

Online analysis of children handwritten words in dictation context

Omar Krichen, Simon Corbillé, Eric Anquetil, Nathalie Girard, and Pauline Nerdeux

Univ Rennes, CNRS, IRISA,F-35000 Rennes, France `firstName.lastName@irisa.fr`

Abstract. This paper presents a method for fine analysis of children handwriting on pen-based tablets. This work is in the context of the P2IA project, funded by the French government, which aims at designing a virtual notebook in order to foster handwriting learning for primary school pupils. In this work, we consider the task of analysing handwritten words in the context of a dictation exercise. This task is complex due to different factors: the children do not master yet the morphological aspects of handwriting, nor do they master orthography or translating phonetic sounds to actual graphemes (parts of word). In order to tackle this problem, we extend **to the context of dictation exercises** an analysis engine that was developed previously to deal with copying exercises. Two strategies were developed, the first one is a baseline approach and relies on double child input: the pupil types the word on a virtual keyboard after writing it with the stylus, thus the prior knowledge of the written word will drive the engine analysis. The second one relies on a single input: the child handwritten strokes. To drive the analysis, the strategy consists in generating hypotheses that are phonetically similar to the dictated instruction, which will act as probable approximations of the written word (sequence of letters), to cover potential orthographic mistakes by the pupil. To assist the learning process of the pupils, the engine returns different types of real-time feedbacks, that depend on the confidence of the analysis process (confident assessment on errors, warning, or reject).

Keywords: Handwriting recognition · Online handwriting · Digital learning

1 Introduction

This work is part of an Innovation and Artificial Intelligence Partnership (P2IA¹), which supports the construction of solutions serving fundamental learning in French and mathematics in cycle 2 (CP, CE1, CE2). Here we are interested in defining a solution to help elementary school students in learning spelling. We propose a method to automatically analyse the production of handwritten words, written on pen tablets, in the context of dictation exercises.

¹ <https://eduscol.education.fr/1911/parteneriat-d-innovation-et-intelligence-artificielle-p2ia>

The pedagogical foundation of this work lies in several studies that demonstrate the positive impact of using educational systems in the classroom, especially by using pen based tablets. In a critical review study [10], the authors reported that, among 12 highly trustworthy studies, 9 observed positive learning outcomes for the pupils, whereas 3 observed no difference in learning outcomes between the tablet setup and traditional pen and paper one. Moreover, the authors of [9] demonstrate that providing prompt feedback, which is facilitated with digital tools usage, is a key factor in improving learning performance.

In this context, defining a solution to help learning spelling involves different tasks. The first is to recognise children’s handwritten words, and the second is to understand potential mistakes to give children appropriate feedback for each type of mistake.

The task of children handwritten words analysis is an open challenge. Indeed, even if deep learning based methods have made great progress in handwritten words recognition [13,14], most of them are targeted to adult data, and not suited to cope with children distorted handwriting. This work is an extension of previous project called IntuiScript, where the objective was to help preschool children learn to write. To achieve this goal, copy exercises have been designed, where the word instruction is displayed to the child who must reproduce it on the interface. The scientific challenge was to finely analyse the *handwriting quality*, in terms of letters shape, direction and order, in order to provide children with feedback on improving their writing skills. As the word instruction was displayed, it served as prior knowledge (groundtruth) to guide the process of recognising and analysing the child’s written words and thus limit misinterpretation. The challenge was to deal with the degraded nature of handwriting (incorrect letter shapes). This previous work, with positive pedagogical results presented in [1], was transferred to "Learn & Go company" and integrated into the "Kaligo" solution, now used in French and English schools. Since this handwriting analysis method is based on knowledge of the word instruction and its display to the child, it is not robust enough to reliably extract letter-level segmentation when the child does not write the expected letters, which happens when the word is dictated (without display). As an extension of IntuiScript, the pedagogical objective of the P2IA project is to help children acquire orthographic knowledge, *i.e.* to learn graphemes and phonemes. The target population is primary school children who have acquired prior handwriting skills and, dictation/spelling exercises are proposed, such as the dictated instruction is heard but not seen by the child. The scientific challenge is to design an intelligent tutoring system [12] that is able to **provide orthographic feedback** to the child.

In a dictation context, the analysis task is more complex since the engine does not know what the child has written. We are faced with orthographic and phonetic errors, since the child only hears the instruction, as well as morphological errors. For clarity purposes, we define three important notions that will be present all along this paper:

- The instruction: the dictated word that the child has to correctly spell/reproduce;

- The handwriting/handwritten strokes: what the child actually wrote using the tablet stylus;
- The groundtruth: the letters sequence corresponding to the child production.

Figures 1, 2, 3 illustrates examples of the errors we encounter in this context. The instruction (or dictated word) is written in the box with a black border.

The figure 4 illustrates an example of the wide variety of pupils orthographic and morphological errors when the instruction "mes" (my in French) is dictated. As a consequence, the groundtruth is not available since its is likely to be unrelated to the dictated instruction.

To address this challenge, **we propose a phonetic and morphological analysis strategy in a dictation context**. This strategy is based on the extension of the IntuiScript existing analysis engine suited to the copy context. This approach will be based on two independent modes:

- Double input strategy: in addition to writing the dictated word with the pen, the child also uses the tablet virtual keyboard to reproduce his/her production. This straightforward strategy provides the analysis engine with the necessary groundtruth prior knowledge to interpret the word correctly. This represents an intermediate solution that allows us to have an ideal baseline for the engine performance.
- Single input, phonetic hypotheses generation strategy: in order to be free from the user typed input and to cover the eventual errors made by the pupils, we integrate a phonetic engine, which role is to generate, given an



Fig. 1. Morphological error on the shape of "o", orthographic/phonetic error (missing "s" at the end of the word)



Fig. 2. Morphological error on the shape of the letter "d", orthographic error : substitution of "eux" by "e"



Fig. 3. Orthographic errors : substitution of "c" by "qu", likewise for "mm" and "m"

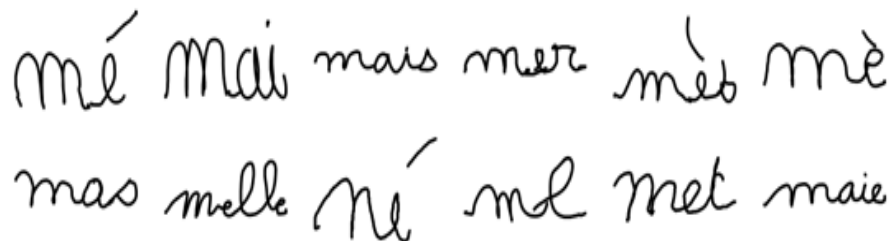


Fig. 4. Examples of pupils errors for the word "mes"

instruction, phonetically similar pseudo-words (same sound as the instruction). Guided by these hypotheses, the analysis engine will try to predict the *actual* groundtruth.

As children are in a learning process, one of the challenges is to provide relevant feedback in real time to the child and to be as precise and exact as possible. So we need to minimise feedback errors, but we also need to moderate the details of the comments with the confidence of the analysis. Therefore, **we define a moderation strategy** (feedback generation mechanism), **based on the confidence of the analysis engine**. We evaluate these contributions on children data, 1087 words collected in the classroom.

The paper is organised as follows. The existing analysis engine is presented in Sec. 2. Sec. 3 describes the engine extension and adaption to the dictation context, whereas Sec. 4 illustrates the typology of the generated feedback. Experiments are presented in Sec. 5. Conclusion and future works are given in Sec. 6.

2 Existing copying analysis engine

In this section, we present the main principles of the existing analysis engine, that was designed for a copy context. The figure 5 illustrates the analysis workflow.

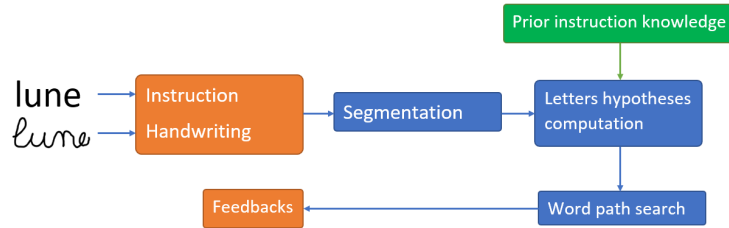


Fig. 5. Workflow of analysis engine in copy context

As discussed earlier in the paper, the inputs of the engine are the instruction and the pupil handwriting. The word analysis is divided into multiple steps as follows.

2.1 Segmentation

The segmentation process is based on two steps. First, the online signal is segmented into primary element by extracting all possible cutting points around the significant descending areas [2]. Second, as illustrated in Figure 6 for the word "juste", a **segmentation lattice** is constructed. The first level of the lattice/graph is built from the primary segmentation by associating ascending areas

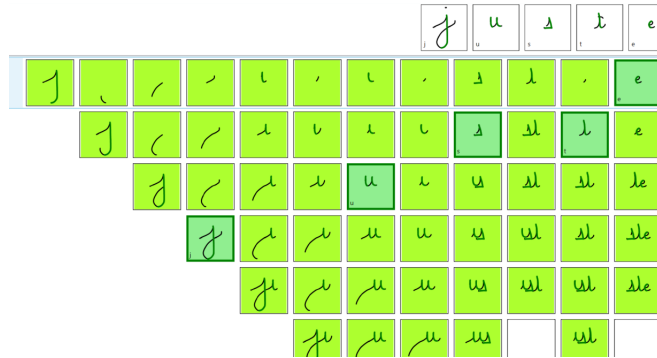


Fig. 6. Segmentation graph for the word "juste"

with a descending area. The second level is made by merging two nodes from the first level and so on for the next levels. Then, the goal is to find the path in this lattice which corresponds to the character decomposition of the word written by the pupil. In Figure 6, the best path is highlighted in green, whereas the explicit segmentation result is in the right top corner of the figure. This explicit segmentation is needed to analyse the letters in context, and provide precise feedback. This is not possible with current deep neural network approaches.

More details regarding to the construction of the segmentation lattice can be found in [3].

2.2 Letter hypothesis computation

The next step is to compute letter hypotheses for each node of the segmentation lattice. First of all, a **recognition score** is computed with Evolve classifier [4], based on fuzzy inference [5]. These letter hypotheses are filtered in order to keep only the ones corresponding to expected letters in the instruction, as well as the ones with the best scores. This is the **recognition step**. Afterwards, an **analysis score** is computed for each hypothesis with a confidence based classifier [6], which use an intra and inter class scoring to deal with confusion between letters. Only the *n best* letters hypotheses belonging to the expected word and whose score is superior to a defined threshold are selected as valid hypotheses and the other ones are discarded. If there is no letter hypothesis verifying these two conditions, the letter with the best recognition score is kept as the sole valid hypothesis of the node. This is the **analysis step**.

This two-step process allows the information contained in the instruction to guide the selection of letter hypotheses. However, this approach is only suited if the pupil does not commit any orthographic error. That is why **this approach is suited to a copy context**, where the child sees the instruction to be reproduced. In a dictation context, this approach has to be adapted to the fact that the child does not know necessarily the instruction spelling.

2.3 Best segmentation path search

The computation of the letters hypotheses is the entry point of the **word segmentation paths** search within the segmentation lattice. A word-level analysis score is calculated using the filtering of the node-level analysis scores for each possible path. This is combined with an n-gram and a spatial coherence scores. The **n-gram** score is related to the presence of bigram or letters of the word instruction. The **spatial coherence score** is calculated with the letters hypotheses of the paths, models of character and the handwriting. That makes it possible to check the consistency between a hypothesis and its real size. For more details, see [3]. This score (analysis, n-gram and spatial coherence) provides a metric on the writing quality, which is used as a reward feedback for the children in previous works [1].

The final path of the segmentation lattice chosen by the analyser as the recognised word is the path that minimises the edit distance with the instruction. The edit distance considered is a **Demerau Leveinshtein edit distance** [7], with optimised edition costs learned from the letters analysis (to deal with the confusion errors of the recognition and analysis process). This best path is not necessarily the one that maximises the analysis score. This way of retrieving the best path is well suited to the **copy context**, but becomes obsolete when the prior knowledge of groundtruth becomes unknown in the dictation context.

In this section, we have presented the principles of the analysis engine, specifically how the knowledge of the expected word, in a copy context, guides the analysis process. In the next section, we will present our new contribution to adapt this engine to a dictation context.

3 Adaption of the engine in a dictation context

Since we are in a dictation context, the impact of the instruction in the guidance of the analysis process becomes obsolete when the child makes orthographic errors. To deal with this problem, we have designed two strategies that allow the engine to overcome this new challenge.

3.1 Double input, baseline strategy

The first strategy to adapt the engine to this new context is a straightforward one: after the completion of the handwritten production, the system asks the child to enter with the keyboard, what he has written. As a consequence, the engine has an explicit knowledge of the groundtruth, which is the *childtyping* (what the child typed). To illustrate the key role of this prior knowledge, figures 7 and 8 present the analysis results of a pupil's handwritten word in two modes: using only the instruction as prior knowledge, and using the childtyping as prior knowledge. The difference in the two modes lies in the analysis results of the "a" node. Since the instruction is to write the word *mes*, the letters "m", "e", and "s" will guide the analysis process in the letter hypotheses computation step.



Fig. 7. Analysis driven by the Instruction (mes), recognised: mei



Fig. 8. Analysis driven by Childtyping (mai), recognised: mai

Even if the letter "a" is the best ranked hypothesis for the highlighted node, it will be discarded due to the analysis filters and another hypothesis, "e", will be considered. It is clear that using the explicit groundtruth as prior knowledge solves this problem.

This intermediary strategy provides a baseline of how the engine would perform in ideal conditions. Surprisingly, the teachers associated to the project estimated that asking the child to type what he wrote could also have pedagogical benefits.

3.2 Phonetic hypotheses generation strategy

The second strategy aims to be free from the user defined groundtruth and to predict it given only the instruction and the handwritten strokes. It is based on the integration of a phonetic hypotheses generation module to the analysis workflow. This module is based on the **Phonetisaurus engine** [8], a stochastic Grapheme to Phoneme (G2P) WFST (Weighted Finite State Transducer). This WFST is based on the principle of joint sequences [11] to align graphemes sequences with their corresponding phonemes in the learning phase. An N-Gram model is generated from the aligned joint sequences and transformed into a WFST model. This G2P model is then able to predict the pronunciation of a new word. To be adapted to our problematic, the G2P model is combined with a P2G (Phoneme to Grapheme) model so that the output of the combined model, given a new word, is a set of phonetically similar pseudo-words. We choose to generate, for each instruction, the 50 best hypotheses according to the Phonetisaurus engine ranking (for more details, see [8]). The figure 9 illustrates the new **phonetic analysis chain**.

A) Phonetic hypotheses as prior knowledge in the analysis process

The first adaptation of the analysis chain lies in modifying the filters that are used for computing the letter hypotheses. We have explained in the last section that for each segmentation node, only the letters that belong to the instruction/groundtruth are kept as valid hypotheses in the node.

Since the groundtruth is unknown, the new analysis chain is guided by all the generated phonetic hypotheses from the instruction (dictated word).

For the example illustrated in figures 7 and 8, instead of having "mes" (the instruction) or "mai" (the child-typed groundtruth) as a the expected sequence, the engine will have {mes, mais, mai, met, med...}. This means that neither "e"

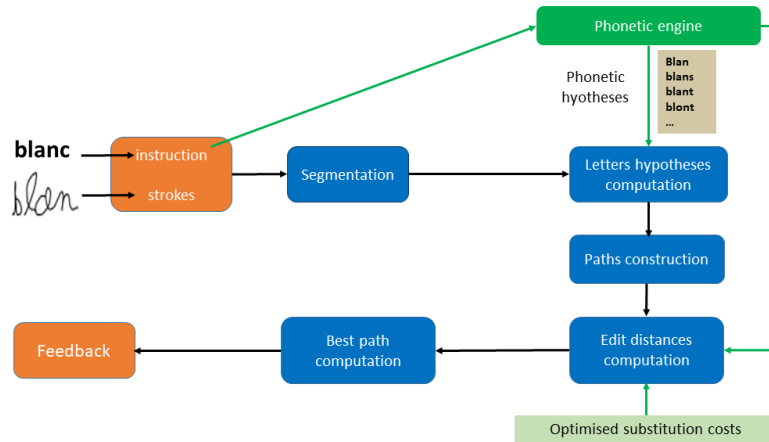


Fig. 9. Phonetic analysis chain workflow

nor "a" will be discarded *a priori* by the analysis filters. The filter criteria is that the letter belongs to one of the expected sequences and that the analysis score of the letter is amongst the n best scores. Another impact of having these phonetic hypotheses guiding the analysis is that other hypotheses will be taken into account, such as the letter "d", as shown in the figure 10. As a consequence, "e" here is discarded since it is not among the n best hypotheses.

The word paths construction is the same as in the basic analysis chain. The difference is in the computation of the n -gram scores of each path, since all bi-grams of all phonetic hypothesis are included in the computation of this score.

B) Phonetic hypotheses as best path decision criteria We have presented in section 2 the edition distance computation between the analyser word hypotheses, and the instruction/groundtruth, that enables the analysis engine to retrieve the handwritten word. This decision criterion is still suited to the *double input strategy*, since the child types the written word on the keyboard. However, this strategy **becomes obsolete without the prior knowledge of the groundtruth**. To tackle this problem, we compute the *phonetic correspondence* of each word segmentation path generated by the engine. By phonetic correspondence, we mean the phonetic hypothesis that has the minimum edition

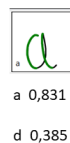


Fig. 10. Analysis scores of the "a" segmentation node

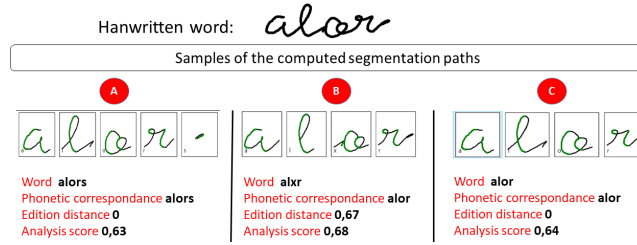


Fig. 11. Segmentation paths with their phonetic correspondences

distance with the segmentation path. The edition score of each path will then depend on the generated phonetic hypotheses, as well as the optimised letter substitution costs learned by the analyser. As an example of this process, in Figure 11, we can see that two segmentation paths (A and C in the figure) have a minimal edition distance score (0), since they are equal to two phonetic hypotheses, with "alors" being the dictated word, *i.e.* the instruction. We can also see that the edition score of "alxr" (B in the figure) is equal to 0.67, this number represents the optimised substitution cost of "o" with "x". This is an interesting example since there is a path that corresponds to the instruction (alors). However the groundtruth is equal to "alor", which means that the pupil made an orthographic mistake. It is clear that if the engine relied on the instruction to guide the interpretation process, it would have made an analysis error by choosing "alors" as the best path.

In this particular case, the edition distance is not sufficient since we have two competing paths having the same edition score. To solve this issue, we include the analysis score in the decision criteria since this score reflects of the handwriting quality and enables the engine to discriminate between competing paths. **The phonetic analysis score is defined as follows.**

$$Score(path) = \frac{1}{1 + |EditScore(Path)|} * 0.3 + analysisScore(path) * 0.7$$

With this phonetic analysis score, the best path returned by the engine corresponds, in this example, to the groundtruth "alor" since the third path (C in the figure) has the highest analysis score of the two competing paths.

The phonetic correspondence is also important because it reflects the confidence of the phonetic analysis engine. Since there is no prior knowledge of the groundtruth, this phonetic correspondence is a relatively efficient approximation.

C) Optimisation of the phonetic analysis chain Using all the letters from the phonetic hypotheses as prior knowledge in the letter hypotheses computation step enables to consider more possibilities in the path construction step, compared to the restrictive aspect of the filters in the copying analysis engine or with the double input strategy. However, one drawback of this phonetic strategy

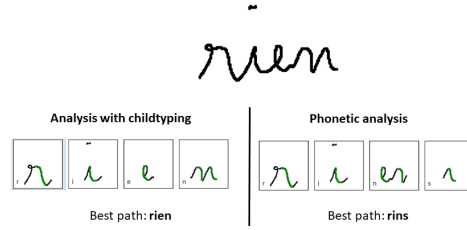


Fig. 12. Degradation of the analysis performance by adding phonetic hypotheses

is that the correct letter hypothesis can sometimes be skipped if favour of others, as shown in the example in figure 12. For the word "rien", when all the phonetic hypotheses letters are added to the analysis filters, the analysis engine is not able to retrieve the groundtruth, whereas it is found when the analysis filters are restricted to "r", "i", "e", and "n".

To tackle this issue, we integrate the notion of **analysis competition**, illustrated in figure 13. This competition is between two analysis instances in order to find the best path, such as the first analysis (basic analysis in the figure) is guided by the instruction, and the second one (phonetic analysis) is guided by the phonetic hypotheses related to the expected word. If the basic analysis best path is equal to the expected word, its phonetic analysis score is computed and compared to the phonetic analysis score of the phonetic analysis best path. The path with the highest score is chosen as the final analysis result. If in the basic analysis best path is different from the expected path, the phonetic analysis best path is considered as the final result. This process enables the engine to retrieve a portion of the correctly written words that were misinterpreted before. We will study the impact of this optimisation in section 5.

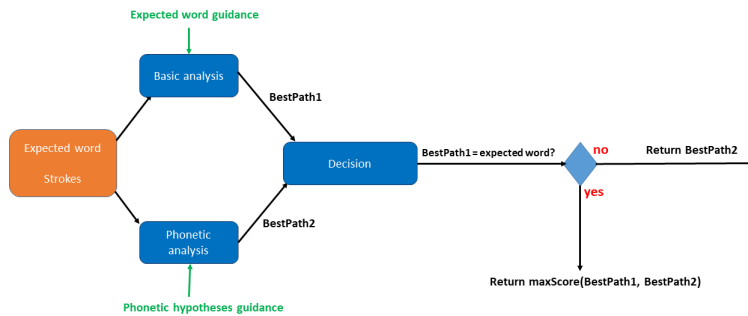


Fig. 13. Competition between analysis instances

After having presented our two strategies related to the orthographic analysis of pupils handwritten words, we present in the next section the feedback generation approach.

4 Feedback typology

The objective of feedback generation by the system is to make the pupil aware of his/her orthographic errors. They are related to the difference between what was dictated and what was written. Therefore, these feedbacks highlight, in the pupil handwriting, the eventual insertions, deletions, substitutions of letters and accents. Table 1 illustrates some examples of such generated feedbacks. The red colour highlights a wrong insertion, whereas orange highlights a substitution.

Table 1. Feedback examples for dictated words

Instruction	Recognized	Error	Feedback
belle	belle	none	
alors	allore	insertion substitution	
céréale	cerèâle	Accents	

- **High confidence:** the recognised word is equal to the childtyping if the first strategy is activated, or is equal to its phonetic correspondence if the phonetic analysis strategy is activated \implies the feedbacks are returned with a high degree of confidence. The feedbacks in table 1 are *high confidence feedbacks*.
- **Medium confidence:** there is one letter that distinguishes between the recognised word and the childtyping/phonetic correspondence \implies the system generates a warning on the uncertain zone/letter and other feedbacks are returned with a lesser confidence degree.
- **Reject:** there is more than one letter that distinguishes the recognised word and the childtyping/phonetic correspondence, no fine feedback is given to the child: the system informs the pupil that it was not able to analyse the production.

Table 3 presents two examples of medium confidence feedback and reject. For the first example, the engine recognises *alard* instead of the groundtruth *alord*, while the instruction is *alors*. The blue feedback on the "o" corresponds to a warning directed to the pupil, whereas the substitution of "s" by "d" is highlighted in grey, since there is a lesser degree of confidence in this feedback. For the second example, the phonetic analysis engine recognised *lemjeur*, which is completely unrelated phonetically to the instruction *bonjour*, or any of its phonetic hypotheses. Therefore, the system does not provide any fine feedback.

Table 2. Examples of "medium confidence" feedback and reject

Dictated	Ground truth	Recognized	Confidence	Feedback
alors	alord	alard	Medium	<i>alord</i>
bonjour	lezour	lmjeur	reject	<i>lezour</i>

In this section, we have presented the feedbacks typology and our defined strategy to cope with analysis uncertainty. We present in the next section the performance of the analysis engine, as well as the pertinence of the generated feedback to the children.

5 Results

In this section, we will base our evaluation of the dictation adapted engine on a data-set of 1078 pseudo-words collected from children that use the system in the classroom. Due to GDPR restrictions on children's private data, we are not yet able to share this dataset publicly today. Table 3 presents some samples of this data-set enriched by the engine feedbacks.

5.1 Analysis results

Table 4 presents the performance of each analysis mode/strategy on the test set. By analysis performance, we mean the correct segmentation and recognition of the handwritten strokes that lead automatically to a correct feedback. The fact that the ground truth is already available in the analysis with double input

Table 3. Feedback examples for dictated words: green (H) → correct high confidence feedback, green (M) → correct medium confident feedback, red (H) → error in high confidence feedback, blue (R) → feedback reject

Instruction	Child handwriting, feedback								
Partir	#1-6 partir <i>partir</i> H	#2-6 partir <i>partir</i> H	#3-7 partir <i>partir</i> H	#4-7 partir <i>partir</i> H	#5-8 partir <i>partir</i> R	#6-6 partir <i>partir</i> R	#7-5 partir <i>partir</i> M	#8-7 partir <i>partir</i> H	#9-6 partir <i>partir</i> H
Croire	#0-5 croire <i>croire</i> M	#1-4 croire <i>croire</i> M	#3-4 croire <i>croir</i> H	#4-6 croire <i>croir</i> H	#5-5 croire <i>croir</i> H	#6-4 croire <i>Croir</i> M	#2-3 croire <i>croir</i> H	#7-5 croire <i>quoir</i> H	#8-3 croire <i>cowar</i> M
Mes	#1-4 mes <i>mai</i> H	#2-5 mes <i>mai</i> H	#14-3 mes <i>mas</i> H	#15-1 mes <i>melle</i> H	#24-2 mes <i>me</i> H	#44-5 mes <i>mel</i> H	#45-3 mes <i>mais</i> H	#46-3 mes <i>mai</i> H	#47-3 mes <i>ml</i> H

Table 4. Analysis performance of each strategy

Approach	Correctly analysed	Analysis rate
Copying analysis	633	58.72%
Double input analysis (childtyping)	869	80.6%
Phonetic analysis	748	69.4%
Optimised phonetic analysis	770	71.4%

mode allows this strategy to have the best analysis rate by far (80.6 %). It is interesting however to note that even with this ideal setting baseline, the engine has still an error rate of 19.4%, which demonstrates the complexity of the task in hand. The phonetic analysis strategy achieves a significantly lower recognition rate (69.4 %), however by far better than the existent copying analysis strategy (with only the instruction as prior knowledge). The gap between phonetic analysis and double input analysis can be explained by the fact that there are a lot of incorrectly written productions that are not phonetically similar to the instruction, which renders the phonetic guidance obsolete. The competition between analysis instances **in the optimised phonetic analysis strategy** results in a gain of 2% (71.4 % recognition rate).

In any case, there is room for improvement, as in optimising the recognition engines used to identify letter hypotheses, by using the amount of data that is being collected to improve the letter models. Moreover, the phonetic analysis chain can be improved by relearning the substitution costs and optimising the phonetic hypotheses generation process.

Given that there is some uncertainty in the interpretation robustness, its it important that the feedback generation strategy minimises the analysis engine errors.

5.2 Feedback results

Table 5 presents the feedback generation results for the baseline double input analysis mode (three first columns) and for the optimized phonetic analysis mode (three last columns).

Table 5. Feedback generation pertinence

Confidence	Analysis with double input			Optimised phonetic analysis		
	Ratio	Errors	Errors rate	Ratio	Errors	Errors rate
High	857 (79.4%)	2	0.2%	819 (75.9%)	111	13.5%
Medium	137 (12.7%)	0	0%	85 (7.9%)	0	0%
Reject	88 (8.16%)	0	0%	174 (16.1%)	0	0%
Total feedback	994 (92.2%)	2	0.2%	904 (83.8%)	111	12.2%

We can see for the baseline double input, that even though the analysis rate is "only" 80 %, the feedback error ratio (total feedback errors / total feedback)

is limited to 0.2%. This is due to the fact that most of the words that were not correctly analysed were either rejected or considered as medium confidence feedback. We can therefore conclude that the defined feedback statuses and the feedback generation decision criteria enable the system to have a good performance in the context of the analysis with double input.

Moreover, it is clear from the table that using the phonetic correspondence of the recognised word as a criterion for feedback generation is not as precise as using the baseline double input strategy, since it is an approximation of the ground truth. We can also see that there is more reject (174 versus 88), which is explained by the fact that we did not yet find a way to deal with children production that are phonetically incoherent with the production. In any case, we do observe the same improvement on the error ratio (feedback error ratio = 12.2% whereas analysis error ratio = 28.6%), which is encouraging. Finally, the possible improvements discussed relating to the analysis process would have a big impact on the feedback pertinence.

6 Conclusion

In this paper, we present an original approach for the orthographic analysis of children handwritten words in a dictation context. This approach is based on the extension of an existing analysis engine that was suited for copying exercises. Dictation exercises are more challenging since the child only hears the word he/she has to reproduce. As a consequence, we are faced with more morphological and orthographic errors. We defined two strategies to cope with this challenge. This first intermediary approach puts the user in the analysis loop, as the pupil has to type the word he/she has written on the keyboard after the production competition. This explicit groundtruth is then used as prior knowledge to drive the handwriting analysis process and to retrieve the written word. The second strategy aims to add fluidity to the interaction and to be free from the user defined groundtruth and is based on the generation of phonetically similar hypotheses for each instruction, that can cover a wide range of orthographic errors. We can consider that each phonetic hypothesis is a probable approximation of the groundtruth. We adapted the analysis process and the paths search decision criteria to cope with the fact that the ground truth is unknown. The experiments showed the improvement of the system performance with the integration of these new strategies, and the pertinence of the feedback generated to the pupils. Our future works consists in extending this analysis engine to the interpretation short sentences.

Acknowledgement

”P2IA” is funded by the French government. We would like to tank the project partners from Learn & Go, the University of Rennes 2, LP3C lab, INSA Rennes, University of Rennes 1 and IRISA lab. Additionnallys, parts of these works were

supported by LabCom "Scripts and Labs" funded by the French National Agency for Research (ANR).

References

1. Bonneton-Botté, N., Fleury, S., Girard, N., Magadou, M., Cherbonnier, A., Renault, M., Anquetil, E., Jamet, E. (2020). "Can tablet apps support the learning of handwriting? An investigation of learning outcomes in kindergarten classroom". *Computers & Education*. pp.38.
2. Anquetil, E., Lorette, G. (1997). "On-line Handwriting Character Recognition System Based on Hierarchical Qualitative Fuzzy Modeling". *Progress in Handwriting Recognition*, pp.109-116.
3. Simonnet, D., Girard, N., Anquetil, E., Renault, M., Thomas, S. (2018). "Evaluation of Children Cursive Handwritten Words for e-Education". *Pattern Recognition Letters*, Elsevier, 121. pp.133-139.
4. Almaksour, A., Anquetil, E. (2011). "Improving premise structure in evolving takagi-sugeno neuro-fuzzy classifiers". *Evolving Systems 2*, pp. 25-33.
5. Takagi, T., Sugeno, M. (1985). "Fuzzy identification of systems and its applications to modeling and control". *IEEE Transactions on Systems, Man, and Cybernetics SMC-15.1* (1985), pp. 116-132.
6. Simonnet, D., Anquetil, E., Bouillon, M. (2017). "Multi-criteria handwriting quality analysis with online fuzzy models". *Pattern Recognition*, 69, pp.310-324.
7. Damerau, F.J. (1964). "A technique for computer detection and correction of spelling errors". *Commun. ACM* 7, pp. 171-176.
8. Novak, J., Dixon, P., Minematsu, N., Hirose, K., Hori, C., Kashioka, H. (2012). "Improving WFST-based G2P conversion with alignment constraints and RNNLM N-best rescoring".
9. Chickering, A. W., Zeldin, F. G. (1987). "Seven Principles for Good Practice in Undergraduate Education". *AAHE Bulletin*, 39.7, pp. 3-7.
10. Haßler, B., Major, L., Hennessy, S. (2016). "Tablet use in schools: a critical review of the evidence for learning outcomes". *Journal of Computer Assisted Learning*, 32, pp. 139- 156.
11. M. Bisani, M., Ney, H. (2008). "Joint-sequence models for grapheme-to-phoneme conversion". *Speech Communication* 50, pp. 434-451.
12. Nkambou, R., Mizoguchi, R., Bourdeau, J. (2010). "Advances in Intelligent Tutoring Systems". t, 308.
13. Sheng, H., Schomaker, L. (2018). "Deep Adaptive Learning for Writer Identification based on Single Handwritten Word Images". *Pattern Recognition*. 88.
14. Kang, L., Rusinol, M., Fornes, A., Riba, P., Villegas, M. (2020). "Unsupervised writer adaptation for synthetic-to-real handwritten word recognition". *The IEEE Winter Conference on Applications of Computer Vision*, pp. 3502-3511.