, and C. Yu. Using geometric moments to explain human letter recognition near the acuity limit. *Journal of Vision*, 9(1), 2009.

[LM05] S. Lehmann and M. M. Murray. The role of multisensory memories in unisensory object discrimination. *Brain Research. Cognitive Brain Research*, 24(2):326–334, 2005.

[LWGN90] W. J. Lovegrove, J. William, R. P. Garzia, and S. B. Nicholson. Experimental evidence for a transient system deficit in specific reading disability. *Journal of the American Optometric Association*, 61(2):137–146, 1990.

[LWM97] K. Landerl, H. Wimmer, and E. Moser. *Der Salzburger Lese- und Rechtschreibtest (SLRT)*. Verlag Hans Huber, Bern, 1997.

[Mac99] C. A. MacArthur. Overcoming barriers to writing: Computer support for basic writing skills. *Reading & Writing Quarterly*, 15:169–192, 1999.

[Mar99] E. Maris. Estimating multiple classification latent class models. *Psychometrika*, 64:187–212, 1999.

[Mis96] R. J. Mislevy. Test theory reconceived. *Journal of Educational Measurement*, 33(4):379–416, 1996.

[MK09] P. Miller and A. Kupfermann. The role of visual and phonological representations in the processing of written words by readers with diagnosed dyslexia: evidence from a working memory task. *Annals of Dyslexia*, 59:12–33, 2009.

[MN89] P. McCullagh and J. A. Nelder. *Generalized Linear Models*. Chapman and Hall, London, 1989.

[Mur01] K. Murphy. The bayes net toolbox for matlab. *Computing Science and Statistics*, 33, 2001.

[NF90] R. I. Nicolson and A. J. Fawcett. Automaticity: a new framework for dyslexia research? *Cognition*, 35(2):159–182, 1990.

[NM65] J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.

[NR81] A. Newell and P. S. Rosenbloom. Mechanisms of skill acquisition and the law of practice. In J. R. Anderson, editor, *Cognitive skills and their acquisition*, pages 1–55. Hillsdale, NJ, 1981.

[OL08] K. Oberauer and S. Lewandowsky. Forgetting in immediate serial recall: Decay, temporal distinctiveness, or interference? *Psychological Review*, 115:544–576, 2008.

[PB00] J. C. Pinheiro and D. M. Bates. *Mixed Effects Models in S and S-Plus*. Springer, 2000.

[Ped65] P. Pedersen. The mel scale. *Journal of Music Theory*, 9(2):295–308, 1965.

[Pfi05] B. Pfister. Definition of the phone inventories to be used for mixedlingual tts synthesis. Technical report, ETH Zuerich, 2005.

[PP87] M. I. Posner and D. E. Presti. Selective attention and cognitive control. *Trends in Neuroscience*, 10(1):13–17, 1987.

[PR87] M. I. Posner and R. D. Rafal. Cognitive theories of attention and the rehabilitation of attentional deficits. In R. J. Meier, A. C. Benton, and L. Diller, editors, *Neuropsychological Rehabilitation*. Churchill Livingstone, Edinburgh, 1987.

[PZ84] J. J. Pollock and A. Zamora. Automatic spelling correction in scientific and scholarly text. *Communications of the ACM*, 27(4):358–368, 1984.

[Ram03] F. Ramus. Developmental dyslexia: specific phonological deficit or general sensorimotor dysfunction? *Current Opinion in Neurobiology*, 13(2):212–218, 2003.

[RCC08] A. M. Re, M. Caeran, and C. Cornoldi. Improving expressive writing skills of children rated for adhd symptoms. *Journal of Learning Disabilities*, 41(6):535–544, 2008.

[Rei89] P. Reitsma. Orthographic memory and learning to read. In P. G. Aaron and R. M. Joshi, editors, *Reading and writing disorders in different orthographic systems*, volume 52, pages 51–74. Kluwer Academic Publishers, Dordrecht, 1989.

[RRD+03] F. Ramus, S. Rosen, S. C. Dakin, B. L. Day, J. M. Castellote, S. White, and U. Frith. Theories of developmental dyslexia: insights from a multiple case study of dyslexic adults. *Brain*, 126(4):841–865, 2003.

[SE97] R. Spooner and A. D. N. Edwards. User modelling for error recovery: A spelling checker for dyslexic users. In A. Jameson, C. Paris, and C. Tasso, editors, *Proceedings of the Sixth International Conference on User Modeling*, pages 147–157. Springer, 1997.

[SG64] A. Savitzky and M. J. E. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 8(36):1627–1639, 1964.

[Sha49] C. E. Shannon. *The Mathematical Theory of Information*. University of Illinois Press, 1949.

[SKD+04] G. Schulte-Korne, W. Deimel, J. Bartling, and H. Remschmidt. Neurophysiological correlates of word recognition in dyslexia. *Journal of Neural Transmission*, 111(7):971–984, 2004.

[SP96] V. J. Shute and J. Psotka. Intelligent tutoring systems: Past, present, and future. In D. H. Jonassen, editor, *Handbook of Research for Ed-*

ucational Communications and Technology, pages 570–600. Simon & Schuster Macmillan, New York, 1996.

[Spe07] K. Spencer. Predicting children's word-spelling difficulty for common english words from measures of orthographic transparency, phonemic and graphemic length and word frequency. *The British Psychological Society*, 98:305–338, 2007.

[SS] 4Mation Educational Resources Ltd. Superspell 2. `http://www.4mation.co.uk/cat/superspell.html`.

[SS08] L. Shams and A. R. Seitz. Benefits of multisensory learning. *Trends in Cognitive Science*, 12(11):411–417, 2008.

[Tal80] P. Tallal. Auditory temporal perception, phonics, and reading disabilities in children. *Brain and language*, 9(2):182–198, 1980.

[Tat85] K. K. Tatsuoka. A probabilistic model for diagnosing misconceptions by the pattern classification approach. *Journal of Educational Statistics*, 10(1):55–73, 1985.

[TR99] U. Tewes and P. U. S. Rossmann. *Hamburg-Wechsler-Intelligenztest für Kinder*. Verlag Hans-Huber, Bern, 1999.

[US] eReflect Learning Solutions. Ultimate spelling. `http://www.ultimatespelling.com/`.

[Ver88] J. Veronis. Computerised correction of phonographic errors. *Computers and Humanities*, 22(1):43–56, 1988.

[WF74] R. A. Wagner and M. J. Fischer. The string-to-string correction problem. *Journal of the Asscociation for Computing Machinery*, 21:168–173, 1974.

[WHO93] World Health Organization. Classification of mental and behavioural disorders. ICD-10. The international classification of diseases, Vol. 10, Geneva, 1993.

[ZGF02] P. Zimmermann, M. Gondan, and B. Fimm. Kitap testbatterie zur aufmerksamkeitsprüfung für kinder. Psytest, Vera Fimm, Herzogenrath, 2002.

[ZZS01] F. Zheng, G. Zhang, and Z. Song. Comparison of different implementations of mfcc. *Journal of Computer Science & Technology*, 16(6):582–589, 2001.