Neural machine translation for automated feedback on children's early-stage writing

Jonas Vestergaard Jensen*[1], Mikkel Jordahn[1], and Michael Riis Andersen[1]

[1]Technical University of Denmark
{jovje, mikkjo, miri}@dtu.dk

Abstract

In this work, we address the problem of assessing and constructing feedback for early-stage writing automatically using machine learning. Early-stage writing is typically vastly different from conventional writing due to phonetic spelling and lack of proper grammar, punctuation, spacing etc. Consequently, early-stage writing is highly non-trivial to analyze using common linguistic metrics. We propose to use sequence-to-sequence models for translating early-stage writing by students into conventional writing, which allows the translated text to be analyzed using linguistic metrics. Furthermore, we propose a novel robust likelihood to mitigate the effect of label noise in the dataset. We investigate the proposed methods using a set of numerical experiments and demonstrate that the conventional text can be predicted with high accuracy.

1 Introduction

Learning to write is extremely important for both educational and communication purposes. Specific and frequent formative feedback can improve learning, but it is a time-consuming process, especially in a class-room setting [1]. In this work, we study the problem of using machine learning to assist elementary school teachers in assessing and constructing formative feedback for early-stage writing.

Children's early writing can be studied and quantified in several ways ranging from simple count statistics to more sophisticated linguistic metrics and such metrics can be tracked over time to assess and facilitate learning [2]. However, emergent writing is often characterized by phonetic spelling as well as lack of proper grammar, spacing, and punctuation etc. [3], which makes automatic and quantitative analysis highly non-trivial.

To address this problem, we propose to use neural machine translation models to "translate" the early-writing of a student to the equivalent "conventional" writing. This makes it possible to compare and quantify the difference between student texts and the corresponding conventional texts and to evaluate the texts using linguistic metrics of interest. More specifically, we model the writing produced by students as noisy observations of the corresponding conventional writing and thus aim to denoise the student texts using sequence-to-sequence models.

Due to the current success of Transformer-based models in many natural language processing (NLP) applications [4], we employ the so-called BART architecture, which is a Transformer-based denoising autoencoder for sequence-to-sequence problems [5]. BART is pre-trained to reconstruct corrupted documents and is therefore a natural choice for our application since our objective of denoising student texts is well-aligned with BART's pre-training task. For a review on other recent sequence-to-sequence methods for neural machine translation, see e.g. Stahlberg [6] or Tan et al. [7].

Machine learning has previously been used for automated essay scoring (AES) [8], but AES generally focuses on automated grading rather than automated feedback and often targets more senior student populations than those considered in our work [9]. For recent systematic reviews on AES, we refer to Ramesh and Sanampudi [9] and Klebanov and Madnani [10].

We train and evaluate the model on a dataset collected using a digital learning platform[1]. The dataset consists of $N = 36,610$ pieces of early writing produced by students and the corresponding conventional texts produced by teachers after interacting with the students, i.e. the dataset is $\mathcal{D} = \{(x_n, y_n)\}_{n=1}^{N}$, where x_n and y_n are the nth student and teacher text, respectively. Examples of students texts and their corresponding conventional writing are given in Table 1.

The dataset contains a significant amount of noise caused by students or teachers using the learning platform in unintended ways. In approximately 25% of the data, there is no relationship between the texts in the pair (x_n, y_n). For example, a student wrote 'norah loves peas!' while the corresponding teacher text is 'Elephants are big.' We refer to this type of noise as "label noise" and say that y_n is a noisy label for x_n if the two texts are unrelated. To combat the label noise, we propose a novel robust likelihood for sequence-to-sequence modelling.

To evaluate the proposed methods, we investigate

*Corresponding Author.

[1]www.writereader.com

Proceedings of the 5th Northern Lights Deep Learning Conference (NLDL), PMLR 233, 2024.
© 2024 Jonas Vestergaard Jensen, Mikkel Jordahn, & Michael Riis Andersen. This is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/4.0/).

Table 1. Examples of student texts and the corresponding conventional texts.

Student text	Conventional text
We lern ubut eath in sins.	We learn about Earth in Science.
thedinousouisrune	The dinosaur runs.
ledkos boo fune thengs	Leprechauns do funny things.

and report how accurately a teacher text y_* can be predicted from the student text x_* given a training set $\mathcal{D}$. We also consider the task of estimating two readability metrics (Flesch–Kincaid [11] and LIX [12]) from the student texts.

The main contributions of this paper are 1) to demonstrate that the student texts can be denoised with high accuracy, 2) to demonstrate that translation of the student texts to conventional writing significantly improves the accuracy of the estimated linguistic metrics, 3) introduction and evaluation of a novel likelihood for sequence-to-sequence data with noisy lables, and 4) investigation of whether the quality of a translation can be assessed through its average likelihood.

2 Methods

2.1 Sequence-to-sequence modelling

We frame the problem as a sequence-to-sequence problem where the goal is to estimate the teacher text y_n given a student text x_n. In other words, given a training set $\mathcal{D} = \{(x_n, y_n)\}_{n=1}^N$, our goal is to estimate the distribution $p(y_*|x_*, \mathcal{D})$ for some new student text x_*. Formally, each sequence $x_n = (x_{n,1}, x_{n,2}, \ldots, x_{n,N_x})$ consists of an ordered list of tokens from a fixed vocabulary $\mathcal{A}$, i.e. $x_{n,i} \in \mathcal{A}$, where N_{x_n} denotes the length of x_n. We use K to denote the size of the vocabulary, i.e. $K = |\mathcal{A}|$. Furthermore, we use the notation $x_{n,1:j}$ to denote the first j tokens of x_n, i.e. the subsequence $x_{n,1:j} = (x_{n,1}, x_{n,2}, \ldots, x_{n,j})$. We use the same notation for y_n. Finally, we sometimes omit the data index n and simply write y_i for the ith token in y and $y_{1:j}$ for the first j tokens in y.

Model archictecture We use the BART sequence-to-sequence architecture [5] to model $p(y_n|x_n)$. BART uses an encoder-decoder architecture with a bidirectional Transformer model [13] as the encoder and an autoregressive Transformer model [14] as the decoder yielding the following likelihood for the nth observation

$$p(y_n|x_n) = \prod_{i=1}^{N_y} p(y_{n,i}|x_n, y_{n,1:i-1}). \quad (1)$$

After training the model, we can predict y using greedy search as follows

$$\hat{y}_{n,i} = \arg\max_k p(y_{n,i} = k|x_n, \hat{y}_{n,1:i-1}), \quad (2)$$

where $\hat{y}_{n,i}$ denotes the prediction for the ith token in the nth example.

Loss function and label smoothing Due to the autoregressive nature of the model, predicting each toking in y_n is a multi-class classification problem, and hence, the cross-entropy loss function is a natural choice

$$\ell(x, y) = -\sum_{i=1}^{N_y}\sum_{k=1}^{K} q(y_i = k) \log p(y_i = k|x, y_{1:i-1}),$$

where $q(y_i = k) = \delta_{k,y_i}$ is a Kroncker's delta function such that $\delta_{k,y_i} = 1$ if $y_i = k$ and 0 otherwise.

We employ label smoothing for regularization like Lewis et al. [5]. That is, $q(y_i)$ is replaced with a mixture between $q(y_i)$ and a uniform distribution over the vocabulary [15]

$$q'(y_i = k) = (1 - \epsilon)\delta_{k,y_i} + \epsilon\frac{1}{K}, \quad (3)$$

where K is the size of the vocabulary and $\epsilon \in [0, 1]$ is the smoothing parameter.

2.2 Robust likelihood for noisy data

As mentioned in the introduction, a significant proportion of the observations in the dataset have noisy labels, i.e. their target sequence y_n is unrelated to their input sequence x_n. It has been shown that noise in training datasets can dramatically decrease prediction performance [16]. Hernández-Lobato et al. [17] proposed a likelihood for multi-class classification that accounts for labelling errors in the dataset. Inspired by this work, we propose a novel likelihood for robust sequence-to-sequence modelling to mitigate the effect of the noise in the data.

We construct the likelihood using the following generative process. For each example, we introduce a latent binary variable, $\theta_n \in \{0, 1\}$, indicating whether the corresponding target sequence y_n is a noisy label ($\theta_n = 1$) or not ($\theta_n = 0$). If y_n is noise-free, i.e. $\theta_n = 0$, then we model y_n conditionally as $p(y_n|x_n, \theta_n = 0) = p_{\text{BART}}(y_n|x_n)$ from eq. (1). On the other hand, if y_n is a noisy label (i.e. $\theta_n = 1$),

then we assume $p(y_n|x_n, \theta_n = 1) = p_{\text{LM}}(y_n)$, where $p_{\text{LM}}(y_n)$ is a language model independent of x_n. Thus, the robust likelihood for the nth example is

$$p(y_n|x_n, \theta_n) = p_{\text{BART}}(y_n|x_n)^{1-\theta_n} p_{\text{LM}}(y_n)^{\theta_n}. \quad (4)$$

Imposing i.i.d. Bernoulli distributions on the latent indicator variables, i.e. $\theta_n \sim \text{Ber}(\alpha)$, and marginalizing yields the following robust likelihood

$$\begin{aligned} p(y_n|x_n) &= \sum_{i\in\{0,1\}} p(y_n, \theta_n = i|x_n) \\ &= \sum_{i\in\{0,1\}} p(y_n|x_n, \theta_n = i)p(\theta_n = i) \quad (5) \\ &= (1-\alpha)p_{\text{BART}}(y_n|x_n) + \alpha p_{\text{LM}}(y_n), \end{aligned}$$

where $\alpha \in [0, 1]$ controls the rate of noisy examples. Eq. (5) implements the assumption that the dataset contains $\alpha \cdot 100\%$ noisy labels, i.e. pairs (x_n, y_n) where y_n is unrelated to x_n, but we do not know for which n. In the special case, where $\alpha \to 0$, we recover the classic likelihood from eq. (1). On the other hand, when $\alpha \to 1$ the model becomes a language model independent of x_n. The language model can range from a simple uniform distribution to an n-gram model to a complex neural language model.

For a pair (x_n, y_n) with no relationship between x_n and y_n, we expect that $p_{LM}(y_n) > p_{\text{BART}}(y_n|x_n)$ on average and therefore a lesser contribution to the loss. We confirmed this behavior empirically.

2.3 Calibration and decision-making

Reconstruction of the teacher text y_n from x_n can be an ill-posed problem in the sense that y_n is not always uniquely determined from x_n. For example, if students are really early in their writing development or not focused on the writing task, then x_n may contain very little information about what the student intended to write and in these cases, it is not possible to predict y_n from x_n alone. For example, the conventional writing of the student text 'de bea r pae' is 'These bears are playing.', which is not obvious. However, using eq. (2) always leads to a prediction.

To be able to reject predictions for such examples, we investigate to what degree the average log-likelihood, i.e.

$$C(\hat{y}|x) = \frac{1}{N_{\hat{y}}} \sum_{i=1}^{N_{\hat{y}}} \log p(\hat{y}_i|x, \hat{y}_{1:i-1}) \quad (6)$$

of the translation $\hat{y}$ of x reflects the quality of the predicted text $\hat{y}$. Intuitively, a translation $\hat{y}$ with a low likelihood should be a poor translation of the student text, and therefore, unsuitable to use as a basis for evaluating downstream linguistic metrics.

Model calibration It is well-known that calibrated uncertainties are required for optimal decision-making [18] and the argument above does indeed assume that the models are calibrated [19]. However, neural networks can be overconfident in their predictions [19–21]. Therefore, we consider two methods for improving model calibration: recalibration via temperature scaling [20] and Deep Ensembles (DE) [22] and investigate whether they improve the calibration of the model, and subsequently, lead to better decision-making.

Temperature scaling In general, the probabilities $p(y_i = k|x, y_{1:i-1})$ are computed by feeding the network outputs, $z_{i,k}$, through the softmax function, i.e.

$$p(y_i = k|x, y_{1:i-1}) = \frac{\exp z_{i,k}}{\sum_{k=1}^{K} \exp z_{i,k}}. \quad (7)$$

In temperature scaling, the logits $z_{i,k}$ are simply scaled by $\frac{1}{T}$ where $T > 0$ is the temperature [20]. This has the effect that the network becomes less confident for $T > 1$ and more confident for $T < 1$. The temperature T is selected to maximize the log-likelihood of the validation data. In our setup, temperature scaling only affects the likelihood $p(\hat{y}|x)$ of a translation $\hat{y}$ but not the translation itself as we use the greedy search strategy in eq. (2).

Deep Ensembles DEs [22] have been shown to not only provide well-calibrated probabilities, but also to provide superior predictive performance in many settings [23]. DEs are typically implemented by keeping the model architecture and training parameters fixed and simply changing the initialization of the network before training. After fitting the model using S different initializations, we can make predictions by averaging the individual models' probabilities

$$p_{\text{DE}}(y_i = k|x, y_{1:i-1}) = \frac{1}{S} \sum_{s=1}^{S} p^s(y_i = k|x, y_{1:i-1}),$$

where $p_{\text{DE}}(y_i = k|x, y_{1:i-1})$ denotes the predictive distribution for the DE and p^s denote the sth model in the ensemble.

3 Experiments

To investigate the proposed methods, we designed and conducted a number of numerical experiments.

3.1 Data

The dataset has been collected from a digital learning platform[2] and consists of $N = 36,610$ pairs of student and teacher texts, where the teacher text

[2] www.writereader.com

for a given student text is the conventional writing of the student text, i.e. the student text modified to have proper spelling, grammar etc. The collected data was filtered to remove sensitive information. We use 80% of the data for training, 10% for validation, and 10% for testing.

The data contains a substantial amount of label noise due to unintended use of the learning platform. For example, a student may have written text both in the designated student text field but also in the text field designated for the teacher. Based on a sample of 1,000 pairs, we estimate the percentage of faulty pairs to be around 25%. To ensure reliable performance estimates, the validation and test dataset have been manually filtered by a teacher to remove faulty pairs. The resulting validation set and test sets contained $N_{\text{val}} = 2,586$ and $N_{\text{test}} = 2,767$ observations, respectively.

Data augmentation We synthetically increase the amount of data by simulating student texts and use this synthetic data in an extra fine-tuning step. We augment conventional children's books provided by Danish publishers with several operations to emulate student text, including word and letter deletions, shortening of words to their initial letter, cutting of word endings, and introduction of common misspellings of letters and bigrams based on the real training data. All operations are applied randomly with heuristically chosen frequencies. This resulted in 279,553 simulated pairs of student and teacher texts. The synthetic data is only used for training, i.e. no synthetic data is included in the real-world test set used for evaluating the models.

3.2 Hyperparameters

In all experiments, we use the "base" version of BART, which has 6 layers in both the encoder and decoder and a total of 140 million parameters. We use the pre-trained BART model from FAIRSEQ, which has been trained to denoise corrupted text in English [5, 24]. We fix the label smoothing parameter to $\epsilon = 0.1$. We further regularize the model with dropout [25] and weight decay [26]. The dropout rate and the amount of weight decay are selected using a grid-search on the validation data, where the median normalized edit distance (see Section 3.3) is used as the selection criterion. Similarly, we train the model until it has converged in terms of the validation median normalized edit distance using the AdamW [26] optimizer with a learning rate of 3e−5. The text data is encoded with the GPT-2 byte pair encoding [27], which has a vocabulary size of $K = 50,260$.

3.3 Reconstructing the teacher texts

The purpose of this experiment is to quantify how accurate the teacher texts can be reconstructed from the student texts. We assess the quality of a reconstructed text $\hat{y}$ for x by its character-level edit distance (the Levenshtein-distance) to the true teacher text y, i.e. $\text{ED}(y, \hat{y})$, that counts the number of deletions, insertions, and substitutions needed to transform $\hat{y}$ into y [28]. Since the ED depends on the length of the inputs, we also consider a normalized ED, which is in the interval $[0, 1]$:

$$\text{NED}(y, \hat{y}) = \frac{\text{ED}(y, \hat{y})}{\max(|y|, |\hat{y}|)}, \tag{8}$$

where $|y|$ and $|\hat{y}|$ denote the length of the sequences.

We compare the fine-tuned models to three different baseline models. The simplest baseline is the Identity model that simply provides the input text x as the reconstruction $\hat{y}$, i.e. $\hat{y} = x$. We also compare against the pre-trained BART model without any fine-tuning, and with reconstructions provided by ChatGPT. A description of how ChatGPT was used to produce denoised student texts is given in Appendix A.

In the first section of Table 2, we report both mean and median EDs for the test set. It is seen that both the Identity model and the pre-trained BART without fine-tuning achieve a mean NED of 0.16, whereas ChatGPT improves the NED to 0.11. Fine-tuning of the BART model with the training dataset further reduces the mean NED to 0.09. The best mean NED of 0.08 and median NED of 0.01 was achieved by first fine-tuning with the synthetic student texts and subsequently fine-tuning with the real training data.

3.4 Robust likelihood

The next experiment is designed to investigate the benefit of the novel robust likelihood proposed in eq. (5). This likelihood requires the use of an external language model. We employed simple n-gram models with $n = 2, 4, 6$ on the token level, which were estimated using the training data via the KenLM toolkit [29]. We set $\alpha = 0.25$ in the Bernoulli distribution and also regularize the model with dropout and weight decay as with the label smoothed cross-entropy loss.

In the middle section of Table 2, we again report mean and median EDs. We observe that the proposed robust likelihood with a 2-gram language model reduces the mean ED to 3.20 and the use of a more complex language model (6-gram) further reduces the mean ED to 3.11 and the median ED and NED to 0. We do, however, observe that the standard errors of the mean ED and NED for the robust likelihood models overlap with the standard errors

Table 2. The baselines and the results of fine-tuning the BART model with the methods described in Section 2 and the data and metrics described in Section 3. ± indicates the standard error of the mean. (synth., synthetic; temp., temperature; ED, edit distance; NED, normalized ED; MAE, mean absolute error; FK, Flesch-Kincaid; ECE, expected calibration error; MCE, maximum calibration error).

	MEAN		MEDIAN		MAE			
	ED	NED	ED	NED	FK	LIX	ECE	MCE
Identity	5.40±0.12	0.16±0.00	4.00	0.12	1.36±0.06	5.66±0.26		
ChatGPT	5.19±0.22	0.11±0.00	2.00	0.05	0.78±0.05	3.36±0.20		
BART (no fine-tuning)	5.67±0.14	0.16±0.00	4.00	0.12	1.39±0.06	5.77±0.26	0.19	0.29
BART (fine-tuning)	3.65±0.14	0.09±0.00	1.00	0.02	0.58±0.03	2.88±0.14	0.04	0.65
BART (synth. data, fine-tuning)	3.23±0.14	**0.08**±0.00	1.00	0.01	0.57±0.03	2.65±0.14	0.04	**0.07**
BART (synth. data, fine-tuning, robust 2-gram)	3.20±0.13	**0.08**±0.00	1.00	0.01	0.57±0.03	2.65±0.14	0.03	0.14
BART (synth. data, fine-tuning, robust 4-gram)	**3.11**±0.14	**0.08**±0.00	1.00	0.01	**0.55**±0.03	2.61±0.14	0.04	0.23
BART (synth. data, fine-tuning, robust 6-gram)	**3.11**±0.14	**0.08**±0.00	**0.00**	**0.00**	0.56±0.03	**2.60**±0.13	0.04	0.14
BART (synth. data, fine-tuning, temp. scaling)	3.23±0.14	**0.08**±0.00	1.00	0.01	0.57±0.03	2.66±0.14	**0.02**	0.11
BART (synth. data, fine-tuning, deep ensemble)	3.16±0.14	**0.08**±0.00	1.00	0.01	0.56±0.03	2.68±0.14	0.04	0.12

for the best model fine-tuned with the smoothed cross-entropy loss.

3.5 Predicting the linguistic metrics

In the third experiment, we investigate how accurately we can estimate the linguistic metrics. We compute the linguistic metrics on the teacher texts y_n and consider those the ground truth. We then evaluate the same metrics using the reconstructed texts $\hat{y}_n$ and compare these to the ground truth.

In this work, we focus on two simple metrics for text complexity and readability, namely the Flesch-Kincaid grade level formula (FK) [11] and the readability index LIX [12], which are given by

$$\text{FK} = 0.39\frac{\text{No. words}}{\text{No. sentences}} + 11.8\frac{\text{No. syllables}}{\text{No. words}} - 15.59$$
$$\text{LIX} = \frac{\text{No. words}}{\text{No. sentences}} + 100\frac{\text{No. long words}}{\text{No. words}},$$

where long words are defined as words with more than 6 characters. We clip the FK and LIX predictions to the interval $[a, 2b]$, where a is the theoretical lower limit of the metric and b is the threshold value for very complex texts. We have that $(a, b) = (-3.4, 18)$ and $(a, b) = (1, 55)$ for FK and LIX, respectively.

Table 2 summarizes the results. We observe that all fine-tuned models achieve comparable performance for LIX, but substantially lower MAEs compared to the baseline models. We also observe the same pattern for the FK metric.

In Appendix B, we provide the mean and median LIX and FK computed on the student texts, the predicted conventional texts, and the ground truths and see that the predicted texts are more similar to the ground truths than the student texts, as expected.

3.6 Calibration and decision-making

The purpose of the last experiment is to evaluate and compare the models in terms of calibration and to investigate whether the average log likelihood of the predicted sequences can be used to identify poor predictions. The miscalibration of a model is the difference between the models' confidence and the probability of the model being correct, where the confidence of the model is the probability of the predicted token. We quantify the calibration error using both the expected calibration error (ECE) and the maximum calibration error (MCE) [20]. We compute the calibration metrics over all tokens in the test set.

For this experiment, we further compare against a temperature scaled model and a DE constructed using three fine-tuned BART models from random initializations of the model parameters. The last two columns in Table 2 summarizes the results. It can be observed that all models (except the BART model with pre-training only) perform similar in terms of ECE. Nonetheless, the temperature scaled model is best calibrated, as expected.

In terms of MCE, the temperature scaled model is, however, not the best calibrated model. The BART model fine-tuned only on the real training data shows the largest MCE, but interestingly, the fine-tuning with synthetic student data greatly reduces the MCE. We also note that the DE method does not lead to improved calibration compared to the single model. However, it does slighly improve the reconstructions w.r.t. the mean ED.

Figure 1 shows accuracy-rejection curves [30] for the mean NED, FK MAE, and the LIX MAE for a selection of models from Table 2. The data points in the plots are obtained by varying the log-likelihood threshold for accepting and rejecting predictions. Figure 1 reveals that the average log-likelihood $C(\hat{y}|x)$ is indeed a viable feature for implementing a reject option as all three metrics improves as we reject more translations $\hat{y}$. For example, the mean

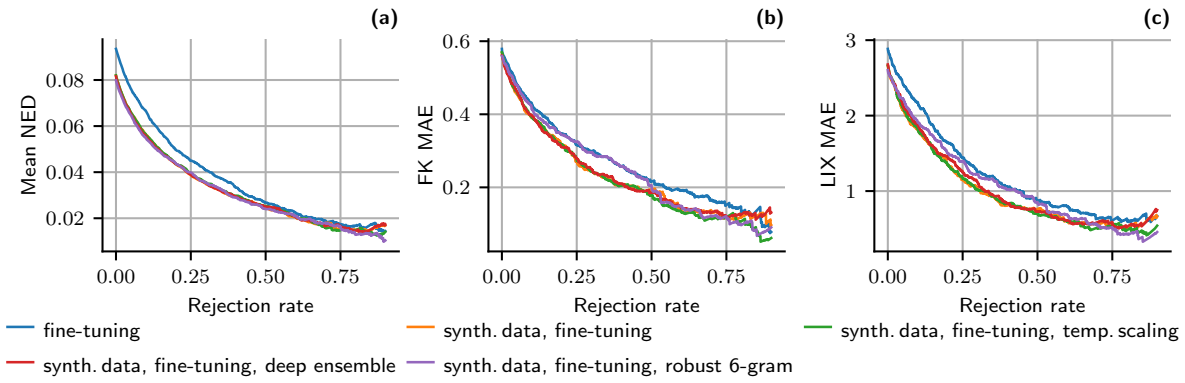

Figure 1. **(a)** Accuracy-rejection curve for the mean NED and a selection of models from Table 2. **(b)-(c)** Same curves as in **(a)** but for the FK MAE and LIX MAE, respectively. The data points in all figures are obtained by varying the rejection threshold for the average log-likelihood $C(\hat{y}|x)$ of a translation $\hat{y}$. "Lower" curves are better.

NED can be reduced from approx. 0.08 to 0.04 if one is willing to reject 25% of the test observations.

Finally, we see that the rejection curves for all models are quite similar. This suggests that the model calibration does not influence the decision-making abilities of the models, although this result could also be due to the relatively small calibration and performance differences between the models.

4 Conclusion

In this work, we have framed the automated feedback on children's early-stage writing as a machine translation problem, where we translate students' early writing into conventional writing. We demonstrated that the conventional writing can be predicted with high accuracy by fine-tuning a pre-trained BART architecture. We also showed that the readability metrics, Flesch-Kincaid and LIX, can be estimated with significantly higher accuracy using the translations compared to the student texts directly. Furthermore, as an alternative to the label-smoothed cross-entropy loss function, we proposed a novel robust likelihood to mitigate the effects of label noise in the observed data. Our experiments indicated a slightly improved predictive accuracy. Finally, we have shown that the log-likelihood can be used as a criterion for identifying poor translations by the sequence-to-sequence models, inducing a trade-off between accuracy and rejected predictions.

Acknowledgments

We acknowledge funding from Innovation Fund Denmark through the InnoBooster program (grant number 2055-00497B). We sincerely thank Janus Madsen and Lasse Sørensen from WriteReader for providing and curating the dataset as well as providing the ChatGPT-based translations of the student texts. We also thank the publishers Alinea and Gyldendal for providing the children's books used to create synthetic data.

References

[1] D. Saliu-Abdulahi, G. O. Hellekjær, and F. Hertzberg. "Teachers' (Formative) Feedback Practices in EFL Writing Classes in Norway". In: *Journal of Response to Writing* 3 (1 2017), pp. 31–55.

[2] J. Bundsgaard, K. Kabel, and J. Bremholm. "Validating scales for the early development of writing proficiency". In: *Writing and Pedagogy* 13.1-3 (July 2022), pp. 89–120. DOI: 10.1558/wap.21491.

[3] K. Kabel, J. Bremholm, and J. Bundsgaard. "A framework for identifying early writing development". In: *Writing and Pedagogy* 13.1-3 (July 2022), pp. 51–87. DOI: 10.1558/wap.21467.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is All you Need". In: *Advances in Neural Information Processing Systems*. Vol. 30. 2017.

[5] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020, pp. 7871–7880. DOI: 10.18653/v1/2020.acl-main.703.

[6] F. Stahlberg. "Neural machine translation: A review". In: *Journal of Artificial Intelligence Research* 69 (2020), pp. 343–418. DOI: 10.1613/jair.1.12007.

[7] Z. Tan, S. Wang, Z. Yang, G. Chen, X. Huang, M. Sun, and Y. Liu. "Neural machine translation: A review of methods, resources, and tools". In: *AI Open* 1 (2020), pp. 5–21. DOI: 10.1016/j.aiopen.2020.11.001.

[8] F. Dong, Y. Zhang, and J. Yang. "Attention-based recurrent convolutional neural network for automatic essay scoring". In: *Proceedings of the 21st conference on computational natural language learning (CoNLL 2017)*. 2017, pp. 153–162. DOI: 10.18653/v1/k17-1017.

[9] D. Ramesh and S. K. Sanampudi. "An Automated Essay Scoring Systems: A Systematic Literature Review". In: *Artificial Intelligence Review* 55 (Mar. 2022), pp. 2495–2527. DOI: 10.1007/s10462-021-10068-2.

[10] B. B. Klebanov and N. Madnani. "Automated Evaluation of Writing – 50 Years and Counting". In: *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, July 2020, pp. 7796–7810. DOI: 10.18653/v1/2020.acl-main.697.

[11] J. P. Kincaid, R. P. Fishburne Jr, R. L. Rogers, and B. S. Chissom. *Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel*. 1975. DOI: 10.21236/ada006655.

[12] C. Björnsson. *Läsbarhet*. Liber, Stockholm, Sweden, 1968.

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, June 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.

[14] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. *Improving Language Understanding by Generative Pre-Training*. 2018. URL: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. "Rethinking the Inception Architecture for Computer Vision". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 2818–2826. DOI: 10.1109/cvpr.2016.308.

[16] S. Gupta and A. Gupta. "Dealing with Noise Problem in Machine Learning Data-sets: A Systematic Review". In: *Procedia Computer Science* (2019). DOI: 10.1016/j.procs.2019.11.146.

[17] D. Hernández-Lobato, J. Hernández-Lobato, and P. Dupont. "Robust Multi-Class Gaussian Process Classification". In: *Advances in Neural Information Processing Systems*. Vol. 24. 2011.

[18] J. O. Berger. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media, 2013. DOI: 10.1007/978-1-4757-4286-2.

[19] M. Minderer, J. Djolonga, R. Romijnders, F. Hubis, X. Zhai, N. Houlsby, D. Tran, and M. Lucic. "Revisiting the calibration of modern neural networks". In: *Advances in Neural Information Processing Systems*. Vol. 34. 2021, pp. 15682–15694.

[20] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. "On calibration of modern neural networks". In: *34th International Conference on Machine Learning*. PMLR. 2017, pp. 1321–1330. DOI: 10.48550/arXiv.1706.04599.

[21] S. Desai and G. Durrett. "Calibration of Pre-trained Transformers". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Nov. 2020, pp. 295–302. DOI: 10.18653/v1/2020.emnlp-main.21.

[22] B. Lakshminarayanan, A. Pritzel, and C. Blundell. "Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles". In: *Advances in Neural Information Processing Systems*. Vol. 30. 2017.

[23] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, S. Nowozin, J. Dillon, B. Lakshminarayanan, and J. Snoek. "Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift". In: *Advances in Neural Information Processing Systems*. Vol. 32. 2019. DOI: 10.48550/arXiv.1906.02530.

[24] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli. "fairseq: A Fast, Extensible Toolkit for Sequence Modeling". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*

(Demonstrations). Association for Computational Linguistics, June 2019, pp. 48–53. DOI: 10.18653/v1/n19-4009.

[25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15.56 (2014), pp. 1929–1958.

[26] I. Loshchilov and F. Hutter. *Decoupled Weight Decay Regularization*. 2017. URL: http://arxiv.org/abs/1711.05101.

[27] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. *Language Models are Unsupervised Multitask Learners*. 2019. URL: https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

[28] V. I. Levenshtein. "Binary codes capable of correcting deletions, insertions, and reversals". In: *Dokl. Akad. Nauk SSSR*. Vol. 163. 4. 1965, pp. 845–848.

[29] K. Heafield. "KenLM: Faster and Smaller Language Model Queries". In: *Proceedings of the Sixth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, July 2011, pp. 187–197.

[30] M. S. A. Nadeem, J.-D. Zucker, and B. Hanczar. "Accuracy-Rejection Curves (ARCs) for Comparing Classification Methods with a Reject Option". In: *Proceedings of the third International Workshop on Machine Learning in Systems Biology*. Vol. 8. PMLR, Sept. 2009, pp. 65–81.

A Denosing student texts with ChatGPT

To obtain denoised early-stage writing with ChatGPT, we used OpenAI's chat completion API. The context given to ChatGPT consisted of a description of the task, ChatGPT's role in this task, examples of inputs and desired outputs, and finally the student text that should be denoised. This query was then repeated for each student text to obtain their denoised versions. The GPT model used was GPT-3.5 Turbo.

B Overall readability and text complexity

In Table B.1, we provide the mean and median FK and LIX computed on the raw student texts x, the conventional texts predicted by the models, and the ground truth conventional texts y provided by teachers. Table B.1 shows that there is indeed a difference between the readability and text complexity of the student texts and the teacher texts. Furthermore, Table B.1 reveals that the predicted texts are qualitatively more similar to the ground truths than the student texts, and that the models trained with the robust likelihood generally produce the texts that are the most similar to the ground truths in terms of readability and text complexity.

Table B.1. Text complexity and readability scores computed on the student texts x, the predicted conventional texts, and the teacher texts (ground truths) y. ± indicates the standard error of the mean. (synth., synthetic; temp., temperature; FK, Flesch-Kincaid).

	MEAN		MEDIAN	
	FK	LIX	FK	LIX
x	1.76±0.08	17.50±0.34	0.80	10.00
ChatGPT	1.83±0.07	17.77±0.30	1.31	14.57
BART (no fine-tuning)	1.79±0.08	17.59±0.34	0.80	10.00
BART (fine-tuning)	1.79±0.06	17.35±0.27	1.31	14.52
BART (synth. data, fine-tuning)	1.78±0.06	17.24±0.27	1.31	14.33
BART (synth. data, fine-tuning, robust 2-gram)	1.85±0.06	17.51±0.27	1.31	14.59
BART (synth. data, fine-tuning, robust 4-gram)	1.89±0.06	17.74±0.28	1.31	15.00
BART (synth. data, fine-tuning, robust 6-gram)	1.90±0.06	17.77±0.27	1.31	15.61
BART (synth. data, fine-tuning, temp. scaling)	1.78±0.06	17.24±0.27	1.31	14.25
BART (synth. data, fine-tuning, deep ensemble)	1.78±0.06	17.18±0.27	1.31	14.19
y	1.91±0.06	17.82±0.27	1.31	16.50