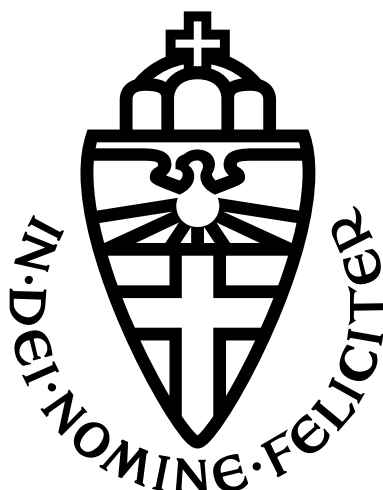


RADBOD UNIVERSITY NIJMEGEN



Automatic Quantitative Analysis of Spelling Errors Made by Dutch Elementary School Children

A BASIScript CORPUS STUDY

MASTER THESIS IN ARTIFICIAL INTELLIGENCE

Author:

Wieke Noa HARMSSEN
s4590996

Supervisor:

Dr. Helmer STRIK^{1,2,3}

Second Assessor:

Dr. Iris HENDRICKX¹

¹ Centre for Language Studies

² Centre for Language and Speech Technology

³ Donders Institute for Brain, Cognition and Behaviour
Radboud University

2 July 2021

PREFACE

Dear reader,

After exactly a year of hard work, I present to you proudly my master thesis titled “Automatic quantitative analysis of spelling errors made by elementary school children: A BasiScript corpus study”. This thesis is the end product of my research project in which I developed an algorithm that automatically detects spelling errors and annotates them with the spelling principle that is violated. Application of this algorithm to the BasiScript corpus, which consists of more than eighty thousand digitized handwritten texts by Dutch children, makes automatic quantitative research to Dutch spelling errors for the first time possible.

The topic of this thesis emerged from work that I did during my internship at the Centre for Language and Speech Technology. Here, I detected reading errors automatically, by aligning what a subject should read and actually read. From here, it is a small step to detecting spelling errors by aligning the original version of a written text with its corrected version. I am happy that in this thesis I was able to apply the knowledge of artificial intelligence, that I acquired during my studies, in the field of language and education, where I am interested in.

In the first place, I wrote this thesis to meet the graduation requirements of my master’s degree in Artificial Intelligence. However, in retrospect, this thesis brought me so much more than only my diploma. Putting into practice all the knowledge and skills that I learned during my studies, made me realize that I am capable of doing much more than I ever thought. In this way, problems that seem unsolvable in the beginning, turned out to be solvable in the end. In addition, I wrote this thesis in a difficult time, during the COVID-19 pandemic. On one hand, writing a thesis is a good activity while you have to stay at home because of intelligent, partial and full lockdowns. On the other hand, life outside my studies became quite boring and lonely. Happily, I have really great parents, brothers, a sister, friends and housemates that supported me during this crisis. I would like to thank them a lot, without them writing this thesis would have been much more difficult.

In addition, I would also like to thank my supervisor, Helmer Strik, for guiding me in my research. He gave me the space to carry out this research project autonomously and in my own way, but also gave help quickly when I asked for it. Furthermore, he offered me the opportunity to publish my research at conferences. This resulted in a paper for the EduLearn21 conference and a poster for the CLIN31 meeting. Writing this paper and poster were great experiences from which I learned a lot. Catia Cucciarini helped me with this writing process and I would like to thank her for that. In addition, I also got the opportunity to discuss my research with two spelling experts, Martine Gijssel and Ronja Laarmann-Quante. I would like to thank them for making time to meet me and answering my questions. Last, but definitely not least, I would like to thank Iris Hendrickx for being my second assessor.

I hope you enjoy reading this thesis!

Wieke Noa Harmsen

Nijmegen, July 2nd 2021

CONTENTS

1	Abstract	4
2	Introduction	5
3	Background	7
3.1	The spelling process	7
3.1.1	Three important competences for spelling in Dutch	7
3.1.2	The dual route model for spelling	8
3.1.3	Spelling is more difficult than reading	8
3.2	Methodologies of earlier research to spelling errors	9
3.2.1	Three different types of annotation schemes	9
3.2.2	Manual annotation of Dutch spelling errors	10
3.2.3	Automatic annotation of German spelling errors	10
3.3	Findings from earlier research to Dutch spelling errors	11
3.3.1	Gender	12
3.3.2	Spelling development	12
4	Methodology	14
4.1	The data: texts from the BasiScript corpus	14
4.2	The spelling error detection and annotation algorithm	14
4.2.1	Preprocessing of BasiScript texts	15
4.2.2	Detection of PCUs and spelling errors	18
4.2.3	Annotation of PCUs with spelling principles	19
4.3	Evaluation of the algorithm	22
4.4	Visualization of spelling errors in the BasiScript corpus	22
4.4.1	Normalized Spelling Error Frequency	22
4.4.2	Relative Spelling Error Frequency	23
5	Results	24
5.1	Description of the analysed data	24
5.1.1	Analysed texts	24
5.1.2	Analysed words and PCUs	24
5.2	Evaluation of the algorithm	25
5.2.1	PCU segmentation and alignment	25
5.2.2	PCU annotation	26
5.3	Exploration of the Data	26
5.3.1	Gender	26
5.3.2	Homophone errors	27
5.3.3	Most frequently violated spelling principles	27
5.3.4	Most problematic spelling principles	30

6 Discussion	32
6.1 The algorithm	32
6.1.1 Limitations to the algorithm design	33
6.1.2 Limitations to the algorithm evaluation method	33
6.2 Gender	34
6.3 Children’s spelling development	34
6.3.1 Number of spelling errors	34
6.3.2 Homophone errors	34
6.3.3 Most frequently violated spelling principles	34
6.3.4 Most problematic spelling principles	35
6.4 Possible future directions	36
6.4.1 Further analysis of the BasiScript corpus	36
6.4.2 Improvement and extension of the algorithm	36
6.4.3 Machine learning applications	37
6.4.4 Other applications of the algorithm	37
7 Conclusion	39
References	40
Appendices	42
A Computer phonetic alphabet CGN2	42
B Example of multilayered annotated word	44
C Technical overview of the algorithm	45
D Phoneme-grapheme alignment rules	46
E Annotation scheme	47
F Precision scores	52

1 ABSTRACT

Recent findings about the development of writing skills by Dutch children in primary school show an alarming decrease in spelling proficiency in 2019 with respect to 2009 (Inspectie van Onderwijs, 2021). These results raise concerns about the quality of spelling education and call for innovative solutions. Insights into which spelling errors are most common and problematic would help design such solutions, but so far this type of quantitative research has not been conducted.

A recently realized corpus of handwritten texts by elementary school children, *BasiScript* (Tellings et al., 2018a), makes this kind of innovative, quantitative research now possible. However, manual analysis of spelling errors in this corpus is really time-consuming and difficult. Therefore, the first aim of the present study is to develop an algorithm that automatically detects spelling errors and annotates them with the spelling principles that are violated. Spelling principles are rules that children need to master to write Dutch texts flawless. Subsequently, the algorithm is employed to the texts from the *BasiScript* corpus to find out whether boys or girls make less errors and to gain insight into the development of children’s spelling performance between grade two and six of elementary school.

The *BasiScript* corpus contains two digitized versions of each handwritten text: the original text (a typed version of what the child actually wrote) and the target text (a corrected version of the original text, without spelling errors). The algorithm first aligns these two text versions and splits them into words. After that, each target word is annotated with multiple layers of information, like the phonetic transcription, lemma, morphemes, and POS-tags. Using the phonetic transcription, the graphemes of each word are grouped into Phoneme-Corresponding Units (PCUs). A PCU is a sequence of graphemes that corresponds to one phoneme (Laarmann-Quante, 2016). Spelling errors are detected as PCUs that are deleted, inserted or substituted in an alignment of the target and original PCU segmentations.

The spelling errors are then annotated with the spelling principles that are violated, using a Dutch linguistic annotation scheme. Most spelling principle annotations in this scheme can also be used to annotate correctly spelled PCUs. In this way, it is possible determine how often a spelling principle is applied incorrectly with respect to the total frequency of a spelling principle.

The results reveal that girls made on average less errors than boys. In addition, with increasing grade, the average number of spelling errors decreased, the probability that an incorrectly written word has the same phonetic transcription as the target word increases, spelling principles involving only phoneme-grapheme conversions were most frequently violated, and spelling principles concerning knowledge about Dutch syntax and semantics were most problematic to learn.

In conclusion, the present approach allows quantitative analyses of spelling errors that have so far not been possible. In the future, the spelling errors in *BasiScript* could be explored more and the developed algorithm could be employed in computer applications that provide detailed feedback to written materials. This feedback does not only indicate which letters are written incorrect, but also why they are incorrect.

2 INTRODUCTION

Being able to understand (read) and produce (write) written language are very useful skills. They allow humans among other things to communicate with others that are not at the same place (e.g., using text messages or e-mails), to store information and share it with large groups of people (e.g., in news articles, books or data bases) or to make notes that help with reminding things (e.g., a grocery list or a calendar).

One important aspect of writing is spelling. In an alphabetical language like Dutch, this involves choosing the right letters (graphemes) to represent the speech sounds (phonemes) that a word consists of. Therefore, it is important that before children start learning to read and write, they are already able to speak in that language. Being able to speak and understand spoken language is not an innate skill, but it is learned quite easily by most children from the moment they are born, just by listening to and imitating spoken language. In contrast, reading and spelling are more complex tasks to learn, that require more effort and specific reading and spelling lessons.

Dutch children start learning to read and spell in the first grade of elementary school, when they are around 6-7 years old. Research has shown that spelling lessons should meet three facets to be effective. In the first place, direct instruction of the spelling principles from the Dutch orthographic system is necessary to learn them, because children are not able to discover the underlying spelling principles of the language they are learning to write by themselves (Assink, 1986). In addition, active practice is essential to become a good speller. This means for example that just by reading words and not writing them, it is not possible to develop good spelling skills (van Leerdam et al., 1998). In addition, Cordewener et al. (2016) stated that it is essential to write complete words and not only parts of the words, or single letters. Thirdly, children also need to get feedback on their writing. Harward et al. (1994) showed that immediate feedback on each written word is more effective for developing good spelling skills than delayed feedback, which is given after a complete list of words was written.

When spelling lessons include direct instruction, enough practice and good feedback, most children should be able to develop good spelling skills during elementary school. Unfortunately, a recent report by the Dutch Education Inspection (Inspectie van Onderwijs, 2021) shows that the spelling proficiency of Dutch sixth graders (i.e., the last class of elementary school, 11-12 years old) is currently not good enough. In this report, the spelling proficiency of Dutch children was evaluated by comparing their spelling proficiency with the fundamental and target levels that are defined by the committee Meijerink (Meijerink, 2009). The fundamental levels show which spelling level the children should minimally have and the target level represents the desired level of spelling ability. The results show that in 2019 the reference spelling proficiency levels are not reached: 73% (instead of the expected 75%) of the students reach the fundamental spelling proficiency level and only 28% (instead of the expected 50%) reach also the target level (Inspectie van Onderwijs, 2021). For special elementary education these percentages are even lower (respectively 33% and 9%). In addition, an increase in the number of spelling errors was found in 2019 with respect to 2009.

These alarming findings raise concerns about the quality of spelling lessons in the Netherlands and call for innovative solutions to improve spelling instruction, practice and feedback. Quantitative

research to which spelling errors are most frequent and problematic in texts written by children in each grade of elementary school can help to design these solutions. With the recent publication of the BasiScript corpus (Tellings et al., 2018a), which consists of more than eighty thousand texts that are written by Dutch elementary school children, and the growing possibilities of artificial intelligence and natural language processing techniques to analyse large text corpora efficiently, automatic quantitative research to children's spelling errors can now be conducted for the Dutch language. Therefore, I pose the following research questions in this study:

1. Is it possible to develop an algorithm that is able to automatically detect and annotate spelling errors in the BasiScript corpus?
2. Who makes less spelling errors in each grade of elementary school: boys or girls?
3. How does the spelling performance of elementary school children, measured by the number and type of spelling errors that they make, develop over time?
 - (a) Does the average number of spelling errors that children make decrease between grade two and six of elementary school?
 - (b) Does the percentage of spelling errors that are homophones increase between grade two and six of elementary school?
 - (c) Which spelling principles are most frequently violated in each grade of elementary school?
 - (d) Which spelling principles are most problematic in each grade of elementary school?

3 BACKGROUND

In this chapter, the spelling process and the structure of Dutch orthography are addressed, as well as earlier research to spelling errors in Dutch and German. These topics provide information that is important for the development of the automatic spelling error detection and annotation algorithm. After that, this chapter continues with an overview of findings from earlier qualitative research to Dutch spelling errors. Based on these findings, hypotheses are formulated that describe Dutch children's development of spelling ability with respect to their gender and the type and number of errors that they make.

3.1 THE SPELLING PROCESS

There are different ways in which a language can be written. Dutch uses the Latin script and is written using an alphabet of 26 letters. This alphabet contains both consonants and vowels. A Dutch word consists of several speech sounds (phonemes) and each speech sound can be written using one or a sequence of these letters (graphemes). The process of choosing the right letters to write a word is called spelling.

3.1.1 Three important competences for spelling in Dutch

For being able to spell flawless in Dutch, Schijf (2009) states that the development of three competences is essential: the phonological, orthographic and morphological competence. A competence is a cluster of specific knowledge and skills.

A phonological competent writer is able to divide a word in auditory form into phonemes (speech sounds) and knows how to convert these phonemes into graphemes (letters) by applying phoneme-to-grapheme conversion rules.

Since Dutch is a relatively transparent language (Borgwaldt, Hellwig, & De Groot, 2004), most words can be written correctly by using only these phoneme-to-grapheme conversion rules. However, sometimes, these rules are not sufficient enough to derive the correct spelling. For example, when there are multiple possibilities of (sequences of) graphemes possible to write a certain phoneme, or when a phoneme is silent, but still needs to be written. In these cases, application of autonomous spelling principles is necessary (Nunn, 1998). An example of such a principle is "the rule of vowel singulation". This rule states that a long vowel at the end of a syllable should be written with only one grapheme (e.g., "a" in "praten" (speak)), while normally, long vowels are written using two graphemes (e.g., "aa" in "wraak" (revenge)). The orthographic competence involves knowledge about orthographic patterns and applying these autonomous spelling principles correctly.

The last competence is the morphological competence. The Dutch language is based on a morphological principle. This principle states that related words are always written in the same way. For example, the words "trouwen" (marry), "getrouwd" (married), "trouwdag" (wedding day) and "vertrouwen" (trust) are all related to the word "trouw" (faithful). Therefore, the morpheme "trouw" is always written in the same way in these examples. A writer with good morphological knowledge is able to recognize the morphemes that make up a word and to spell them correctly.

3.1.2 The dual route model for spelling

Many models have been developed that describe the cognitive process that is necessary to obtain the right spelling of a word (e.g. Assink (1983); Ellis (1984); Verhoeven (1985); Barry and Seymour (1988); Nunn (1998)). These studies have in common that their models all have two routes. In the first route, the phonological, orthographic and morphological competence are used subsequently to obtain the right spelling of a word. In the second route, the correct spelling is achieved automatically from long-term memory.

The first, phonological, route for spelling Dutch words is described by Nunn (1998) in the following way. First, the morphological competence is used to split the word that needs to be spelled into "auditory" morphemes. A morpheme is the smallest part of a word that has its own meaning. After that, the phonemes that make up each morpheme are converted into graphemes (phonological competence). The next step is to apply autonomous spelling rules to obtain the right spelling of each morpheme (orthographic competence). Lastly, the spelled morphemes are merged, such that a written word is formed. Obtaining the right spelling of a word using this route makes great demand to the speller's short term memory, because the results of the intermediate steps are all saved there.

The second route is called the lexical route. Nunn (1998) describes this route as follows. At a certain moment, advanced spellers have spelled a morpheme or word so many times, that the complete orthographic representation of this morpheme or word is saved in long-term memory. At that moment, the speller can achieve the spelling of these frequently spelled morphemes directly from long-term memory and using the phonological and orthographic competence is not necessary. For example: the word "hond" (dog) is always written with a "d" at the end, while you hear a /t/ sound. This is caused by an autonomous spelling principle called final devoicing. At a certain moment, spellers know that "hond" is always written with a final "d" and never with a "t". This means that conscious application of the final devoicing spelling principle is not necessary anymore. When this second route is used, the spelling process is automated and the correct spelling is achieved faster. This route is also called the lexical route, because it is comparable with a lexicon in which phonetic and written transcriptions of words are matched.

The model by Nunn (1998) takes as starting point that for spelling a word, both the phonological and lexical route can be used interchangeably while spelling the different morphemes of this word. This is also supported in research by Perry and Ziegler (2004) and Houghton and Zorzi (2003).

3.1.3 Spelling is more difficult than reading

Reading and spelling are two processes that are inextricably connected with each other. For reading, also development of the phonological, orthographic and morphological competences are necessary (Schijf, 2009). In addition, the reading process is also often described using a dual route model, with one route for step-by-step recognition of a word and one route for direct, automatic recognition using long-term memory (Perfetti, 1985).

However, spelling is a more difficult task than reading (Bosman & Van Orden, 1997). One reason for this is described by Borgwaldt (2003) and concerns the phonological competence. This

competence involves the ability to convert phonemes into graphemes (writing) and vice versa (reading). The Dutch language has a high forward consistency, which means that transforming a grapheme to a phoneme is not very hard, because there are not many ways in which a grapheme can be pronounced. So, pronouncing a written word is relatively easy. On the other hand, the backward consistency of Dutch is low. This means that a phoneme can be written using many different combinations of graphemes. For example the phoneme /k/ is written as "ch" in *christen* (christian), "c" in *café* (cafe) and "k" in *kapper* (hairdresser).

A second reason why spelling is harder than reading is described by ? (?). They state that semantic knowledge can be really helpful when reading, but not when writing. For example, when someone needs to read the word "chauffeur" (driver), but does not know whether the initial "ch" should be pronounced as /S/ (as in chef) or /G/ (as in *toch*), the reader can try both pronunciations and choose the one that produces a word that the reader recognizes. For writing a word, this trick does not always work. When someone does not know how to write /b @ r EI t/ (CGN2 alphabet, see appendix A), there are several options where the writer can choose from, like "bereid, bereidt, berijd and berijdt". Semantic knowledge can help to choose between the "ei" and "ij", because both words mean something else when "ei" or "ij" is used (respectively "prepare" and "ride"). Semantic knowledge cannot be used to distinct between writing "d" or "dt" at the end of the word. Both "bereid" and "bereidt" have the same meaning and the right choice between these two spellings can only be made by adapting spelling rules.

3.2 METHODOLOGIES OF EARLIER RESEARCH TO SPELLING ERRORS

3.2.1 Three different types of annotation schemes

To investigate spelling errors, an annotation scheme to annotate the errors is absolutely essential. Horbach-Kleijnen (1992) describes three different types of annotations schemes. The first type are etiological systems (i.a., van der Geest et al. (1978); Dumond (1985)), in which the annotations represent the assumed origin or cause of the error. Secondly, there are systems based on didactic applications (i.a., Kort (1987); Ojemann (1970)). These systems are to a certain extent similar to etiological systems, because they also annotate spelling errors with the assumed origin or cause of the error. Only, the annotations in these systems also contain information about how the error should be corrected.

A disadvantage of the etiological schemes and schemes based on didactic applications, is that they both interpret the errors immediately. For example: when a writer writes "boom" (tree) as "boon" (bean), this is classified as a visual error, because it is assumed that the error is a result of the fact that "m" and "n" are visually similar letters. However, there are also other interpretations possible. The error could for example also be caused by the fact that the writer heard wrongly what was said. In that case, it is an auditory error. So, immediate interpretation of errors is not always possible, and therefore using these schemes can lead to inconsistent or incorrect spelling error annotation.

To overcome this problem, recent studies mainly use the third annotation scheme type: linguistic classification systems (i.a., Horbach-Kleijnen (1997); Schijf (2009)). Here, errors are only described

using Dutch orthographic spelling principles, and not yet interpreted.

3.2.2 Manual annotation of Dutch spelling errors

Previous studies to Dutch spelling errors have in common that they use manual spelling error annotation, which is very time-consuming. These studies differ from each other with respect to the annotation scheme they use, and the number and type of written materials that are analysed.

The study with the largest number of different spelling error annotations was done by Horbach-Kleijnen (1997). She used a mutually exclusive annotation scheme of 49 rules divided over six categories to annotate errors written by 211 students from 12-18 years old. These students all had to write the same dictation, consisting of ten sentences (a total of 117 words). The errors were manually detected and annotated. In another study with more participants, Schijf (2009) asked 689 students (12-13 years old) to write a dictation containing 40 words. She annotated the errors in these words manually using an annotation scheme of five annotation labels. These categories are comparable to the six categories of Horbach-Kleijnen's annotation scheme. Both studies used annotation schemes that are based on descriptive properties of the Dutch orthographic system and are mutually exclusive.

Keuning and Verhoeven (2008) adopted an annotation scheme based on the strategy that needs to be applied to spell a word correctly. They analysed dictations written by 1308 primary school children (8-12 years). The errors in these dictations were manually annotated with one of four categories (i.e., phonetic, analogy, rule-based and visual imprint). These categories were not mutually exclusive.

The most recent study, and the only study in which the BasiScript corpus was used, was conducted by Tellings et al. (2018b). They analysed the spelling of diphthongs in word dictations of 4,739 children from grade 2 and 3 (7-9 years old) and the spelling of weak verb suffixes (so-called d/t-errors) in 4,742 word dictations from fifth and sixth graders (10-12 years old). They detected these two types of errors automatically in the dictations and reported the error frequency for each word type. However, addressing only two types of error annotations and a limited set of analysed word types (diphthongs: 13 word types; weak verbs: 19 word types), yields limited insight into children's spelling ability.

So, in Dutch there has never been done a study in which not-dictated, freely-written texts are analysed on spelling errors. The main reason for this is probably that manual annotation of spelling errors using an annotation scheme consisting of many categories is difficult and very time-consuming. In addition, a large corpus of freely-written Dutch texts did not yet exist. Happily, this has recently changed with the publication of the BasiScript corpus (Tellings et al., 2018a).

3.2.3 Automatic annotation of German spelling errors

While automatic spelling error detection and annotation algorithms are not yet developed for Dutch, they are for German. Recently, two studies (Laarmann-Quante (2016); Berkling and Lavalley (2015)) presented algorithms to detect and annotate spelling errors automatically in German texts. Because German and Dutch are both relatively transparent languages (Borgwaldt et al., 2004), I

think that it should also be possible to develop such an algorithm for Dutch.

The algorithms described in these two German studies have in common that they first align the original spelling (containing spelling errors) and target spelling (in which spelling errors are corrected) of each word in each text. After that, they annotate each not-corresponding (sequence of) grapheme(s) with the spelling principle that is violated. The studies differ on two aspects from each other. The first aspect is the way in which the alignment is carried out, the second how the spelling errors are annotated.

Berkling and Lavalley (2015) first automatically obtains the phonetic transcription of both the original and target spelling. After that, they align these two phonetic strings using MARY (Schrödel & Trouvain, 2003), an algorithm based on articulatory features of phonemes. Using this phonetic alignment, the graphemes of original and target spelling can also be aligned with each other and errors can be detected as substitutions, insertions or deletions in the alignment. These errors are annotated with the spelling principles that is violated using a set of rules.

In the study by Laarmann-Quante (2016), alignment of original and target spelling is carried out differently. She first generates a lexicon containing all possible misspellings of a target word that occur when one or more spelling principles are applied incorrectly. From these possible misspellings, both the alignments with the reference and the violated spelling principle that causes the error are known. So, to detect and annotate the spelling errors in the original word, this word only needs to be compared with the generated possible error candidates. When a match is found, the alignment with the target spelling and the violated spelling principle can be deviated.

Berkling and Lavalley (2015) use 16 different tags to annotate spelling errors. These tags are divided over three main categories: capitalization, devoicing and vowel length. The scheme is not mutually exclusive, which means that not all errors can be annotated using these tags. On the contrary, Laarmann-Quante (2016) uses a large set of 69 error tags to annotate the spelling errors. These tags are divided over five main categories: phoneme-grapheme correspondence, syllable, morphology, syntax and other. Because of the “other” category, every error can be annotated, which makes the scheme mutually exclusive.

In conclusion, previous research shows that it is possible to develop algorithms that can automatically detect and annotate spelling errors in German texts. Because German is just as Dutch an alphabetical and relatively transparent language (Borgwaldt et al., 2004), the hypothesis of the first research question is that it should also be possible to develop an automatic spelling error detection and annotation algorithm to analyse Dutch texts from the BasiScript corpus.

3.3 FINDINGS FROM EARLIER RESEARCH TO DUTCH SPELLING ERRORS

So far, only qualitative research has been conducted to measure Dutch children’s spelling ability by analysing the spelling errors that they make in dictations. These previous studies presented several findings with respect to gender and the development of spelling proficiency, which will be discussed here.

3.3.1 Gender

Multiple studies have shown that gender plays a critical role in spelling development, with girls scoring generally higher than boys and no change in this pattern over all grades of elementary school (Schijf (2009); Keuning and Verhoeven (2008)). I expect to see this trend also in the BasiScript corpus. Therefore, the hypothesis to the second research question is that girls make on average less errors than boys in each grade of elementary school

3.3.2 Spelling development

In the first place, Keuning and Verhoeven (2008) have shown that children's spelling ability increases systematically from the beginning of second grade to the end of sixth grade. In addition, children in elementary school get spelling lessons and are expected to become better spellers by learning from these lessons. Therefore, the hypothesis to research question 3a is that the average number of errors made in each grade of elementary school decreases with increasing grade.

Two words are homophones when they are spelled differently, but their pronunciation is the same. An example of two words that are homophones are "flour" and "flower". An incorrectly written word and its correct spelling can also be homophones, like "school" (school) written as "sgool". In this study, these errors will be named homophone errors. Schijf (2009) found that the majority (62%) of spelling errors that children make in the first grade of high school (12-13 years old) are phonetically acceptable. Additionally, weak spellers have a worse developed phonological competence than normal spellers (Kay (1990); Van Luijn (1992); Horbach-Kleijnen (1997)). Therefore, the hypothesis of research question 3b is that misspelled words of older children are more likely to be homophone errors than misspelled words of younger children.

Furthermore, Horbach-Kleijnen (1997) compared Dutch weak and good spellers between 12 and 18 years old with respect to the type and number of spelling errors that they make. She found that good spellers make most errors in the morphology category (i.e., 31% of all errors), followed by the unmarked category (i.e., 23% of all errors). The unmarked category involves phoneme-grapheme conversion rules. For weak spellers, this is the other way around. Most errors are made in the unmarked category (i.e., 30%), followed by the morphology category (i.e., 22%). For both weak and normal spellers, slightly more than half the spelling errors that are made belong to the morphology or unmarked category. Therefore, the hypothesis for research question 3c is that spelling principles from the morphology and unmarked category are most frequently violated and that together, they label around 50% of the total amount of errors made.

In the last research question, the spelling errors in which a certain spelling principle is applied incorrectly are analysed with respect to the number of possible spelling errors. This yields a relative score that explains which spelling principles are most problematic to master in each grade. Schijf (2009) found in research among first graders of high school (12-13 years old) that grammatical morphological spellings are most difficult to master. This category contains all spelling principles concerning verb spelling. In addition, Bosman (2005) found that verb spelling performance of Dutch elementary school children is relatively low, only just above chance level. Assink (1985) provides a possible reason for this finding. He states that for writing the right verb suffix in Dutch, which

is often a “d” or a “t”, active application of syntax spelling principles is always necessary (i.e., first route of spelling process), while for writing other words, the orthography of a certain word is at a certain moment saved in long-term memory and can be achieved automatically (i.e., second route of spelling process). For example, the verbs “geloofst” (believes) and “geloofd” (believed) both exist and are pronounced in exactly the same way. Whether a “t” or a “d” has to be written, depends on the function of the verb in the sentence: a past participle ends in "d", a 2nd/3th person present singular form in "t". In contrast, for knowing that the word “hond” (dog) is written with a final “d” and not with a "t", the final devoicing rule from the morphology category needs to be applied. However, the word “hont” does not exist, so children can just remember that “hond” is always written with a “d”. Based on these earlier studies, the hypothesis of research question 3d is that spelling principles from the syntax category are most problematic in each grade.

4 METHODOLOGY

The methodology of this study consists of four sections and starts with a description of the data that is used in this study: Dutch texts from the BasiScript corpus. This is followed by a detailed step-by-step description of the design, implementation and application of the spelling error detection and annotation algorithm. Lastly, is explained how the algorithm is evaluated and how the detected and annotated spelling errors are visualized.

4.1 THE DATA: TEXTS FROM THE BASISCRIPPT CORPUS

The written data used in this study is part of the BasiScript corpus (Tellings et al., 2018a). This is a corpus containing two types of data written by elementary school children: dictations (lists of words) and texts. Both types of data are written in Dutch, which is the mother tongue of the children. In the present study, only the texts were analysed. These texts are handwritten by elementary school children from grade 2 (7-8 years old) to grade 6 (11-12 years old).

The data had been collected in six consecutive rounds. Spread over three school years, there was a data collection round every autumn and spring. In each round, the participating children wrote maximally five items with pen on paper: one dictation, two texts about a given theme and two free-themed text. Sometimes, not all five items were written by all children. In addition, not all schools participated in all data collection rounds they were invited to.

The BasiScript corpus contains two digitized (i.e., typed) versions of each handwritten text: the target text (the intended text, without spelling errors) and the original text (what the child actually wrote, including crossed-out words and spelling errors). Each target text contains lemma, morphemes and part-of-speech tag annotations, that were obtained using Frog 0.13, which is an advanced natural language processing suite for Dutch (Van den Bosch et al., 2007). In addition, also metadata is saved for each text. This metadata contains additional information about the text, like the name and grade of the author, the theme of the text and the date on which the text was written.

For each BasiScript text, the target text, original text, Frog annotations and metadata are saved in a FoLiA file (van Gompel & Reynaert, 2013). This is a practical format for XML-based linguistic annotation.

4.2 THE SPELLING ERROR DETECTION AND ANNOTATION ALGORITHM

This study entails the development of an algorithm that automatically detects spelling errors in the BasiScript texts and annotates these errors indicating the spelling principle that was violated. To be able to do this correctly, a multilayered approach is used. This means that each word in the BasiScript corpus is first annotated with multiple layers that all describe different orthographic properties, like the phonetic transcription and lemma of a word, or the date on which the text was written. The annotations are used to detect and annotate the spelling errors correctly.

Because Dutch is an alphabetical language, and each phoneme is represented using one or a sequence of letters, the algorithm was developed in such a way that it is able to detect errors

not at word or grapheme level, but at Phoneme-Corresponding Unit (PCU) level. This term was introduced by Laarmann-Quante (2016) in her research to automatic detection and annotation of German spelling errors, and refers to a sequence of graphemes that corresponds to one phoneme. In Dutch, this can be a sequence of only one grapheme (e.g., “k” or “p”), but also of two (e.g., “oe” or “ch”) or even three graphemes (e.g., “sch”). An example of how the word "school" is split into PCUs is visible in Table 1.

In the end, the developed algorithm consists of three parts: preprocessing of the texts, detection of PCUs and spelling errors, and annotation of PCUs with spelling principles. These three steps result in a multi-layered analysis of each BasiScript word. In appendix B an example is visible of the word "scholen" and all its annotation layers.

The complete algorithm was implemented and evaluated using Jupyter Notebook scripts in Python 3. Also visualization of the data was done in Jupyter Notebooks. For an overview of the technical pipeline, see appendix C

Table 1: Segmentation of the word "school" into PCUs. Although the word "school" is pronounced differently in both Dutch and English, the spelling and meaning are the same. The phonetic transcriptions are written using computer phonetic alphabet CGN2 (see appendix A).

Dutch (CGN2)	s	x	o	l
English (CGN2)	s	k	u	l
PCUs	s	ch	oo	l

4.2.1 Preprocessing of BasiScript texts

The BasiScript FoLiA XML files constituted the input to the algorithm. From these files, first the relevant information (i.e., target text, original text, metadata and additional Frog annotations) was extracted. After that, the data was preprocessed, such that the data was presented in the right format to detect errors in it.

Remove tags from original texts Each original text (i.e., the text with spelling errors) contained annotations to represent crossed-out tokens, letters written in mirror image, personal information and unreadable tokens. In the first preprocessing step, the algorithm removed all these tags, because they are not written in the target texts and would therefore complicate alignment of the original and target text.

Align target texts with the corresponding original texts After that, the algorithm aligned each target text with the corresponding original text using the ADAPT algorithm (Elffers, van Bael, & Strik, 2013). This is a dynamic algorithm based on the Levenshtein distance. Originally, ADAPT was developed for the alignment of phoneme strings. However, the current algorithm employs an altered version of ADAPT. In this version, the feature matrices of the algorithm have been modified, such that alignment of grapheme strings is possible. These modifications included extension of the two feature matrices that were originally present in the ADAPT-algorithm: accented vowels (e.g., ë, é and è) were added to the vowel feature matrix and accented consonants (e.g. ñ and ç) to the

consonant matrix. In addition, the new version of the ADAPT algorithm now also contains a third and fourth feature matrix, one for numbers and one for punctuation marks. In this way, all feature matrices together contain all possible characters that occur in the BasiScript texts.

The output of ADAPT constituted of an aligned target and original text. Table 2 shows how this looks like. In the final alignment, dashes ("-") have been added to the target text to represent inserted symbols and to the original text to represent deleted symbols.

Before the ADAPT algorithm was applied to the text data, the data had to be transformed into the right format. In the first place, hyphens that occur naturally in the texts, for example in words like "tv-toestel" (TV set) or "vrijdag- en zaterdagmiddag" (Friday and Saturday afternoon), were replaced with the hashtag symbol ("#"). This was done to avoid confusion with the dashes in the output of the ADAPT alignment. The hashtag symbol was hardly used by the children in their written texts, therefore this symbol was chosen. Secondly, all spaces were replaced with pipe symbols ("|"), because the ADAPT-algorithm interprets pipes as spaces. Thirdly, each text was split into shorter strings of approximately 150 characters. These shorter strings served subsequently as input to the ADAPT algorithm. This last step was necessary, because ADAPT is a dynamic algorithm. This means that it is not capable of handling large strings of symbols, like whole texts, because this costs too much memory. To make sure that the original and target texts were both split at a corresponding location, the algorithm checked whether the five characters before and after the split were exactly the same.

Table 2: Example of alignment using ADAPT of the original word "sgoole-" with its target word "scholen".

LAYER	VALUE
word	Scholen
target	Scho-len
original	s-goole-

Split the text into tokens After these short strings of approximately 150 characters had been aligned with each other, the algorithm split them into tokens (i.e., words, digits and punctuation marks). The spaces in the target string are used to determine where a new token starts.

File-specific metadata Subsequently, for each token, the corresponding metadata was saved in a set of layers. The metadata consisted of the anonymized name of the author, the gender of the author, the grade that the author is in, the date the text was written, the theme of the text, and the original file name. An example is visible in Table 3.

Token-specific Frog annotations The BasiScript corpus also contains lemma, morpheme and part-of-speech tag annotations of each token that the target transcription consists of. These annotations had been automatically obtained using Frog 0.13 (Van den Bosch et al., 2007), an advanced natural

Table 3: An example of an aligned target and original word with text metadata.

LAYER	VALUE
word	Scholen
target	Scho-len
original	s-goole-
author	JurC
gender	j (jongen, boy)
grade	6
date	najaar_2014 (autumn 2014)
theme	ThemaRarewoorden (Theme weird words)
fileName	d389055.xml

language processing suite for Dutch. The algorithm matched each token with the corresponding Frog annotations. An example is visible in Table 4.

Unfortunately, it was in some cases not possible to match the Frog annotations correctly. This is because some texts were split incorrectly into tokens in the previous step, which created non-existing tokens (e.g., the token "!Hallo" does not exist). To avoid problems with missing annotations, texts that contained non-existing tokens were removed from the analysed data.

Additional token-specific annotations Additionally, because the tokens were still in the right order, it was also possible to obtain information about whether a capital letter is at the beginning of a sentence, whether a punctuation mark is at the end of a sentence and whether two words are homophones of each other. For each token, I added three layers that represent the value of these variables. An example is visible in Table 4.

Table 4: An example of an aligned target and original word with Frog annotations and additional annotations. These annotations are all based on the target word.

LAYER	VALUE
word	Scholen
target	Scho-len
original	s-goole-
morphemes	[school, en]
lemma	school
pos-tag	N(soort,mv,basis)
punctEndSentence	False
capitalBeginSentence	True
homophones	True

Select words from all BasiScript tokens In the last preprocessing step of the algorithm, the complete set containing all annotated tokens, was split into three subsets: one containing only words, one with punctuation marks and one containing digits. Because the current study was not focused on punctuation errors, only the set containing words was selected for further analysis. Also the set with digits was not included, because using ADAPT for aligning a digit representation of a number (e.g., 16) with a text representation of a number (e.g., sixteen) yielded incorrect alignment. Finally, also words with an empty target or original transcription were removed from the data. These words were probably a result of incorrect alignment.

4.2.2 Detection of PCUs and spelling errors

In the second part of the algorithm, spelling errors were detected at Phoneme-Corresponding Unit (PCU) level. A PCU is a sequence of graphemes that correspond to one phoneme (Laarmann-Quante, 2016).

Lexicon Firstly, the algorithm created a list of all unique target words from the BasiScript corpus. The words in this list were cleaned and preprocessed, by removing the dashes that represent insertions and converting the words to lowercase. For each word in this list, the phonetic transcription was automatically obtained using a Dutch grapheme-to-phoneme converter webservice (ten Bosch, n.d.). The phonetic transcriptions were given in a computer phonetic alphabet called CGN2 (see appendix A). The target words together with their phonetic transcriptions formed a lexicon.

Rule-based grapheme-phoneme alignment Subsequently, each target word in the lexicon was aligned with its phonetic transcription. To perform this task, a rule-based sub-algorithm was developed. This sub-algorithm contained rules that describe which combinations of graphemes can be used to write one phoneme. This is an example of two of these rules: in Dutch, the phoneme /a/ can be written using an "aa" as in "maan" (moon), or an "a" as in "maken" (to make). The complete list of rules is included in appendix D. The resulting phoneme-grapheme alignments provided information about which sequence of graphemes corresponded to each phoneme, which are the PCU-segmentations. The algorithm used this sub-algorithm to extend the lexicon with the PCU-segmentations of each unique target word.

Derive the PCU-segmentations from the lexicon Finally, the algorithm used the created lexicon to deduce the PCU-segmentations of the aligned original and target word. Therefore, in the first place, the PCU-segmentation of the target word was obtained by selecting the PCU-segmentation of the target word from the lexicon. Next, dashes were inserted to the PCU-segmentation to represent inserted PCUs. The position of these dashes was deviated from the target alignment. After that, the PCU-segmentation of the original word was deduced from the PCU-segmentation of the target word and the alignment of the original with the target word. This was done by copying the PCU-segmentation boundaries from the target PCU-segmentation to the original word. In this final alignment of target and original PCUs, spelling errors were detected as inserted, deleted or

substituted PCUs. Table 5 presents an example of the aligned PCU-segmentations of the target and original transcription.

Table 5: The three last layers in this table represent an alignment of the target phonemes with the target PCU-segmentation and original PCU-segmentation. In this example, four spelling error are detected

LAYER	VALUE					
word	Scholen					
target	Scho-le-					
original	s-goole-					
phon_target	s	x	o	l	@	-
pcus_target	S	ch	o	l	e	n
pcus_original	s	g	oo	l	e	-

4.2.3 Annotation of PCUs with spelling principles

In the third part of the algorithm, the PCUs were annotated with spelling principles. An example of how these annotation layers look like is visible in Table 6. The goal of this part is to annotate each incorrectly spelled PCU with the spelling principle that was violated (the error and error_capital layers) and to annotate each correctly spelled target PCU with the spelling principles that should have been applied to write that PCU correctly (the basis and basis_capital layers). In the resulting multilayered annotation scheme, the error layers represent information about which spelling principles are violated in the spelling errors, and the basis layers make it possible to put this information into context.

Table 6: Example of word in which spelling errors are labeled (error and error_capital) and in which properties of the target spelling are labeled (basis and basis_capital).

LAYER	VALUE					
word	Scholen					
target	Scho-len					
original	s-goole-					
phon_target	s	x	o	l	@	-
pcus_target	S	ch	o	l	e	n
pcus_original	s	g	oo	l	e	n
error	-	UnSub1	CoVs1	-	-	MoEndN1
error_capital	SyCap1	-	-	-	-	-
basis	Un	Un	CoVs1	Un	Un	MoEndN1
basis_capital	SyCap1	-	-	-	-	-

Annotation scheme To annotate the PCUs with spelling principles, an annotation scheme is necessary. I decided to base the annotation scheme that the algorithm uses largely on the one by Horbach-Kleijnen (1992), because this is a linguistic scheme that describes Dutch orthography.

In total, the used annotation scheme consists of 35 basis spelling principles and 40 error spelling principles. The set of basis spelling principles is a subset of the error spelling principles. The number of spelling principles in both sets differ, because some error spelling principles depend on the value of the original PCU and can therefore not be used as basis spelling principle.

For clarity, the spelling principles are grouped into two main categories: unmarked PCUs and marked PCUs. Unmarked PCUs can be written correctly using only phoneme-to-grapheme conversion spelling principles. Marked PCUs are PCUs for which application of an autonomous spelling rule is necessary to write them correctly. In the used annotation scheme, these marked spelling principles are divided into four categories: Marked by Context (the context of a PCU determines how it is written), Marked by Morphology (morphemes with the same meaning are written in the same way), Marked by Syntax (the function of a word determines the spelling of the suffix), and Marked by Semantics (the meaning of a word influences the spelling). Table 7 shows for each category how many error and basis spelling principles belong to that category. In addition, this table also presents for each category an example of a spelling principle that belongs to that category. Appendix E contains an overview of the complete annotation scheme with all spelling principles. Some spelling errors are insertions of PCUs. This means that the incorrectly written original PCU is aligned with an empty target PCU. In this case, this empty target PCU is annotated with "Ins" which stands for insertion. It is not possible to annotate correctly written PCUs with this annotation.

Naming of spelling principles The spelling principles in the annotation scheme are named using a naming strategy. The first two/three letters always represent the category, which is Context (Co), Morphology (Mo), Syntax (Sy), Semantics (Sem) or Unmarked (Un). After that, an abbreviation of the theme is written down, for example Apostrophe (Ap) or Consonant Doubling (Cd). This subcategory is followed by a number (i.e., 1, 2, 3, etc.), which indicates the different spelling principles within a subcategory.

In some cases, it is possible to split a spelling principle into several sub-rules. These sub-rules are marked with the suffix a, b, c, etc. An example of spelling principle 17 with its two sub-rules is visible in Table 8. This example shows that MoFd1 is the first final devoicing (Fd) spelling principle from the morphology (Mo) category. This rule can be split into MoFd1a (voiced obstruent "d" is written as unvoiced "t") and MoFd1b (voiced obstruent "b" is written as unvoiced "p"). The sub rules always depend on the value of the target PCU and original PCU, therefore they cannot be used as basis spelling principles. The focus of this research is on the spelling principles, and not on the sub-rules principles. However, the sub-rules are often used to detect whether a PCU should be annotated with a spelling principle.

PCU annotation process For each spelling principle, the algorithm contained a Boolean rule that gets as input the target PCU, original PCU, and some extra information from the annotation layers (e.g., lemma, morpheme(s), and the position of the PCU in the word). Using these inputs, each

Table 7: This table represents an overview of the annotation scheme used to annotate both incorrectly spelled PCUs (error) and correctly spelled PCUs (basis). The spelling principles in the annotation scheme can be subdivided over five categories. For each category is represented how many spelling principles it contains, and one spelling principle used to annotate errors is shown as example.

Category	Number of Spelling Principles		Example Spelling Principle (used for error annotation)		
	Error	Basis	Name	Description	Example (original/target)
Marked by Context	9	8	CoCd1	Consonant Doubling: A consonant is doubled if it is written after a short vowel (excluding sjwa) and if it is not at the end of the word.	joken / jokken
Marked by Syntax	15	15	SyVt1	When the stem of a verb ends in an unvoiced sound (the sounds in "'t kofschip"), the past tense suffix starts with an unvoiced /t/ sound ("t").	hij krasde / hij kraste
Marked by Morphology	10	10	MoFd1	Final devoicing: voiced consonants are pronounced as unvoiced at the end of a word.	hont / hond
Marked by Semantics	1	1	SemCap1	Every proper name, title and some abbreviations start with a capital letter.	nijmegen / Nijmegen
Unmarked	5	1	UnSub2	A PCU is substituted with another PCU that does not correspond to the same phoneme.	kiefde / liefde
Total	40	35			

Table 8: Spelling principle MoFd1, together with its two sub rules. This is the 17th spelling principle from the annotation scheme.

	Category	Name		Description of spelling error	Examples	
		Error	Basis		Target	Original
17	Final Devoicing	MoFd1	MoFd1	Final Devoicing: Voiced obstruent becomes voiceless at end of word		
		MoFd1a	MoFd1	Voiced obstruent "d" is written as "t"	hond	hont
		MoFd1b	MoFd1	Voiced obstruent "b" is written as "p"	club	clup

rule was able to determine whether the spelling principle it represented was violated (in case of error annotations) or should have been applied (in case of basis annotations). The annotation scheme contains two spelling principles that involve capitals letters. These are SemCap1 (every proper name starts with a capital letter) and SyCap1 (every sentence starts with a capital letter). When one of these two annotations were used, they were written down in separate layers, called error_capital and basis_capital. This was done because a PCU can contain both a capital letter error and a lowercase letter error (e.g., compare “America” and “aamerica”).

4.3 EVALUATION OF THE ALGORITHM

After application of the algorithm on texts from the BasiScript corpus, the developed algorithm is evaluated on two aspects. In the first place is verified that the original and target texts had been properly aligned and split into words and PCUs. Therefore, 1000 incorrectly spelled words were generated (0,016% of all correctly and incorrectly spelled words from the BasiScript corpus) and checked on these aspects. This yields a percentage that indicates which part of this sample of 1000 words had been properly aligned and split into PCUs.

The second aspect on which the algorithm is evaluated is the correctness of spelling principle annotations with which the PCUs have been labeled. All target PCUs should have been labeled with a basis annotation that indicates the spelling principle that was applied correctly, and all incorrect original PCUs (i.e., spelling errors) should have been labeled with error annotations that indicate the spelling principle that was violated. To verify whether the target PCUs and incorrect original PCUs had been annotated correctly, for each of the 40 spelling principles used for error annotation and 35 spelling principles used for basis annotation, 20 random words were generated that contain at least one PCU that is annotated with this principle (respectively in the error layer or in the basis layer). In these generated random words was checked whether the spelling principle annotation was used correctly to label a PCU. Subsequently, the precision (see equation 1) is computed for each basis and error spelling principle by dividing the number of correctly used annotations by the total number of annotations (=20). This yields a number between 0 and 1 that indicates to what extent the spelling principle was used correctly to label PCUs. The higher the precision, the better the algorithm is in annotating PCUs with that spelling principle. This is the formula used to compute the precision of each spelling principle X:

$$\text{Precision}(X) = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} = \frac{\text{PCUs truly labeled with X}}{\text{PCUs truly labeled with X} + \text{PCUS falsely labeled with X}} \quad (1)$$

4.4 VISUALIZATION OF SPELLING ERRORS IN THE BASISSCRIPT CORPUS

After implementation, application and evaluation of the algorithm, the multilayered annotations of each BasiScript word are used to explore spelling errors in the data. For this reason, Python 3 is used to create bar plots from the data that visualize the spelling errors for grade two to six with respect to gender (using the layer "gender"), homophones (using the layer "homophones"), frequently violated spelling principles (using the layers "errors" and "errors_capital"), and most problematic spelling principles (using the layers "errors", "errors_capital", "basis" and "basis_capital").

To find out what the most frequently violated and most problematic spelling principles are, two measures were computed. These measures are the Normalized Spelling Error Frequency (NSEF) and Relative Spelling Error Frequency (RSEF).

4.4.1 Normalized Spelling Error Frequency

The NSEF represents for each grade and spelling principle which percentage of the PCUs in that grade is written incorrectly, and with which violated spelling principle these incorrectly written

PCUs are labeled. This percentage is computed for each spelling principle X and grade G in the following way:

$$\text{NSEF}(G, X) = \frac{\text{Nr. of PCUs in grade } G \text{ labeled with } \mathbf{error} \text{ spelling principle } X}{\text{Total number of PCUs in grade } G} \times 100\% \quad (2)$$

Because each spelling principle belongs to one of the five categories (i.e., Unmarked, Marked by Context, Marked by Morphology, Marked by Syntax and Marked by Semantics), the NSEF can also be computed at a more general level for each category. In that case, the X in formula 2 represents a category, and not a spelling principle.

4.4.2 Relative Spelling Error Frequency

Spelling principles from some categories are much more frequent than others. For example, rule SyCap1 (every sentence starts with a capital letter) is applied once in every sentence, while rule SemCap1 (every proper name is written with a capital letter) is used less frequently. This is simply because not every sentence contains a name. For this reason, it is also interesting to inspect the error frequency of each category in relation with how often a rule from a certain category should be applied. This is measured by computing the Relative Spelling Error Frequency (RSEF) for each spelling principle X and grade G as follows:

$$\text{RSEF}(G, X) = \frac{\text{Nr. of PCUs in grade } G \text{ labeled with } \mathbf{error} \text{ spelling principle } X}{\text{Nr. of PCUs in grade } G \text{ labeled with } \mathbf{basis} \text{ spelling principle } X} \times 100\% \quad (3)$$

In equation 3, variable X can also represent a category instead of a spelling principle. The RSEF can only be computed for the basis spelling principles, which is a subset of the error spelling principles.

5 RESULTS

This chapter presents the results of this study. First, a description of the analysed BasiScript data is given. This description is followed by the evaluation of the performance of the developed spelling error detection and annotation algorithm. Lastly, findings from quantitative spelling error analysis in the BasiScript corpus using the multilayered annotations that the algorithm yields are presented.

5.1 DESCRIPTION OF THE ANALYSED DATA

5.1.1 Analysed texts

In total, 6,028,023 words extracted from 70,593 BasiScript texts written by children from grade two to six were analysed by the algorithm. Table 9 shows for each grade how many texts were analysed, by how many unique authors these texts were written and what the average length of the texts is. From this table can be concluded that there were between 12,498 and 15,512 texts analysed in each grade. These texts are written by 3224 to 4943 different authors. In addition, the average length of the texts, measured in words, increases over the grades, from 44.85 words on average in grade 2 to 109.97 words on average in grade 6. Also the variety in text length increases with increasing grade. This variety is measured using the standard deviation. Table 9 also shows that there are texts in the corpus that contain only one word. The text with most words is written by a sixth grader. This text contains 2462 words. So, in each grade, at least twelve thousand texts were analysed. In addition, the older children get, the longer the texts they wrote and the more variety in text length was discovered within each grade.

Table 9: This table shows for each grade how many texts are analysed, by how many different authors these texts are written, and what the mean, standard deviation, minimum and maximum text length is.

Grade	Nr. of texts	Nr. of unique authors	Length texts (words)			
			Mean	Std	Min	Max
2	12,498	3224	44.85	31.40	2	640
3	12,773	3429	72.33	53.15	1	1758
4	14,777	4413	85.88	59.96	1	1220
5	15,512	4943	104.91	61.85	2	977
6	14,979	4574	109.97	77.59	3	2462

5.1.2 Analysed words and PCUs

With respect to the number of words and PCUs that were analysed in each grade (see Table 10), it can be concluded that most analysed words and PCUs are written by sixth graders. In addition, Table 10 also shows how many words were written incorrectly in each grade, and which percentage of the total number of written words this is. From this table can be concluded that the percentage of incorrectly written words and PCUs dropped when children got older. In addition, the percentage

of incorrect words was in each grade lower than the percentage of incorrect PCUs. So, although most analysed words were written by sixth graders, the percentage of incorrect words was lowest for writers from this grade.

Table 10: Number of analysed words and PCUs per grade, together with the fraction of words and PCUs that was spelled incorrectly.

Grade	Nr. of words			Nr. of PCUs		
	Total	Incorrect	% Incorrect	Total	Incorrect	% Incorrect
2	560,543	148,657	26.52%	1,970,033	206,413	10.48%
3	923,863	182,914	19.80%	3,392,240	24,622	7.26%
4	1,268,975	176,935	13.94%	4,732,989	241,760	5.11%
5	1,627,337	172,845	10.62%	5,995,222	229,864	3.83%
6	1,647,305	145,596	8.84%	6,129,588	193,977	3.16%

5.2 EVALUATION OF THE ALGORITHM

5.2.1 PCU segmentation and alignment

Firstly, the developed algorithm was evaluated on how well the original and target texts had been aligned and split into words and PCUs. The results show that from a sample of 1000 words from the BasiScript corpus, 97.9% of these words were aligned and split correctly. The most common types of alignment errors are visible in Table 11. This table shows that the algorithm had especially difficulty with aligning foreign loan words correctly, because some of these words contain phoneme-to-grapheme mappings that have not been defined as phoneme-grapheme alignment rules.

Table 11: Description and frequency of alignment errors encountered in a sample of 1000 words from the analysed BasiScript data.

Description of the error	N	Example
Phoneme-grapheme mapping is not known by phoneme-grapheme alignment algorithm (especially with foreign loan words)	8	tar_phon r e - s @ n tar_pcus r - a c e n orig_pcus r - a c e -
PCU could better have been aligned with PCU next to it	7	tar_phon v e r z o r G d - tar_pcus v e r z o r g d - orig_pcus v e r z o r g ch t
PCU segmentation contains non-existing PCU	4	tar_phon l e - s tar_pcus l e - s orig_pcus l ez e s
Wrong phonetic transcription	1	tar_phon z E G - d @ - tar_pcus z e - i d @ n orig_pcus z - - ij d @ n
Two completely different words are aligned	1	tar_phon w o - n p l E k tar_pcus w oo - n p l e k orig_pcus h a - - - r e n

5.2.2 PCU annotation

The second aspect on which the algorithm was evaluated is the correctness of spelling principle annotations with which the PCUs are labeled. Therefore, the precision was computed for all basis and error spelling principles.

Firstly, for 34 out of 40 spelling principles used for error annotation, a precision of 1.0 was found. For the other six error spelling principles, the precision scores were between 0.6 and 0.95 (see Table 12).

Secondly, the computed precision of the basis spelling principles was 1.0 for 30 out of 35 basis spelling principles. The five remaining basis spelling principles had a precision between 0.2 and 0.95 (see Table 12). What stands out from this table is that spelling principles with a low precision for basis annotation, also have a low precision for error annotation. In addition, SyCoN1 has a really low basis precision, which is only 0.2 and MoAs1 has a low basis and error precision (respectively 0.55 and 0.6). For a complete overview of all error and basis spelling principles and their corresponding precision scores, see appendix F.

Table 12: Precision scores for the six error spelling principles that have a precision lower than 1.0. Five of them also have a precision lower than 1.0 when used as basis annotation.

Name	Spelling Principle		Precision	
	Description	Example (original/target)	Error	Basis
SyCoN1	Between -n needs to be written between two morphemes in some composition words.	bijekorf / bijenkorf	0.8	0.2
MoAs1	Assimilation of stem (unvoiced consonant is pronounced as voiced or vice versa).	zeldsame / zeldzame	0.6	0.55
MoMi1	Miniaturization	achtien / achttien	0.95	0.7
MoAsMi1	Assimilation of stem followed by miniaturization.	opod / opbod	0.8	1.0
MoCoS1	You write a "between s" between two parts of a composition word if you hear that "s".	dorpstraat / dorpsstraat	0.95	0.95
MoCoS2	If you don't hear a "s" between two parts of a composition word, you don't write one.	hoofdsstraat / hoofdstraat	0.75	0.75

5.3 EXPLORATION OF THE DATA

5.3.1 Gender

Figure 1 shows for each grade which part of all written PCUs in that grade was misspelled by boys, and which part by girls. This figure reveals that boys made more spelling errors than girls in all grades. In addition, girls showed more progress over the years, because the percentage decrease between grade 2 and 6 is slightly higher for girls than for boys (respectively 71,41% for girls and 68,78% for boys).

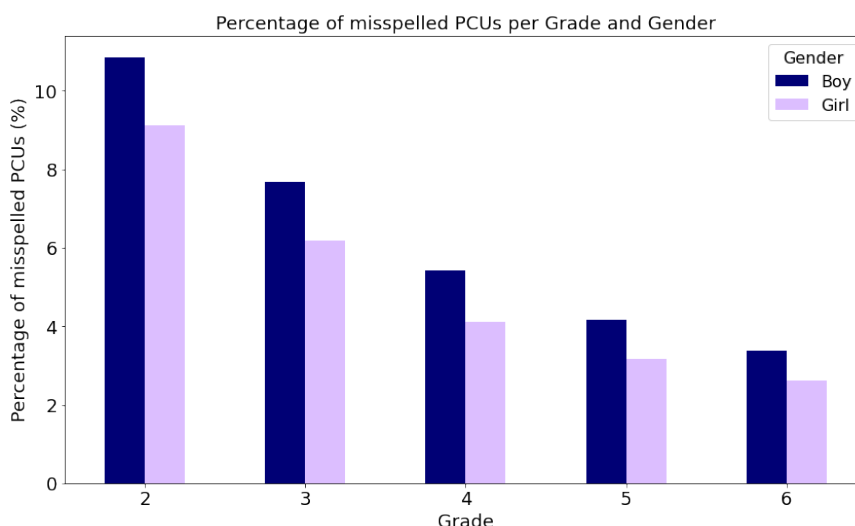


Figure 1: This plot shows for each grade which part of all written PCUs in that grade is misspelled by boys, and which part is misspelled by girls.

5.3.2 Homophone errors

Figure 2 shows for each grade which percentage of the incorrectly spelled words were homophone errors. In homophone errors, the pronunciation of the original and target word are the same, while the spelling of the original and target word do not overlap. An example of a homophone error is "vind" (find) written incorrectly as "fint". Each incorrectly written original word can be classified as a homophone error or non-homophone error. From Figure 2 is derived that the percentage of homophone errors increased with increasing grade, from 41.87% in grade 2 to 57.73% in grade 6.

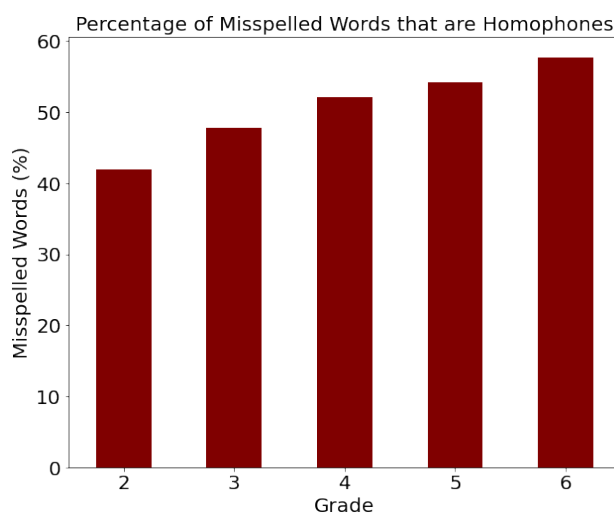


Figure 2: This plot shows for each grade which part of the original misspelled words are homophones with the target word.

5.3.3 Most frequently violated spelling principles

Categories of spelling principles The bar plot in Figure 3 represents for each grade and category the Normalized Spelling Error Frequency (NSEF; see formula 2 in section 4.4.1). This bar plot

shows that in all grades, the majority of spelling errors was labeled with a spelling principle from the Unmarked category and only a very small percentage (which is almost not visible) with a spelling principle from the Marked by Semantics category. In addition, the plot visualizes that the NSEF score of each category decreased with increasing grade, so the children became better spellers over time with respect to the spelling principles from all categories. The strongest decrease was measured for Marked by Context and Marked by Morphology errors (respectively a percentage decrease of 90.54% and 89.92%). Marked by Syntax lagged behind the other categories and had the smallest percentage decrease (only 55.11%).

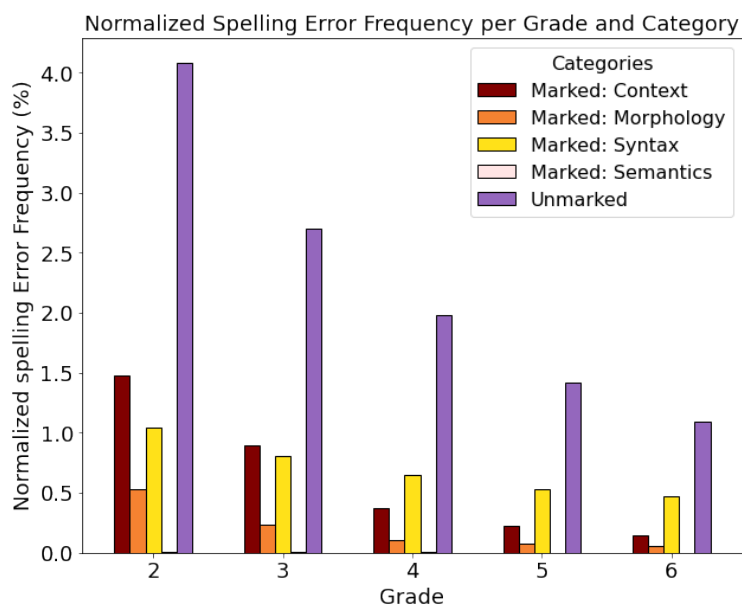


Figure 3: This bar plot represents for each grade and category the Normalized Spelling Error Frequency (NSEF). This measure indicates how many spelling errors (i.e., incorrectly written PCUs) are labeled with a specific category in each grade.

Individual spelling principles The next step is to consider the spelling errors at a more detailed level. This means focusing on the individual spelling principles instead of the categories. In total, the spelling errors were labeled with 40 different spelling principles, so presenting NSEF scores for each individual spelling principle, will result in a large and unclear overview. Therefore, only the NSEF scores of the spelling principles from the Unmarked category (Figure 4) and Marked by Syntax (Figure 5) category are displayed, because these are the categories in which most errors are made in sixth grade (Figure 3). The spelling principles in Figure 4 and 5 are named using a short tag. Appendix E contains a description of the spelling principle that the tag represents, together with an example.

The bar plots in Figure 4 and 5 show that the most frequently violated spelling principles in grade two, three and four was UnSub2 (Substitution of a PCU with a phonetically different PCU), while in grade five and six this was UnDel1 (a deletion of a PCU).

These four spelling principles belong to the Unmarked category, which means that they involve incorrect phoneme-grapheme mappings, and no incorrect application of autonomous spelling rules.

In addition, for almost all spelling principles in these two figures, the number of errors decreased with increasing grade. However, this trend was not detected for SyPer1 (every 2th/3th person singular simple present ends with a "t" suffix) and SyVd2 (past participles ending in a voiced phoneme get a "d" as suffix). The spelling principles UnSub1b (a substitution of a PCU with another PCU that corresponds to the same phoneme) and SyInf1 (Infinitive verbs end in "n") showed the greatest percentage decrease in grade six with respect to grade two. So, children showed most progress in learning these two rules.

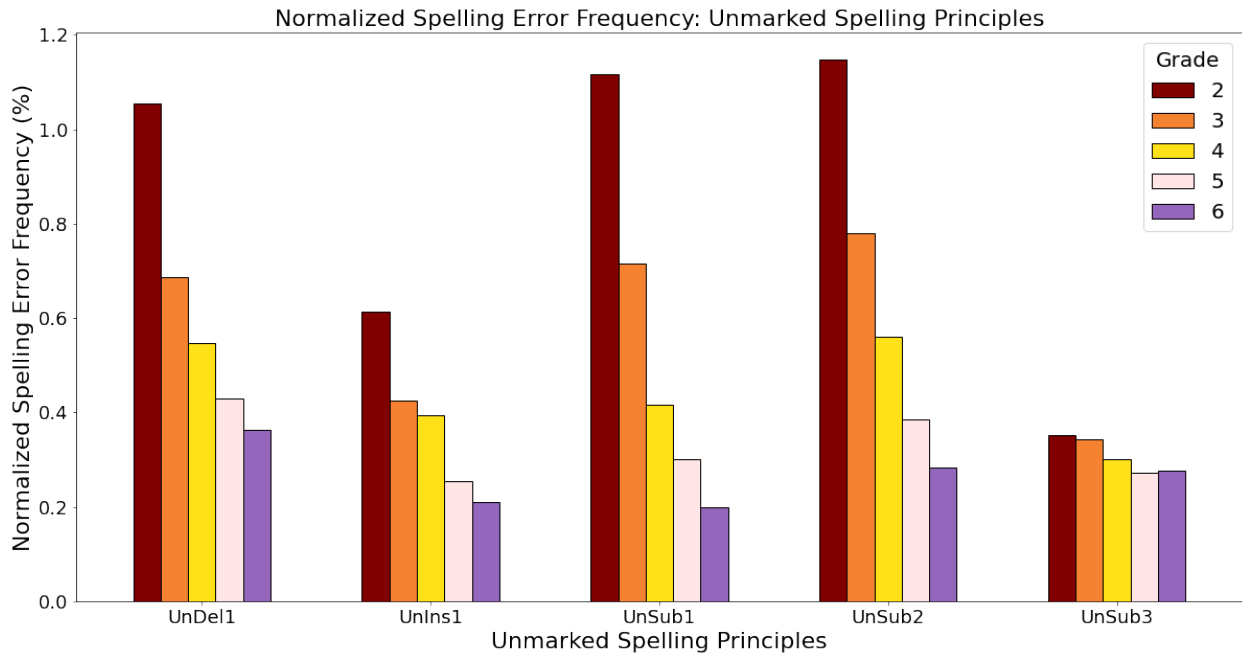


Figure 4: The Normalized Spelling Error Frequencies (NSEF) for all Unmarked spelling principles.

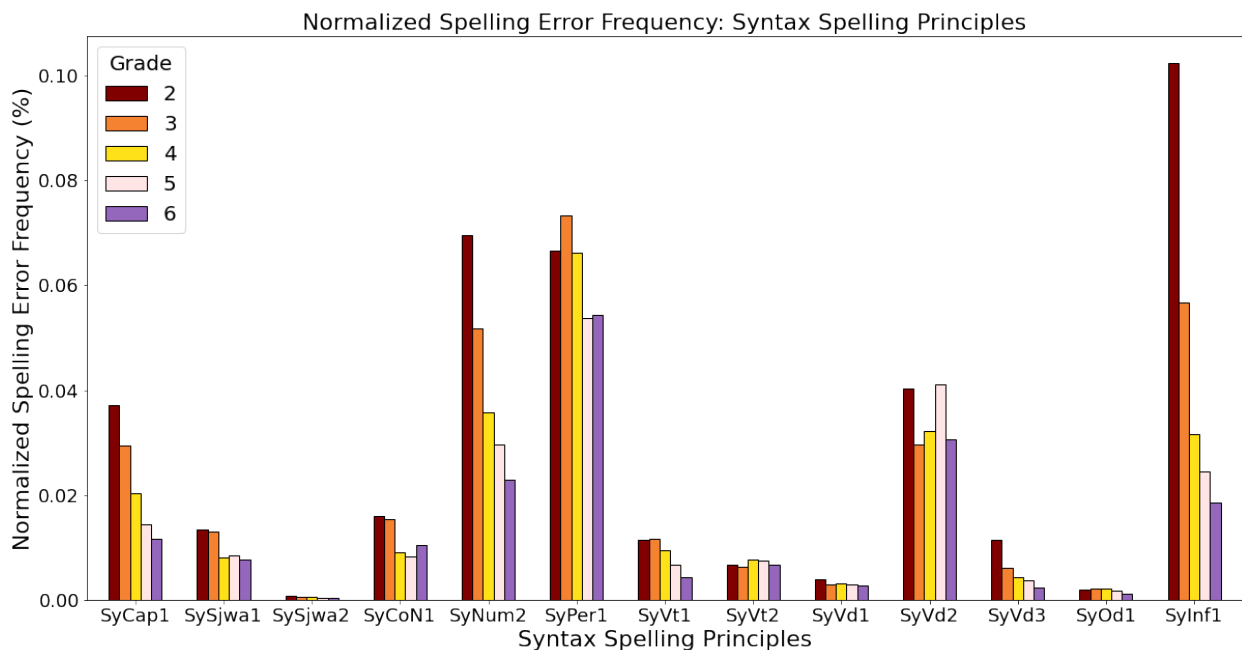


Figure 5: The Normalized Spelling Error Frequencies (NSEF) for all Marked by Syntax spelling principles.

5.3.4 Most problematic spelling principles

Categories of spelling principles The Relative Spelling Error Frequency (RSEF; see formula 3 in section 4.4.2) is a measure that represents the error frequency of a spelling principle in relation with how often that spelling principle should be applied. The RSEF scores of each grade and category (Figure 6), show that spelling principles belonging to the Marked by Semantics category were relatively most frequently violated and therefore most problematic to master in all grades. Secondly, also spelling principles from the Marked by Syntax category were problematic. In all categories, the RSEF score decreased with increasing grade. The smallest decrease is seen for spelling principles from the Marked by Syntax category.

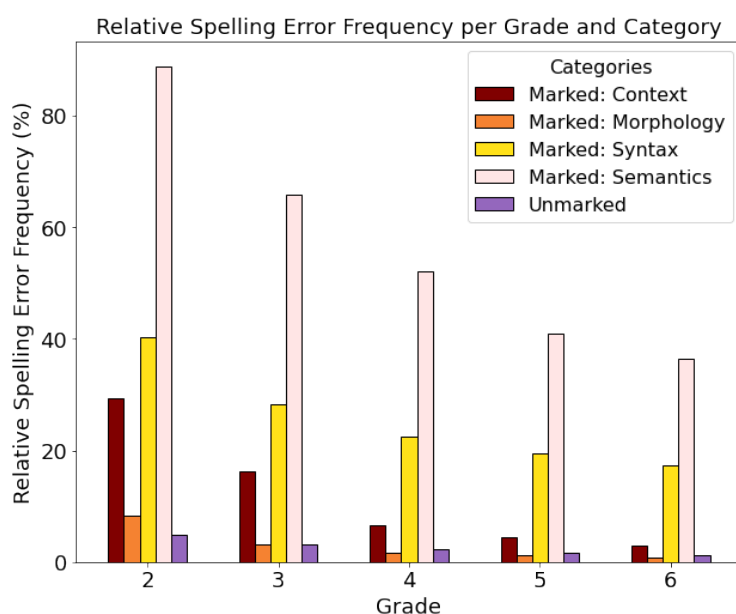


Figure 6: The Relative Spelling Error Frequency (RSEF) per category and grade. The higher the score of a grade, the more problematic to learn the spelling principles belonging to that category are. The bars for the category Marked by Semantics also represent the RSEF scores for spelling principle SemCap1, because this is the only spelling principle that the Marked by Semantics category consists of.

Individual spelling principles When inspecting the spelling principles from the Marked by Semantics and Marked by Syntax categories in greater detail, it is found that the Marked by Semantics category only consists of one spelling principle: SemCap1 (every proper name, title and some abbreviations should start with a capital letter). So, the RSEF score of this spelling principle is represented by the Marked by Semantics category in Figure 6. This figure shows that there was almost no awareness of SemCap1 in grade 2, because the principle was applied incorrectly 88.89% of the times. This percentage decreased over the grades, but in sixth grade, the rule is still applied incorrectly in more than a third (36.51%) of the times that sixth graders have to use this rule.

After that, the RSEF scores of the individual spelling principles from the Marked by Syntax category were inspected (Figure 7). In the first place, these scores show that children became better in applying all these spelling principles, because the RSEF scores decreased when children got older. In addition, there were three spelling principles that were still applied incorrectly in more

than 30% of the times that they should have been applied in sixth grade. These spelling principles are SyVd2 (past participles ending in a voiced PCU should get a "d" as suffix, and not a "t", while the suffix is pronounced with a /t/ sound), SyOd1 (every present participle should end in a "d", while a /t/ sound is heard) and SyCap1 (every sentence should start with a capital letter). So, these three spelling principles were most problematic to learn for the children.

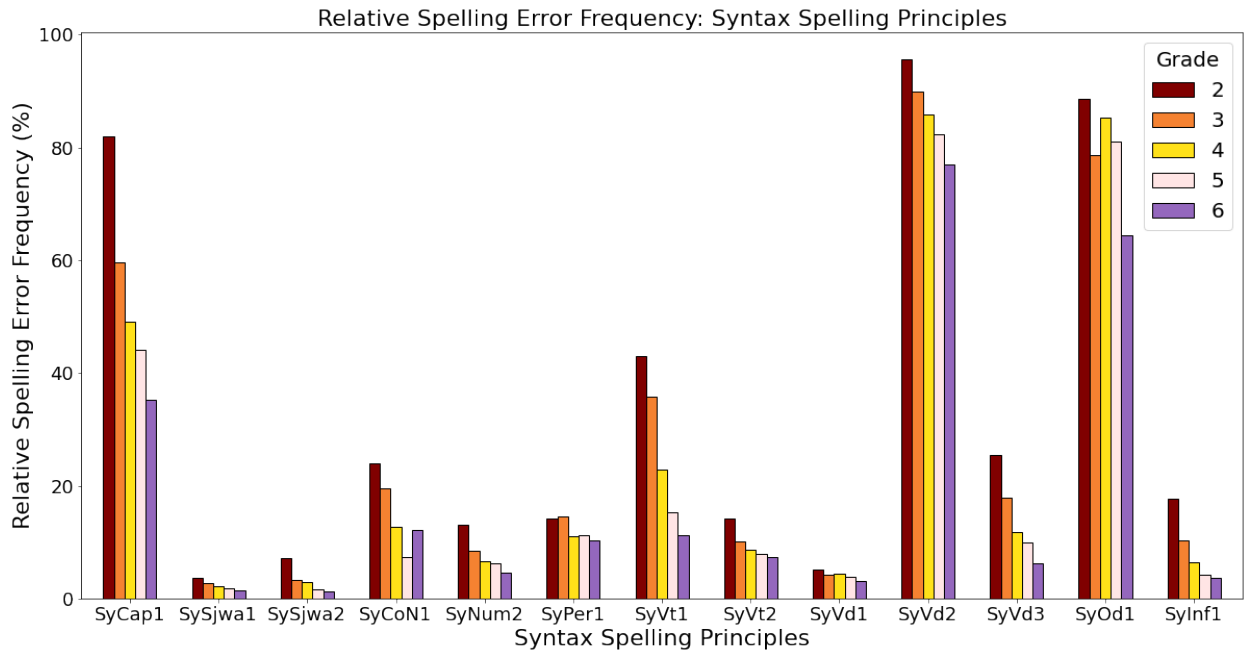


Figure 7: The Relative Spelling Error Frequencies (RSEFs) of the Marked by Syntax spelling principles.

6 DISCUSSION

This chapter contains a discussion of the results of the study. This discussion is structured in correspondence with the order of the research questions. This means that first the results with respect to development and evaluation of the spelling error detection and annotation algorithm will be discussed, followed by the results from the quantitative analysis of the BasiScript corpus with respect to gender and children’s spelling development.

6.1 THE ALGORITHM

The results of this study show that it is possible to develop an algorithm that is able to automatically detect and annotate spelling errors in the BasiScript corpus. So, the hypothesis of the first research question is confirmed. In addition, the current algorithm also has the ability to annotate correctly written PCUs (i.e., target PCUs) with the spelling principle that should be applied to write that PCU correctly. Results with respect to evaluation of the developed algorithm show that its performance on detecting spelling errors, which involves alignment of the PCUs of the target and original word, is very accurate. Furthermore, the high annotation precision for the majority of spelling principles reveals that the algorithm annotates most PCUs with the correct spelling principle.

The fact that Dutch is an alphabetical and relatively transparent language, just as German, plays an important role in the design of the Dutch spelling error detection and annotation algorithm. However, the algorithm design differs on certain aspects with respect to the two German algorithms that were developed earlier by Berkling and Lavalley (2015) and Laarmann-Quante (2016).

In the first place, Berkling and Lavalley (2015) obtain an alignment of the target and original PCUs by generating a phonetic transcription of both the target and original text and aligning these two phonetic transcriptions. In contrast, the algorithm developed in this study employed alignment of grapheme strings followed by phoneme-grapheme alignment. With respect to Berkling and Lavalley (2015), this alignment method has the advantage that automatic conversion of a misspelled word into phonemes is not necessary. This process can be problematic, because the automatic phoneme-grapheme conversion tool is not trained on converting non-existing words, which are misspelled words often.

In addition, Laarmann-Quante (2016) uses yet another method. In this method, first all possible error candidates are generated, together with the spelling principles that are applied incorrectly to obtain that error candidate. For each encountered misspelled word, all possible error candidates are searched to find the corresponding spelling principles that are applied incorrectly. An advantage of this method is that it provides very accurate annotation of the spelling errors. However, a major disadvantage is that the method is very computationally inefficient, and therefore not useful for annotating large corpora like the BasiScript corpus.

The current algorithm employs an annotation scheme with 35 tags for target PCU annotation and 40 tags for spelling error annotation. These tags can be divided into five categories. In comparison with earlier Dutch studies that used annotation schemes this is an extensive scheme. Most studies only annotated errors at a very general level using a few categories (e.g., Keuning and

Verhoeven (2008); Schijf (2009)). Only Horbach-Kleijnen (1997) used a scheme with more tags. Therefore, the currently employed annotation scheme is based on this scheme, but not literally the same. This is because it was not possible to automate the detection of all spelling principles from the scheme by Horbach-Kleijnen (1997).

6.1.1 Limitations to the algorithm design

First, three limitations with respect to the design of the algorithm will be discussed. A problem that occurred while applying the algorithm on the BasiScript texts, is that some BasiScript texts cannot be processed correctly by the algorithm. For these texts, splitting the aligned target and original texts into words was not always successful. The result is that some words that do not form a word together, are seen as one word by the algorithm. These non-existing words cannot be matched with the lemmas, morphemes and POS-tag annotations that each BasiScript file contains on word level. To solve this problem, all texts in which this phenomenon happened were removed from the data. This is a substantial loss of data (16,075 texts; 18.5% of all texts). A better way to overcome this problem would involve changing the method of splitting texts into words, for example by using a Dutch word and punctuation mark tokenizer function provided by the Python package UCTO, which is also incorporated in Frog (Van den Bosch et al., 2007).

Another limitation with respect to the design of the algorithm concerns the grapheme-phoneme alignment process, in which the graphemes of the target word are aligned with their phonemes, which results in a PCU-segmentation. The problem is that this phoneme-grapheme alignment algorithm is not perfect. For example, some rules are missing, like the “b” is only defined with a /b/ sound, and not with a /p/ sound. Also, the algorithm has problems with alignment of some foreign loan words, like “Youtube” and “Wien”, in which graphemes are pronounced with sounds that are not common in Dutch (and thus are not defined as rules in the algorithm). In the current study, all words for which a correct alignment of the graphemes and phonemes of the target word was not possible were deleted. A better way to overcome this problem would be to extend phoneme-grapheme alignment algorithm with extra rules.

The third limitation concerns the low precision scores for six spelling principles used for error annotation, of which five also have a low precision for annotation of target PCUs. This means that some spelling errors or target PCUs are labeled incorrectly with these spelling principles. The algorithm can be improved by changing the functions that recognize these errors and target PCUs in the data, such that a higher precision is achieved.

6.1.2 Limitations to the algorithm evaluation method

There are also two limitations with respect to the method that was chosen to evaluate the algorithm. The major concern is that the automatic annotations are not compared with manual annotations. This comparison would allow computation of the recall metric, next to the precision metric. The recall metric is useful, because it provides insight into the number of false negatives, which are spelling errors that are annotated with an unmarked spelling principle, while they should actually be annotated with a marked spelling principle. Important to mention here is that all spelling error

that do not fit into a marked category are currently annotated with an unmarked spelling principle.

In addition, the currently computed precision scores are for each spelling principle based on a sample of only twenty words, containing at least one PCU that is annotated with the currently selected spelling principle. This is a small sample, especially for high-frequent spelling errors, and makes it possible that certain incorrect annotations are not detected. To enlarge the reliability of the algorithm in a follow-up study, it is recommended that evaluation of the algorithm encompasses the comparison of automatic annotations with manual annotations and the computation of both precision and recall.

6.2 GENDER

The developed algorithm was used to analyse spelling errors in the BasiScript corpus. The results of the analysis with respect to gender show that girls make less errors than boys in all grades of elementary school, which means that girls are on average better spellers. This is in line with findings by Schijf (2009) and Keuning and Verhoeven (2008) and confirms the hypothesis of the second research question.

6.3 CHILDREN'S SPELLING DEVELOPMENT

The third aim of this research is to investigate how the spelling performance of elementary school children develops over time. The spelling performance is measured by the number of spelling errors, the percentage of homophone errors, the most frequently violated spelling principles and the most problematic spelling principles. With respect to these four aspects, the results show that the general trend is that children become better spellers between grade two and six of elementary school.

6.3.1 Number of spelling errors

In the first place, the results of the present study show that the number of errors elementary school children make decreases with increasing grade. This is in accordance with the hypothesis of research question 3a, which is based on findings by Keuning and Verhoeven (2008) that children's spelling ability increases systematically over time. Thus, children show progress in each grade and make less errors over time.

6.3.2 Homophone errors

Subsequently, another finding of the study is that the older children get, the higher the percentage of homophone errors with respect to the total amount of errors. This outcome confirms the hypothesis of research question 3b. A homophone error is an incorrectly written word that has exactly the same phonetic transcription as the correct target word. A possible explanation for this finding is the fact that older children have a better developed phonetic competence than younger children.

6.3.3 Most frequently violated spelling principles

The results show that in each grade, most errors are annotated with a spelling principle from the unmarked category. Within this category, the spelling principle UnSub2 (substitution of a PCU with

a phonetically different PCU) is most frequently violated in grade two to four and spelling principle UnDel1 (deletion of a PCU) is most frequently violated in grade five and six. The hypothesis of research question 3c, that spelling principles from the morphology and unmarked category are most frequently violated and that they label together around 50% of the total number of errors, is not confirmed, because the unmarked category on its own labels more than 50% of the errors in each grade.

The hypothesis and results of this study deviate on the finding that the number of morphology errors is not as high as expected. One possible explanation is that the algorithm does not work perfectly yet on annotating morphology spelling principles. This is illustrated by results from the evaluation of the algorithm. Here, five from the six spelling principles that do not have a precision of 1.0 are morphology spelling principles. In addition, because the algorithm is not compared with manual annotations and evaluated on the number of false negatives, it is possible that some spelling principles are incorrectly annotated with an unmarked spelling principle, since this is the residual category. All PCU that are not annotated with a marked spelling principle are automatically annotated with an unmarked spelling principle.

Another explanation for the deviation between the hypothesis and actual findings is that the hypothesis is based on findings from a study that analysed written dictations by high school children, while in the current study written texts from elementary school children are analysed.

Finally, to write unmarked PCUs correctly, it is only necessary to apply phoneme-grapheme conversion rules. There is no need for the application of autonomous spelling principles, as is the case with marked spelling principles. Since the application of a phoneme-grapheme conversion rule is necessary in writing almost every word, and thus is much more frequent than application of a marked spelling principles in Dutch, it is also more likely that there are more errors made in the unmarked category than in the marked categories. This is a last explanation for the finding that the most frequently violated spelling principles are unmarked

6.3.4 Most problematic spelling principles

The most problematic spelling principles are discovered by inspecting how often a spelling principle is violated with respect to how often it is necessary to apply that spelling principle (i.e., the relative frequency). The results show that spelling principles from the semantics and syntax category have the highest relative frequency in all grades of elementary school, and are therefore most problematic.

In these two categories, there are four spelling principles that are still applied incorrectly by sixth graders in more than 30% of the times that they are used. Two of these spelling principles concern capital letter use. This finding is not in line with the hypothesis to research question 3d, because verb spelling was expected to be most problematic. The other two problematic spelling principles concern past participles and present participles ending in a "d", while you hear a /t/ sound. This result is in line with the hypothesis of research question 3d and supported in findings by Assink (1985) and Bosman (2005).

In conclusion, the hypothesis to research question 3d was partially confirmed. The most surprising finding is that spelling principles concerning capital letter use are relatively often applied

incorrectly, while these rules are quite simple to learn and apply, especially with respect to the difficult verb spelling rules. Therefore, more attention to direct instruction of the capital letter spelling principles during spelling lessons could potentially improve spelling proficiency of Dutch children a lot.

6.4 POSSIBLE FUTURE DIRECTIONS

6.4.1 Further analysis of the BasiScript corpus

This study yields a multilayered analysis of more than six million words extracted from the BasiScript corpus. These layers include information about the child that wrote the word, but also about each word's orthographic properties, phonemes, PCUs, and spelling errors. Using these layers, the spelling errors in the BasiScript corpus can be explored much further than was done in this study.

In a possible future study, the spelling development can for example be analysed in greater detail by involving information from the layer "date". This layer states whether the text was written in spring or autumn, which makes it possible to split each grade into two grades, for example "grade 5 autumn" and "grade 5 spring". Another way to study the spelling development in greater detail, in by using the sub spelling principles (see appendix E) to annotate the spelling errors.

Another direction for further exploration of the BasiScript corpus is research to the development of individual spelling performance. Such a longitudinal study is possible, because many children that wrote the BasiScript texts participated in subsequent data collection rounds.

In addition, it is also possible to extract statistical information from the BasiScript corpus, like the most frequently misspelled words and PCUs and why these words and PCUs are misspelled. Especially in combination with metadata, like gender and grade of the author, the part-of-speech tag of a word, and the theme of the text the word belongs to, it is possible to improve the spelling lessons, such that they better fit the needs and perceptions of the children.

Another corpus, BasiLex, consists of Dutch texts that are specifically written for children. From these texts, word frequencies can be obtained of all words that children from a specific grade are exposed to. Using these word frequencies, it is possible to investigate the relationship between how often a child is exposed to a word and the spelling errors made in that word. This is another route that can be explored in a future study.

6.4.2 Improvement and extension of the algorithm

The current algorithm already performs good on detecting and annotating most spelling errors in the BasiScript corpus, but the performance can still be improved on several aspects, such that the earlier described limitations are overcome. These improvements involve that all texts are split correctly into words, the grapheme-phoneme alignment algorithm is also able to align words with foreign roots, and all spelling principles have a high precision and recall, when used as error annotation, but also when used as target PCU annotation.

The developed algorithm contains 40 spelling principles to annotate spelling errors and 35 spelling principles to annotate target PCUs. These are large numbers, but they do not yet cover all possible spelling principles. For example, there are currently no spelling principles in the annotation

scheme concerning spelling of strong verbs. Addition of these spelling principles would make the algorithm more detailed and better able to recognize all types of errors.

In addition, the annotation scheme does also not contain spelling principles to annotate the punctuation marks. Therefore, the punctuation marks are currently not analysed. The same applies to tokens that contain or are digits. In a future study, the annotation scheme can be extended, such that punctuation use can be analysed automatically. This can yield very interesting insights, because this has not been done before on such a large scale for Dutch writings by children.

Another way to extent the algorithm is by changing the output format. In the current algorithm, the output containing over six million annotated words is saved in one csv file. A disadvantage is that this file is very large. A better output format would be to add new spelling error annotation layers to the BasiScript FoLiA files (van Gompel & Reynaert, 2013) of each text.

6.4.3 Machine learning applications

Using the multi-layered annotations of each word, it is also possible to extract certain features from each text, like the number of errors in a specific category, the rate with which nouns are used, or the occurrence of certain phonemes. These text-based features enable the training of machine learning algorithms on texts written by children.

Possible supervised applications of these algorithms are authorship identification or grade prediction. In addition, a possible unsupervised application is to cluster the texts to discover similarities and differences between texts that tell something about the spelling performance of the writer and whether the writer is a weak, normal or strong speller.

6.4.4 Other applications of the algorithm

Next to automatic analysis of written, digitized texts, like the texts in the BasiScript corpus, the spelling error detection and annotation algorithm can also be used in other applications. For example, in combination with an automatic spelling error correction system, the algorithm can be implemented in a computer application that provides detailed online, real-time feedback on writing. Next to marking the PCU that is written incorrectly, the computer application can also explain why the PCU is written incorrectly, by giving information about the violated spelling principle. Because direct instruction of the spelling principles from the Dutch orthographic system (Assink, 1986) and immediate, not-delayed, feedback on writing (Harward et al., 1994) are essential for developing good spelling skills, it is expected that this computer application has great potential for improving writing skills of Dutch children.

Another possible application of the algorithm is to use it for research to children with spelling problems, like dyslexia. The algorithm can for example help to design tests in which freely-written texts of children are analysed, instead of dictations. These new tests can for example support measurement of children's spelling proficiency, such that the effectiveness of certain treatments can be assessed easily. Another advantage of these tests is that they can help to recognize children with dyslexia in an early stage. This is necessary, because a recent study to the spread of dyslexia in the Netherlands has found that in 2019, 7.5% of the children were in possession of a dyslexia statement

after the last year of elementary school and 14% after the last year of high school (Inspectie van Onderwijs, 2019). These numbers show that dyslexia is a common disorder among Dutch children, and that almost half of the children suffering from it are not diagnosed during elementary school. This means that spelling problems of many children are recognized too late, which makes it harder to help them.

7 CONCLUSION

The present research shows that it is possible to develop an algorithm that automatically detects spelling errors in the BasiScript corpus and annotates them with the spelling principle that is violated. In addition, the algorithm is also able to annotate correctly written PCUs with the spelling principle that was applied correctly. This algorithm, applied to the BasiScript corpus, facilitates automatic quantitative research to spelling errors made by Dutch elementary school children in freely-written texts.

Analysis of spelling errors in more than six million words from the BasiScript corpus shows that girls make less errors than boys. In addition, with increasing grade, the average number of spelling errors made in each text decreases, the percentage of spelling errors that are homophones with the target spelling increases, spelling principles involving only phoneme-grapheme conversions are most frequently violated and spelling principles concerning knowledge about Dutch syntax and semantics are most problematic to learn. These quantitative results are very useful to improve direct spelling instruction, which is essential for developing good spelling skills (Assink, 1986).

In future studies, further analysis of spelling errors in the BasiScript corpus can provide more detailed insights into which spelling errors are made by which type of children. In addition, it is possible to employ the algorithm in a computer application that provides support on spelling. Using the algorithm together with an automatic spelling checker, the application can provide immediate feedback on written texts, indicating not only which letters are written incorrectly, but also what the underlying reason is why these letters are incorrect by displaying the violated spelling principle. Such an application supports direct and immediate feedback on writing, which is essential for becoming a proficient speller (Harward et al., 1994).

REFERENCES

- Assink, E. (1983). *Leerprocessen bij het spellen*. Utrecht: Elinkwijk.
- Assink, E. M. H. (1985). Assessing spelling strategies for the orthography of Dutch verbs. *British Journal of Psychology*, *76*, 353–363.
- Assink, E. M. H. (1986). Verkennen kinderen spontaan orthografische regels? *Tijdschrift voor Taalbeheersing*, *8*, 106-118.
- Barry, C., & Seymour, P. (1988). Lexical priming and sound-to-spelling contingency effects in non-word spelling. , *40*, 5-40.
- Berkling, K., & Lavalley, R. (2015). WISE: A web-interface for spelling error recognition for German: A description and evaluation of the underlying algorithm. *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology*, *40A*, 87-96.
- Borgwaldt, S., Hellwig, F., & De Groot, A. (2004). Word-initial entropy in five languages: Letter to sound and sound to letter. *Written Language and Literacy*, *7*, 165–184.
- Borgwaldt, S. R. (2003). *From onset to entropy: Spelling pronunciation patterns in six languages* (Unpublished doctoral dissertation). Universiteit van Amsterdam.
- Bosman, A. (2005). Development of rule-based verb spelling in Dutch students. *Written Language and Literacy*, *8*, 1-18.
- Bosman, A., & Van Orden, G. (1997). Why spelling is more difficult than reading. In C. A. Perfetti, L. Rieben, & M. Fayol (Eds.), *Learning to spell: Research, theory, and practice across languages* (p. 173-194). Hillsdale, New Jersey: Lawrence Erlbaum Associates.
- Cordewener, K. A. H., Verhoeven, L., & Bosman, A. M. T. (2016). Improving spelling performance and spelling consciousness. *The Journal of Experimental Education*, *84*, 48-74.
- Dumond, J. (1985). *Leerstoornissen 1: Theorie en model*. Rotterdam: Lemniscaat.
- Elffers, B., van Bael, C., & Strik, H. (2013). *ADAPT: Algorithm for dynamic alignment of phonetic transcriptions*. Nijmegen, The Netherlands.
- Ellis, A. (1984). *Reading, writing and dyslexia: A cognitive analysis*. London: Lawrence Erlbaum.
- Harward, S. V., Allred, R. A., & Sudweeks, R. R. (1994). The effectiveness of four self-corrected spelling test methods. *Reading Psychology: An International Quarterly*, *15*, 245-271.
- Horbach-Kleijnen, R. (1992). *Hardnekkige spellingfouten: Een taalkundige analyse*. Lisse: Swets & Seitlinger.
- Horbach-Kleijnen, R. (1997). *Strategieën van zwakke lezers en spellers in het voortgezet onderwijs*. Lisse: Swets & Seitlinger.
- Houghton, G., & Zorzi, M. (2003). Normal and impaired spelling in a connectionist dual-route architecture. *Cognitive Neuropsychology*, *20*, 115-62.
- Inspectie van Onderwijs. (2019). *Dyslexieverklaringen: Verschillen tussen scholen nader bekeken*.
- Inspectie van Onderwijs. (2021). *Peil.Schrijfvaardigheid einde (speciaal) basisonderwijs 2018-2019*. Utrecht.
- Kay, J. (1990). Psychological aspects of spelling. In H. Günther, O. Ludwig, & J. Schwitall (Eds.), *Schrift und schriftlichkeit: Writing and its use* (p. 1074- 1093). Berlin: De Gruyter.

- Keuning, J., & Verhoeven, L. (2008). Spelling development throughout the elementary grades: The Dutch case. *Learning and Individual Differences*, 18(4), 459–470.
- Kort, H. (1987). *Spellingpakket deel 1*. Eindhoven: Onderwijsbegeleidingsdienst.
- Laarmann-Quante, R. (2016). Automating multi-level annotations of orthographic properties of German words and children’s spelling errors. *Proceedings of the 2nd Language Teaching, Learning and Technology Workshop*, 14-22.
- Meijerink, H. (2009). *Referentiekader taal en rekenen: De referentieniveaus*. Enschede.
- Nunn, A. (1998). Dutch orthography: A systematic investigation of the spelling of Dutch words. Den Haag: Holland Academic Graphics.
- Ojemann, P. (1970). *Fouten kijken u aan*. Amsterdam: Tor A’dam.
- Perfetti, C. (1985). *Reading ability*. New York: Oxford University Press.
- Perry, C., & Ziegler, J. (2004). Beyond the two-strategy model of skilled spelling: Effects of consistency, grain size, and orthographic redundancy. *Quarterly Journal of Experimental Psychology*, 57A, 325-256.
- Schijf, T. (2009). *Lees- en spellingvaardigheden van brugklassers*. Amsterdam: SCO-Kohnstamm Instituut.
- Schrödel, M., & Trouvain, J. (2003). The German text-to-speech synthesis system MARY: A tool for research, development and teaching. *International Journal of Speech Technology*, 6, 365–377.
- Tellings, A., Oostdijk, N., Monster, I., Grootjen, F., & van den Bosch, A. (2018a). BasiScript: : A corpus of contemporary Dutch texts written by primary school children. *International Journal of Corpus Linguistics*, 23(4), 494–508. doi: <https://doi.org/10.1075/ijcl.17086.tel>
- Tellings, A., Oostdijk, N., Monster, I., Grootjen, F., & van den Bosch, A. (2018b). Spelling errors of 24 cohorts of children across primary school 2012-2015: a BasiScript corpus study. *Computational Linguistics in the Netherlands Journal*, 8, 83-98.
- ten Bosch, L. (n.d.). *Grapheme to phoneme converter*. Retrieved 19.01.2021, from <https://webservices.cls.ru.nl/g2pservice/>
- Van den Bosch, A., Busser, B., Canisius, S., & Daelemans, W. (2007). An efficient memory-based morphosyntactic tagger and parser for Dutch. *Computational linguistics in the Netherlands: Selected papers from the seventeenth CLIN Meeting*, 99-114.
- van der Geest, A., van der Kooij, H., van Rossum, T., Ruijssenaars, W., & Westerbeek, T. (1978). *Spelling met spellingzwakke kinderen*. Den Bosch: Malmberg.
- van Leerdam, M., Bosman, A. M. T., & van Orden, G. C. (1998). The ecology of spelling instruction: Effective training in first grade. In P. Reitsma & L. Verhoeven (Eds.), *Problems and interventions in literacy development* (p. 307-320). Dordrecht: Kluwer Academic Publishers.
- van Gompel, M., & Reynaert, M. (2013). FoLiA: A practical xml format for linguistic annotation: a descriptive and comparative study. *Computational Linguistics in the Netherlands Journal*, 3, 63-81.
- Van Luijn, T. (1992). *Spelling, spraak en fonemische analyse* (Unpublished doctoral dissertation). Rijksuniversiteit Utrecht.
- Verhoeven, G. (1985). *De strategieën van de speller*. Groningen: Wolters Noordhoff.

APPENDIX A COMPUTER PHONETIC ALPHABET CGN2

Table 13 and 14 show which phonemes correspond to each symbol of the computer phonetic alphabet CGN2. This alphabet is used by the Grapheme-to-phoneme converter webservice (ten Bosch, n.d.).

Table 13: Consonants

	IPA	CGN2	Dutch	English
Plosives	p	p	pak	sport
	b	b	bak	bait
	t	t	tak, had	stop
	d	d	dak	duck
	k	k	kap	school
	g	g	goal	goal
Fricatives	f	f	fel	feats
	v	v	vel	very
	s	s	sok	sock
	z	z	zijn	zip
	ʃ	S	show, sjabloon, chef, action	shall
	ʒ	Z	bagage, jury	vision
	ʁ	G	gaan	loch
	x	x	toch, weg, acht	loch
	ɦ	h	hand, had	behind
Sonorants	m	m	met	man
	n	n	net	neck
	ŋ	N	bang	long
	l	l	land	land
	r	r	rand	river
	ʋ	w	wit, wang	wine
	j	j	ja	yard
	ɲ	J	campagne, oranje	canyon

Table 14: Vowels

	IPA	CGN2	Dutch	English
Long	i	i	vier	deep
	e:	e	veer	made
	a:	a	naam	lad
	o:	o	voor	born
	u	u	voer	boot
	y	y	vuur	goose
	ø:	EU	deur	fur
Short	ɪ	I	pit	sit
	ɛ	E	pet	bed
	ɑ	A	pad	father
	ɔ	O	pot	off
	ʏ	U	put	nurse
Schwa	ə	@	gemak	again
Diphthong	ɛi	EI	fijn	may
	œ	UI	huis	house
	ʌu	AU	goud	out
Loan	ɛ:	E2	crème	square
	œ:	O j	freule	fur
	ɔ:	O	roze	dog
Nasal	ɛ̃:	E n	bassin	doyen
	ɑ̃:	A n	ensemble, genre	croissant
	ɔ̃:	O n	compagnon	montage
Other	a:i	a j	draai	prize
	o:i	o j	mooi	boys
	iu	i w	nieuw	new
	yu	y w	duw	few would
	e:u	e w	sneeuw	say oo
	ui	u j	roeiboot	to eternity

APPENDIX B EXAMPLE OF MULTILAYERED ANNOTATED WORD

LAYER	VALUE					
word	Scholen					
target	Scho-len					
original	s-goo1e-					
author	JurC					
gender	j (jongen, boy)					
grade	6					
date	najaar_2014 (autumn 2014)					
theme	ThemaRarewoorden (Theme weird words)					
fileName	d389055.xml					
morphemes	[school, en]					
lemma	school					
pos-tag	N(soort,mv,basis)					
punctEndSentence	False					
capitalBeginSentence	True					
homophones	True					
phon_target	s	x	o	l	@	-
pcus_target	S	ch	o	l	e	n
pcus_original	s	g	oo	l	e	-
error	-	UnSub1	CoVs1	-	-	MoEndN1
error_capital	SyCap1	-	-	-	-	-
basis	Un	Un	CoVs1	Un	Un	MoEndN1
basis_capital	SyCap1	-	-	-	-	-

APPENDIX C TECHNICAL OVERVIEW OF THE ALGORITHM

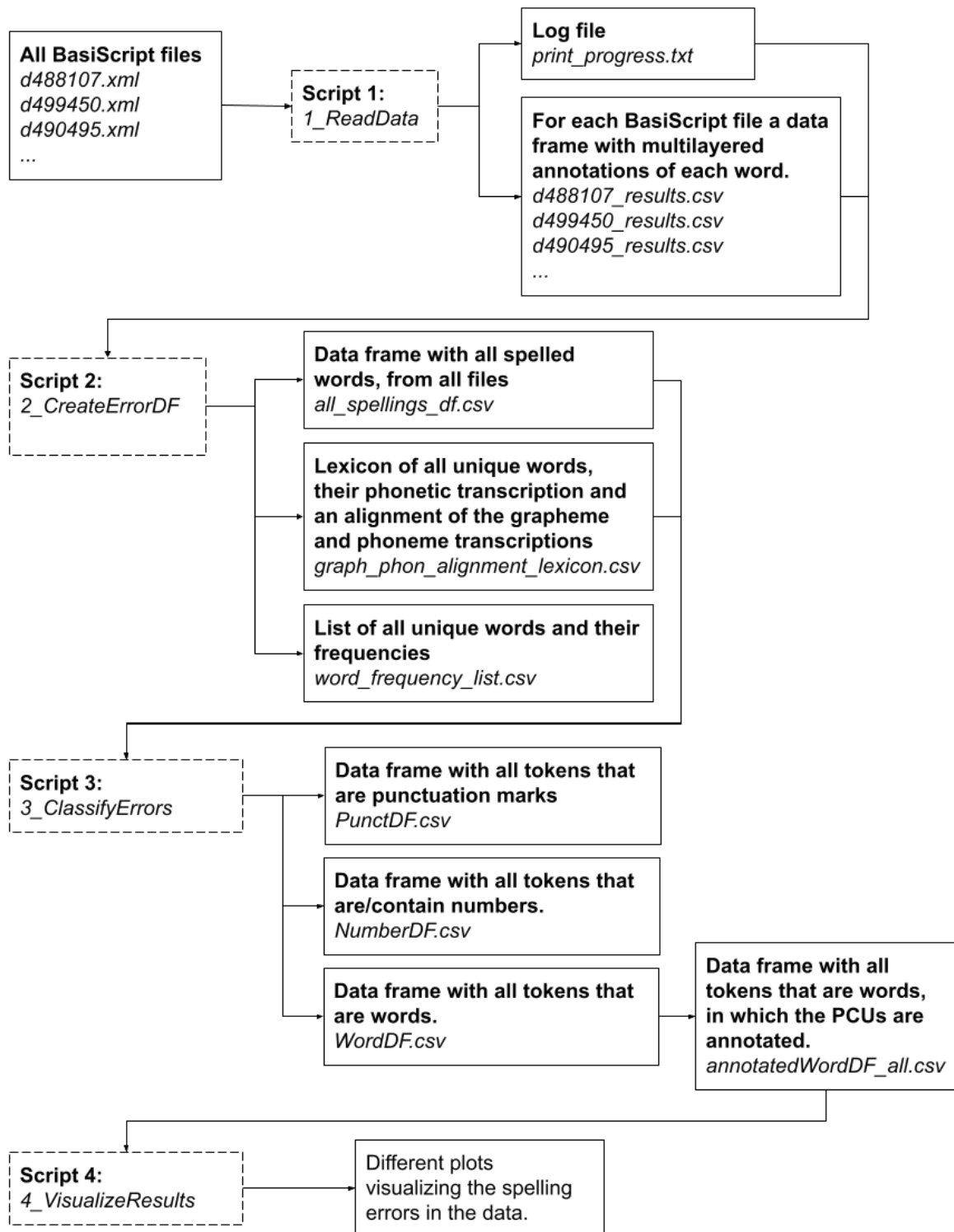


Figure 8: The technical pipeline of the spelling error detection and annotation algorithm. The scripts, input and output files of the whole pipeline can be found on Ponyland in the directory `/vol/ten-users3/wharmsen/BasiScriptErrorAlgorithm`. The scripts are Jupyter Notebook scripts (.ipynb), information about how to open and run these scripts can be found on <https://ponyland.science.ru.nl/>.

APPENDIX D PHONEME-GRAPHEME ALIGNMENT RULES

Table 15: This table represents the phoneme-grapheme alignment rules that are used by the developed spelling error detection and annotation algorithm to align the phonemes of the target word with its graphemes.

Phoneme (CGN2)	Graphemes	Phoneme (CGN2)	Graphemes
a	aa, a, ä, á	n	nn, n, ñ
A	a, e, ä	n j	gn, nh
AU	auw, ouw, au, ou, ow	N	ng, n
b	bb, b	o	eau, au, oo, oi, o, ö, ó
d	dd, d, t	EU	eu
e	eeu, ae, ai, ee, é, e, ë, ay	U	eu, ü
@	ij, e, i, u, ë	UI	eui, ui, uï
E	a, e, i, ë, aï, è	O	o, ö
E2	ae, ai, è, e, aï	p	pp, p
EI	ei, ij, eï	r	rr, r
f	ff, ph, f, v	s	sch, ss, zz, t, z, c, ç, s
g	g, k	S	ch, sh, si, sj, ci, stj
G	gg, g	t	th, tt, t, d
h	h	u	oe, oo, ou, ü, u
i	ieu, ea, ee, ie, i, ij, y, ï, í	v	v, f
l	i, y, ï	w	ww, u, w
j	ill, i, j, y, ï	w A	oi
k	ch, kk, qu, k, c	w l	uï
k s	cc, x	x	ch, g
k w	qu	y	u, uu, ü, ú
l	ll, l	U	u
m	mm, m	z	z, s
		Z	ti, g, j

APPENDIX E ANNOTATION SCHEME

The tables in this appendix give an overview of all spelling principles in the annotation scheme that are used to annotate the target PCUs ("basis") and spelling errors ("error"). Each spelling principles in numbered and an example of an error that is annotated with that principle is provided. Some spelling principles can be subdivided into sub spelling principles. These are written in italics in the overview.

Table 16: Spelling principles from the Unmarked category

	Theme	Name		Description of spelling error	Examples	
		Error	Basis		Target	Original
1	Deletion	UnDel1	Un	Deletion of PCU	straat	straa
2	Insertion	UnIns1	Ins	Insertion of PCU	school	schrool
3	Substitution	UnSub1	Un	Substitution of PCU with another PCU that maps to the same phoneme		
		<i>UnSub1a</i>	<i>Un</i>	<i>A consonant is not doubled if it's written after a long vowel and if it's not at the end of the word</i>	koken	kokken
4	Substitution	<i>UnSub1b</i>	<i>Un</i>	<i>Substitution of PCU with another PCU that maps to the same phoneme, which is not UnSub1a</i>	pauw	pouw
		UnSub2	Un	Substitution of PCU with another PCU that maps to another phoneme.		
		<i>UnSub2a</i>	<i>Un</i>	<i>Target PCU is reversed original PCU.</i>	klein	klien
		<i>UnSub2b</i>	<i>Un</i>	<i>Graphemes of PCU are partly deleted</i>	reus	res
		<i>UnSub2c</i>	<i>Un</i>	<i>Graphemes of PCU are partly inserted</i>	binnen	b u inen
5	Substitution	<i>UnSub2d</i>	<i>Un</i>	<i>Complete substitution of PCU</i>	buiten	boeten
		UnSub3	Un	Substitution involving capital letter PCUs		
		<i>UnSub3a</i>	<i>Un</i>	<i>Every not-first letter of a sentence and non-name is written with a lowercase letter</i>	kat	Kat
		<i>UnSub3b</i>	<i>Un</i>	<i>Substitution of capital letter with lowercase letter that is not SemCap1 or SyCap1</i>	kat	Kat

Table 17: Spelling principles from the Marked by Context category

	Theme	Name		Description of spelling error	Examples	
		Error	Basis		Target	Original
6	Vowel Singulation	CoVs1	CoVs1	A long vowel is written with one vowel symbol if it is at the end of a syllable	ma <u>k</u> en	ma <u>a</u> ken
7		CoVs2	CoVs2	Exceptions: A long vowel is written with two vowel symbols at the end of a syllable in several cases:		
		CoVs2a	CoVs2a	<i>A long vowel is written with two letters if it is followed by "ch" to avoid confusion with the sjwa</i>	g <u>oo</u> chelaar	g <u>o</u> chelaar
		CoVs2b	CoVs2b	<i>A long "ee" is written with two symbols at the end of a word to avoid confusion with the sjwa</i>	z <u>ee</u>	z <u>e</u>
		CoVs2c	CoVs2c	<i>A long "ie" is written with two vowels at the end of a word</i>	drie <u>ie</u>	dri <u>i</u>
		CoVs2d	CoVs2d	<i>A long vowel is written with two vowels before a diminutive suffix</i>	la <u>at</u> je	la <u>at</u> je
8	Consonant Doubling	CoCd1	CoCd1	A consonant is doubled if it is written after a short vowel (excluding sjwa) and if it's not at the end of the word	jo <u>k</u> ken	jo <u>k</u> en
9	Special Cases	CoSc1	CoSc1	The "w" before an "r" is pronounced as a /v/	w <u>r</u> eken	v <u>r</u> eken
10		CoSc2	CoSc2	A /w/ or /j/ pronounced between two vowels is not written	januari, eieren	janu <u>w</u> ari, eij <u>e</u> ren
11		CoSc3	CoSc3	When a "w" is pronounced after an /ee/ or /ie/ sound, an "u" should be written before the "w"	snee <u>uw</u> , nie <u>uw</u>	sneew, nieuw
12	Accents	CoAc1	CoAc1	Some vowels need an accent to simplify pronunciation	Belgi <u>e</u> , café	Belgie, café
13		CoAc2	Un	Most vowels don't need an accent to simplify pronunciation	ka <u>t</u>	ka <u>t</u>
14	Apostrophe	CoAp1	CoAp1	An apostrophe is written in several cases:		
		CoAp1a	CoAp1	<i>An apostrophe is written in proper names that end in an -sis-sound and are used as genitive</i>	Frits'	Frits
		CoAp1b	CoAp1	<i>An apostrophe is written in plural and genitive forms that people could read wrongly with a short vowel instead of long vowel</i>	opa's	opas
		CoAp1c	CoAp1	<i>Apostrophe is written in diminutives of words ending in a consonant + "y"</i>	baby'tje	babytje
		CoAp1d	CoAp1	<i>Apostrophe is written in one symbol words or abbreviations before suffixes of plural and genitive forms</i>	h.b.o.'er sms't	h.b.o.-er sms-t
		CoAp1e	CoAp1	<i>Apostrophe is written instead of other letters to shorten a word</i>	zo'n	zoon/zo-n

Table 18: Spelling principles from the Marked by Morphology category

	Theme	Name		Description of spelling error	Examples	
		Error	Basis		Target	Original
15	Assimilation	MoAs1	MoAs1	Assimilation of stem		
		<i>MoAs1a</i>	<i>MoAs1</i>	<i>Unvoiced consonant is pronounced as voiced</i>	steeds	steets
		<i>MoAs1b</i>	<i>MoAs1</i>	<i>Voiced consonant is pronounced as unvoiced</i>	zeldzame	zeldsame
16	Miniaturization	MoMi1	MoMi1	Miniaturization (Geminaatdelging)	acht <u>t</u> ien	acht <u>i</u> en
17	Ass. and Min.	MoAsMi1	MoAsMi1	Assimilation of stem, followed by miniaturization	op <u>b</u> od	o <u>b</u> od
18	Final Devoicing	MoFd1	MoFd1	Final Devoicing: Voiced obstruent becomes voiceless at end of word		
		<i>MoFd1a</i>	<i>MoFd1</i>	<i>Voiced obstruent "d" is written as "t"</i>	hond	hont
		<i>MoFd1b</i>	<i>MoFd1</i>	<i>Voiced obstruent "b" is written as "p"</i>	club	clup
19	Exception Final Devoicing: Words cannot end in a voiced obstruent	MoFd2	MoFd2	Exception Final Devoicing: Words cannot end in a voiced obstruent		
		<i>MoFd2a</i>	<i>MoFd2</i>	<i>Words cannot end in the voiced obstruent "v"</i>	wer <u>f</u>	were <u>v</u>
		<i>MoFd2b</i>	<i>MoFd2</i>	<i>Words cannot end in the voiced obstruent "z"</i>	muis	muiz
20	Silent "t"	MoEndT1	MoEndT1	t is written, but /t/ is not pronounced	kastje rech <u>t</u> door	kasje rechdoor
21	Silent "n"	MoEndN1	MoEndN1	n is written, but /n/ is not pronounced	binne <u>n</u> fietsen	binne fietse
22	Composition words	MoCoS1	MoCoS1	You write a "between s" between two parts of a composition word if you hear that "s" or when you hear it in similar words	dorps <u>w</u> eg, dorps <u>s</u> traat	dorpweg, dorpstraat
23		MoCoS2	MoCoS2	If you don't hear a "s" between two parts of a composition word, you don't write one	hoofdweg	hoofds <u>w</u> eg
24	Hyphen	MoHy1	MoHy1	Use a hyphen ...		
		<i>MoHy1a</i>	<i>MoHy1</i>	<i>... in case of word repetition</i>	zon_ en feestdagen	zon en feestdagen
		<i>MoHy1b</i>	<i>MoHy1</i>	<i>... in case of word repetition</i>	abc_boek	abc boek
		<i>MoHy1c</i>	<i>MoHy1</i>	<i>... in names</i>	Gert_Jan	Gert Jan
		<i>MoHy1d</i>	<i>MoHy1</i>	<i>... in case of vowel collision</i>	zonne_ energie	zonneenergie
		<i>MoHy1e</i>	<i>MoHy1</i>	<i>... in other categories</i>	-	-

Table 19: Spelling principles from the Marked by Syntax category (1/2)

	Theme	Name		Description of spelling error	Examples	
		Error	Basis		Target	Original
25	Number	SyNum1	SyNum1	Plural noun forms end in "s" or "n"	bureau <u>s</u> kante <u>n</u>	bureau kanter
26		<i>SyNum1a</i> <i>SyNum1b</i>	<i>SyNum1</i> <i>SyNum1</i>	<i>Plural noun forms end in "s"</i> <i>Plural noun forms end in "n"</i>		
		SyNum2	SyNum2	Plural verb forms end in -n (pv tt, pv vt, vd, od, inf)	fietsen <u>n</u>	fiets <u>e</u> r
27	Sjwa	SySjwa1	SySjwa1	Sjwa in non-verb suffixes: Some suffixes consist of an "e" (sjwa), or "e" (sjwa) + "n"	hele <u>e</u>	hel
28		SySjwa2	SySjwa2	Some suffixes have an "e" (sjwa) in second position after the "d" or "t"	dansende <u>e</u>	dansend
29	Spaces	SySp1	Ins	Write words as much as possible together	basgitaar bezoek	bas gitaar be zoek
		<i>SySp1a</i>	<i>Ins</i>	<i>Write composition words that exists of two or more nouns as much as possible together</i>		
		<i>SySp1b</i>	<i>Ins</i>	<i>Write non-composition words also as much as possible together</i>		
30	Composition words	SyCoN1	SyCoN1	Between -n needs to be written between two morphemes in some composition words	bijen <u>n</u> korf	bijekorf
31	Person	SyPer1	SyPer1	Present 2nd/3th person singular forms have suffix "t"	loopt <u>t</u>	loop
		<i>SyPer1a</i>	<i>SyPer1</i>	<i>"t" substituted with "d"</i>	loopt <u>d</u>	loop
		<i>SyPer1b</i>	<i>SyPer1</i>	<i>"t" substituted with another character, which is not a "d"</i>	loopt <u>v</u> ind <u>t</u>	loop vinds
32	Past Simple	SyVt1	SyVt1	When the stem of a verb ends in an unvoiced sound (the sounds in "'t kofschip"), the suffix starts with an unvoiced /t/ sound ("t")	werk <u>t</u> e werk <u>t</u> e wach <u>t</u> t <u>e</u> werk <u>t</u> e wach <u>t</u> t <u>e</u> n	werk <u>d</u> e werk <u>t</u> t <u>e</u> wach <u>t</u> e werk <u>s</u> e wach <u>k</u> e <u>n</u>
		<i>SyVt1a</i>	<i>SyVt1</i>	<i>"t" substituted by "d"</i>		
		<i>SyVt1b</i>	<i>SyVt1</i>	<i>"t" substituted by "tt"</i>		
		<i>SyVt1c</i>	<i>SyVt1</i>	<i>"tt" substituted by "t"</i>		
		<i>SyVt1d</i>	<i>SyVt1</i>	<i>"t" or "tt" substituted by PCU other than "d", "tt" or "t"</i>		
33		SyVt2	SyVt2	When the stem of a verb ends in an voiced sound (the sounds not in "'t kofschip"), the suffix starts with a voiced /d/ sound ("d")	krab <u>d</u> e krab <u>d</u> e brand <u>d</u> e krab <u>d</u> e brand <u>d</u> e	krab <u>t</u> e krab <u>d</u> d <u>e</u> brand <u>e</u> krab <u>m</u> e bran <u>k</u> e
	<i>SyVt2a</i>	<i>SyVt2</i>	<i>"d" substituted by "t"</i>			
	<i>SyVt2b</i>	<i>SyVt2</i>	<i>"d" substituted by "dd"</i>			
	<i>SyVt2c</i>	<i>SyVt2</i>	<i>"dd" substituted by "d"</i>			
		<i>SyVt2d</i>	<i>SyVt2</i>	<i>"d" or "dd" substituted by other PCU than "t", "dd" or "d"</i>		

Table 20: Spelling principles from the Marked by Syntax category (2/2)

	Theme	Name		Description of spelling error	Examples		
		Error	Basis		Target	Original	
34	Past Participle	SyVd1	SyVd1	If last letter of word stem in "'t kofschip", suffix starts with "t"	gepakt	gepak <u>d</u>	
		<i>SyVd1a</i>	<i>SyVd1</i>				<i>"t" substituted by "d"</i>
		<i>SyVd1b</i>	<i>SyVd1</i>				<i>"t" substituted by other character, which is not a "d"</i>
35		SyVd2	SyVd2	If last letter of word stem not in "'t kofschip", suffix starts with "d"	beloof <u>d</u>	beloof <u>t</u>	
		<i>SyVd2a</i>	<i>SyVd2</i>				<i>"d" substituted by "t"</i>
		<i>SyVd2b</i>	<i>SyVd2</i>				<i>"d" substituted by other character, which is not a "t"</i>
36		SyVd3	SyVd3	Some part participles end in -en	geroepen	geroept	
37	Present Participle	SyOd1	SyOd1	The suffix of present participle always starts with "d"	dansend	dansent	
		<i>SyOd1a</i>	<i>SyOd1</i>				<i>"d" substituted by "t"</i>
		<i>SyOd1b</i>	<i>SyOd1</i>				<i>"d" substituted by other character, which is not a "t"</i>
39	Capital	SyCap1	SyCap1	Every sentence starts with a capital letter	Hallo	hallo	

Table 21: Spelling principle from the Marked by Semantics category

	Theme	Name		Description of spelling error	Examples	
		Error	Basis		Target	Original
40	Capital	SemCap1	SemCap1	Every proper name, title and some abbreviations start with a capital letter	Nijmegen	nijmegen

APPENDIX F PRECISION SCORES

Table 22: Precision Table 1/3

Category	Name	Description of spelling error	Precision	
			Error	Basis
Vowel Singulation	CoVs1	A long vowel is written with one vowel symbol if it is at the end of a syllable	1.0	1.0
	CoVs2	Exceptions: A long vowel is written with two vowel symbols at the end of a syllable in several cases:	1.0	1.0
Consonant Doubling	CoCd1	A consonant is doubled if it is written after a short vowel (excluding sjwa) and if it's not at the end of the word	1.0	1.0
Special Cases	CoSc1	The "w" before an "r" is pronounced as a /v/	1.0	1.0
	CoSc2	A /w/ or /j/ pronounced between two vowels is not written	1.0	1.0
	CoSc3	When a "w" is pronounced after an /ee/ or /ie/ sound, an "u" should be written before the "w"	1.0	1.0
* Accents	CoAc1	Some vowels need an accent to simplify pronunciation	1.0	1.0
	CoAc2	Most vowels don't need an accent to simplify pronunciation	1.0	-
Apostrophe	CoAp1	An apostrophe is written in several cases:	1.0	1.0
Assimilation	MoAs1	Assimilation of stem	0.6	0.55
Miniaturization	MoMi1	Miniaturization (Geminaatdelging)	0.95	0.70
Ass. and Min.	MoAsMi1	Assimilation of stem, followed by miniaturization	0.8	1.0
Final Devoicing	MoFd1	Final Devoicing: Voiced obstruent becomes voiceless at end of word	1.0	1.0
	MoFd2	Exception Final Devoicing: Words cannot end in a voiced obstruent	1.0	1.0
Silent "t"	MoEndT1	t is written, but /t/ is not pronounced	1.0	1.0
Silent "n"	MoEndN1	n is written, but /n/ is not pronounced	1.0	1.0

Table 23: Precision Table 2/3

Category	Name	Description of spelling error	Precision	
			Error	Basis
Composition words	MoCoS1	You write a "between s" between two parts of a composition word if you hear that "s" or when you hear it in similar words	0.95	0.95
	MoCoS2	If you don't hear a "s" between two parts of a composition word, you don't write one	0.75	0,75
Hyphen	MoHy1	Use a hyphen in case of word repetition, in composition words containing abbreviations, numbers, single letters and special characters, in names and in case of vowel collision	1.0	1.0
Number	SyNum1	Plural noun forms end in -s or -n	1.0	1.0
Sjwa	SySjwa1	Sjwa in non-verb suffixes: Some suffixes consist of an "e" (sjwa), or "e" (sjwa) + "n"	1.0	1.0
Spaces	SySp1	Write words as much as possible together	1.0	1.0
Composition words	SyCoN1	Between -n needs to be written between two morphemes in some composition words	0.8	0.2
Number	SyNum2	Plural verb forms end in -n (pv tt, pv vt, vd, od, inf)	1.0	1.0
Person	SyPer1	Present 2nd/3th person singular forms have suffix "t"	1.0	1.0
Past Simple	SyVt1	When the stem of a verb ends in an unvoiced sound (the sounds in "'t kofschip"), the suffix starts with an unvoiced /t/ sound ("t")	1.0	1.0
	SyVt2	When the stem of a verb ends in a voiced sound (the sounds not in "'t kofschip"), the suffix starts with a voiced /d/ sound ("d")	1.0	1.0
Past Participle	SyVd1	If last letter of word stem in "'t kofschip", suffix starts with "t"	1.0	1.0
	SyVd2	If last letter of word stem not in "'t kofschip", suffix starts with "d"	1.0	1.0
	SyVd3	Some part participles end in -en	1.0	1.0
Present Participle	SyOd1	The suffix of present participle always starts with "d"	1.0	1.0
Sjwa	SySjwa2	Some suffixes have an "e" (sjwa) in second position after the "d" or "t"	1.0	1.0
Semantics	SemCap1	Every proper name, title and some abbreviations start with a capital letter	1.0	1.0

Table 24: Precision Table 3/3

Category	Name	Description of spelling error	Precision	
			Error	Basis
Deletion	UnDel1	Deletion of PCU	1.0	-
Insertion	UnIns1	Insertion of PCU	1.0	-
* Substitution	UnSub1	Substitution of PCU with another PCU that maps to the same phoneme.	1.0	-
	UnSub2	Substitution of PCU with another PCU that maps to another phoneme.	1.0	-
	UnSub3	Substitution involving capital letter PCUs	1.0	-
To mark target PCUs	Un	Only phoneme-grapheme conversion rule is necessary for correct spelling.	-	1.0