International Journal of Artificial Intelligence in Education (2024) 34:1378–1416
https://doi.org/10.1007/s40593-023-00387-6

ARTICLE

From the Automated Assessment of Student Essay Content to Highly Informative Feedback: a Case Study

Sebastian Gombert[1] · **Aron Fink**[2] · **Tornike Giorgashvili**[2] · **Ioana Jivet**[1,2] · **Daniele Di Mitri**[1] · **Jane Yau**[1] · **Andreas Frey**[2] · **Hendrik Drachsler**[1,2,3]

Accepted: 5 December 2023 / Published online: 25 January 2024

Abstract

Various studies empirically proved the value of highly informative feedback for enhancing learner success. However, digital educational technology has yet to catch up as automated feedback is often provided shallowly. This paper presents a case study on implementing a pipeline that provides German-speaking university students enrolled in an introductory-level educational psychology lecture with content-specific feedback for a lecture assignment. In the assignment, students have to discuss the usefulness and educational grounding (i.e., connection to working memory, metacognition or motivation) of ten learning tips presented in a video within essays. Through our system, students received feedback on the correctness of their solutions and content areas they needed to improve. For this purpose, we implemented a natural language processing pipeline with two steps: (1) segmenting the essays and (2) predicting codes from the resulting segments used to generate feedback texts. As training data for the model in each processing step, we used 689 manually labelled essays submitted by the previous student cohort. We then evaluated approaches based on *GBERT*, *T5,* and bag-of-words baselines for scoring them. Both pipeline steps, especially the transformer-based models, demonstrated high performance. In the final step, we evaluated the feedback using a randomised controlled trial. The control group received *feedback as usual (essential feedback),* while the treatment group received *highly informative feedback* based on the natural language processing pipeline. We then used a six items long survey to test the perception of feedback. We conducted an ordinary least squares analysis to model these items as dependent variables, which showed that highly informative feedback had positive effects on helpfulness and reflection.

Keywords Automated Essay Scoring (AES) · Content Scoring · Writing Assessment · Automated Feedback · Analytic Scoring

Extended author information available on the last page of the article

Introduction

Feedback is an essential component of educational instruction. Formative feedback is an important way to help students improve, especially if they are doing poorly or about average. However, it must go beyond essential feedback forms like points, grades or pass/fail to allow for such improvements. The feedback must be highly informative to enable the students to improve and meet the educational standards. Otherwise, the combination of poor feedback and no study success can initiate unproductive learning processes that are demotivating and make the students faint and hopeless (Nachtigall et al., 2020). The chance of a drop-out is, therefore, increasing with each failure.

Extensive evidence shows that such processes can be effectively broken through highly informative feedback (Hattie, 2009). According to Hattie (2009), feedback strongly affects learning success, with a mean effect size of $d = 0.75$. Wisniewski et al. (2020) even report a mean effect of $d = 0.99$ for highly informative feedback. Following their understanding, highly informative feedback includes feedback on the right/wrong, correct solution, type of processing, possibilities for improvement, hints on self-regulation and learning strategies. Such feedback provides suitable conditions for self-directed learning (Winne & Hadwin, 2008) and effective metacognitive control of the learning process (Nelson & Narens, 1994).

Especially constructed response items such as essays are useful for testing the active knowledge of learners (Livingston, 2009). Moreover, essays offer learners the possibility to reflect and consolidate knowledge. However, only a few years ago, it was simply impossible in terms of available personnel to provide highly informative feedback within large university courses for text-based tasks. Nowadays, computers and especially developments in natural language processing technology promise to provide the possibilities for implementing feedback systems by offering to automate what previously needed manual scoring labour.

Especially neural networks, particularly pre-trained transformer language models (Devlin et al., 2019) demonstrated high performance in analysing student responses in past studies (Bai & Stede, 2022; Ke & Ng, 2019). However, studies which combine the implementation of textual response scoring systems with feedback provision and, on top of that, a practical evaluation of a respective system with authentic learners are comparably rare.

An essential first step towards highly informative feedback is the design of digital learning environments and activities that provide the process and textual data needed for providing feedback for learning processes. With this case study, we provide and evaluate such a learning activity that can deliver highly informative feedback to its learners, namely a system which provides students with feedback on the content of their essays within an introductory lecture on educational psychology for teacher students at Goethe University, Frankfurt, Germany.

For this learning activity, learners must first watch a video in which a German high school student presents ten learning tips. Following this, they must write short argumentative essays to judge whether the provided learning tips are helpful

based on the lecture contents. After the submission deadline, learners are displayed feedback texts informing them of the quality of their essays and which lecture content they likely need to revisit.

For this purpose, we implemented a pipeline of multiple natural language processing components for assessing the essays automatically. Most previous work on essay scoring aimed for more generalisable solutions but mainly focused on holistically assessing the overall writing qualities of the essays at hand (Bai & Stede, 2022; Ke & Ng, 2019). In contrast, our system is distinguished from other essay scoring systems because it focuses on evaluating essays' content rather than the writing style.

For scoring the essays, following basic preprocessing, our system first segments the text into sections addressing the different learning tips. The resulting sections are coded individually following a scoring rubric. We collected and created datasets to train systems for both tasks. We evaluated two transformer-based architectures, *GBERT* and *T5,* and compared their performance against multiple bag-of-word baseline models. In the final step, the predicted scores are matched against a ground truth table offered by the course instructors, and the discrepancies are used to trigger rules populating a feedback template. Following the implementation, the feedback was evaluated with another cohort of learners using a randomised controlled trial.

This paper makes the following contributions:

- A learning activity for formative scenarios that provides learners with content-related feedback.
- A general architecture for the automated analytical assessment of essay content in combination with feedback generation.
- A first evaluation of the quality of the provided feedback with an authentic cohort of students.
- A novel dataset for analytic essay scoring in German.

Background

Feedback Generation

Feedback is arguably one of the most important components of educational instruction. Especially weaker students can profit from highly informative, constructive feedback. It is meant to help students reach a desired goal by pointing out the discrepancies between their current state and the goal (Hattie & Timperley, 2007; Nicol & Macfarlane-Dick, 2006). Hattie and Timperly (2007) distinguish four different levels of feedback:

1. Task level: how well is a given task solved?
2. Process level: how well does solving this task work?
3. Self-regulation level: how well can a learner self-regulate while solving the task?
4. Self-level: personal evaluations and affect of the learner.

Feedback can address one or multiple of these levels. Moreover, according to Hattie and Timperley (2007) and Wisniewski et al. (2020), feedback—at its core—needs to address three key questions: what are the goals (feed-up), how is it going for a student (feed-back) and where to go next (feed-forward). This is in line with Narciss (2008) who proposes that feedback should both reflect on learners' mistakes and guide them on improving.

Feedback can come in many forms and shapes and be transported using various modes, as its concrete form is usually highly dependent on the domain and task it is provided for (Di Mitri et al., 2022). In line with this, research on automated feedback systems has been comparably diverse as the literature review by Cavalcanti et al. (2021) reveals. The earliest attempts to provide automated feedback were made in the context of intelligent tutoring systems (Deeva et al., 2021; Mousavinasab et al., 2021). E.g., in terms of text-based ITS', McDonald et al. (2013) implemented a system which provides immediate feedback to short answers using a rule-based chatbot that communicates with classifiers predicting various analytic scores that trigger rules used to assemble feedback texts. Alobaidi et al. (2013) implemented a system which uses a knowledge base in combination with rules and similarity-based scoring to generate questions. Feedback to student answers is generated by providing students with an indication of whether they were wrong or right together with the respective sample solution.

Generally, feedback systems are tailored towards one specific task or domain model, and the feedback is generated from templates by a combination of rules (Cavalcanti et al., 2021). A popular choice for implementing template-based feedback through a generic system is the *OnTask* system (Pardo et al., 2018), which delivers feedback texts in E-Mails. With *OnTask*, feedback is assembled and sent according to flexible pre-defined rules determining which parts of a given template are rendered and to whom the feedback is sent. This allows for flexible feedback on all four levels, addressed by Hattie and Timperley (2007), but it also requires the manual implementation of feedback rules and templates for each task and course.

Work aimed at providing automated feedback on essay writing has been comparably rare. Horbach et al. (2022) implemented a system for scoring learner essays according to an analytic rubric using bag-of-words classifiers based on gradient boosting. The predicted codes were then used for generating feedback, which was given to students periodically. The students could revise their essays to receive feedback on the revised version.

As this poses problems in terms of scaling feedback provision across different tasks and domains, there have been first attempts at completely automating the generation of feedback without the reliance on pre-made templates. Filighera et al. (2022) created a dataset for training models to generate feedback on short answers. They used it to train the *T5* string-to-string transformer (Raffel et al., 2020) to dynamically generate feedback texts for unseen responses. However, while this seems like the most promising approach regarding flexibility and scalability, the remaining problem with this work is that their dataset targets three narrow domains, and the domain transfer capabilities of this approach have not been assessed. Moreover, the quality of the provided feedback is below that of human-provided feedback.

Automated Essay Scoring

Scoring and evaluating learners' essays is a widely researched topic in educational natural language processing (Bai & Stede, 2022; Ke & Ng, 2019). Technology for scoring essays has been actively researched for decades, and the earliest work in the area was published by Page (1967). Past work often focused on students' writing skills rather than the content and predicted holistic scores or grades, indicating the overall quality of writing (Bai & Stede, 2022; Ke & Ng, 2019).

Most of the earlier work in this research area applied statistical machine-learning techniques combined with hand-crafted feature sets to the problem. Typical techniques include multiple variants of linear regression (Crossley et al., 2011, 2015; Landauer, 2003; Page, 1967), support vector machines/regression (Yannakoudakis et al., 2011; Yannakoudakis & Briscoe, 2012; Horbach et al., 2017; Cozma et al., 2018; Vajjala, 2018), or linear discriminant analysis (McNamara et al., 2015). Typical features include text length (Crossley et al., 2011), lexical features such as *n*-grams (Chen & He, 2013; Crossley et al., 2011; Page, 1967; Phandi et al., 2015; Zesch et al., 2015), syntactic features (Yannakoudakis & Briscoe, 2012), argumentation structure (Nguyen & Litman, 2018), the presence of keywords from curated lists (Crossley et al., 2011), readability (Zesch et al., 2015), semantic features and discourse structure.

Most of the more recent work in essay scoring is based on various neural network architectures and chooses an end-to-end learning approach that skips the step of defining specific features. Taghipour and Ng (2016) introduced the first neural essay-scoring systems. This system encodes an essay text as a sequence of word bigrams using a chain of convolutional and LSTM layers. The centroid of all LSTM outputs is then used as input for a sigmoid regression layer. Subsequent neural models often focused on improving the weaknesses of this approach while using the same general architecture.

In line with the recent success of pre-trained transformer language models such as BERT (Devlin et al., 2019) in natural language processing, Mayfield and Black (2020) tested whether the general high performance of such models would also translate to the use case of holistic essay scoring. However, they claimed that their models performed worse than previously published feature-based approaches, such as the one by Cozma et al. (2018) and similar to a simple naive Bayes classifier they trained on a word *n*-gram feature set. Similar experiments by Rodriguez et al. (2019) suggest that pre-trained transformer language models can reach performance levels on par with the best feature-based models but perform worse than the best neural ones in essay scoring. Generally, this also aligns with results from Beseiso and Alzahrani (2020). Their study compares the appropriateness of hand-crafted features, static word embeddings and BERT-based contextual word embeddings for automated essay scoring, revealing that models based on these different features can achieve similar results. Results by Firoozi et al. (2023) suggest that using BERT-based models trained via active learning in a human-in-the-loop setup can significantly reduce manual coding labour in essay scoring.

While holistic grade prediction can help automate summative assessments' scoring procedures, it is not a reasonable basis for providing learners with highly

informative feedback (Horbach et al., 2017, 2022). Such feedback ideally addresses multiple dimensions of learner performance, essay quality, and content dimensions. This requires implementing more specialised systems to predict labels or scores for multiple scoring dimensions (Horbach et al., 2017). Accordingly, some works went beyond the prediction of holistic scores and addressed aspects of writing quality, such as organisation or persuasiveness (Ke & Ng, 2019). Horbach et al. (2017) trained models for various scoring dimensions addressing different aspects of form, structure and language use. However, this work is focused on the writing style of the essays.

Related Work in Content Scoring

Work focused on scoring the content of student responses dealt mostly with shorter answer formats (Bai & Stede, 2022; Burrows et al., 2015; Zesch et al., 2023). Supervised state-of-the-art approaches in this area try to detect whether a sample solution is entailed in a student response (Dzikovska et al., 2013) or use the different codes from pre-defined coding rubrics as labels for text classification.

Many earlier feature-based approaches aimed at this task used lexical features, semantic similarity, and lexical overlap scores as input for various statistical learners (Bai & Stede, 2022; Burrows et al., 2015). Other approaches tried to measure the semantic similarity of responses and sample solutions directly (Andersen & Zehner, 2021; Andersen et al., 2022; Bexte et al., 2022) using semantic vector spaces produced by, for example, SBERT (Reimers & Gurevych, 2019) or latent semantic analysis (Landauer et al., 1998) to allow for unsupervised or semi-supervised scoring.

However, these approaches are mostly outperformed by transformer-based models such as *BERT* (Devlin et al., 2019) and *T5* (Raffel et al., 2020) in recent work, which could be shown to demonstrate superior performance compared to previous feature-based approaches (Bexte et al., 2022; Camus & Filighera, 2020; Gombert et al., 2022; Sung et al., 2019). This is in stark contrast to the recent results in automated essay scoring where, so far, results are more mixed.

Semantic Segmentation

As discussed in the previous section, past work in essay scoring focused on providing either holistic or dimension-specific scores for different skills related to writing and argumentation style. In contrast, work on content scoring was mostly focused on shorter answer formats. Essays have a guiding topic and usually consist of multiple segments addressing various sub-topics of the guiding topic. How these sub-topics are structured might differ from essay to essay and task to task, making it hard to implement heuristics for thematically segmenting them. Nonetheless, in essay content scoring tasks, the validity of these different segments likely needs to be evaluated separately (Horbach et al., 2022). For this reason, essays might require segmentation before different sub-topics can be addressed differently.

In natural language processing, segmentation is a widely researched topic with a long history (Choi, 2000). The most obvious examples of this are tokenisation and

sentence splitting. However, semantic segmentation procedures go beyond these basic preprocessing steps. They aim to segment texts into distinct units based on content rather than form or function. This is mainly conducted as a preprocessing step in automatic text summarisation to identify semantically coherent text sections, which are then summarised individually to acquire a final summary.

Early work aimed to split texts into sections according to the different topics they addressed. This was first conducted by measuring the semantic similarity between sentences or paragraphs (Hearst, 1997). Subsequent work used techniques from topic modelling built upon Latent Dirichlet Allocation (Blei et al., 2003) to split the texts (Misra et al., 2011; Riedl & Biemann, 2012). In line with the general success of neural networks, more recent work applied LSTM-based neural network architectures (Li & Hovy, 2014; Li & Jurafsky, 2017) or pre-trained transformer language models (Zehe et al., 2021) to the problem of classifying sentences as segment borders.

Natural Language Processing and the German Language

German is, in line with a number of other languages, one of the more established languages in the context of natural language processing. Basic preprocessing and major problems such as constituency parsing or named entity recognition have been dealt with for decades in the context of German-focused NLP, and theres isexists a wide range of models and resources available (Ortmann et al., 2019; Barbaresi, n.d.) available. Nonetheless, there still exists a discrepancy in terms of available resources compared to English. Especially more niche problems that have been dealt with for English remain unsolved for German.

German is an Indo-European language from the West-Germanic branch of the Germanic languages (Durrell, 2006). It is an inflected language with verbal and nominal inflection depending on four cases, three genders, and six tenses with a comparable high number of irregular forms (Cahill & Gazdar, 1999; Haiden, 1997). Moreover, German is highly productive in terms of word formation through compounding (Clahsen et al., 1996). This makes it more likely, compared to English, that inputs to NLP systems contain words unencountered during training. While this problem was dealt with in the past using linguistically motivated approaches, i.e., in the form of morpheme-level word splitting (Riedl & Biemann, 2016), modern NLP often has adapted more statistically motivated approaches such as using character-level *n*-grams as vocabulary instead of linguistically motivated units(Song et al., 2021).

In recent years, transformer language models achieved major successes in German-focused NLP. Transformer language models usually implement one of three architectures: *encoder*, *decoder*, or the *encoder-decoder*, which combines the previous two (Vaswani et al., 2017). *Encoders* are usually aimed at providing vector representations of textual input. This makes them appropriate for text-, sentence- and token-level classification and regression tasks. *Decoders* are trained to predict the next token for a sequence of input tokens. As a consequence, *decoders* and

encoder-decoders can also be used for text generation. The core strength of transformers are their learned task-specific vector representations of text input.

These representations are calculated on the level of sub-word units, statistically frequent character *n*-grams from a respective training data set (Song et al., 2021). While these partially overlap with morphemes, they don't carry a meaning by themselves. Through this, German word forms and compounds can be represented flexibly without the need for complex preprocessing. The attention mechanism in transformer language models encodes vector representations of these sub-words as weighted means of the representations of surrounding sub-words. Through this, the contextual semantics of these sub-words can be flexibly represented within respective vectors. As transformer-based models achieved state-of-the-art results for essential German NLP tasks such as Named Entity Recognition (Chan et al., 2020) or dependency parsing (Grünewald et al., 2021) without the need for language-specific preprocessing beyond sentence splitting and sub-word-level tokenisation, they are able to mitigate many traditional problems in processing German.

Method

Research Questions

This study aims to implement and evaluate an essay content scoring system that provides students with automated feedback on their performance. We aim to illustrate the creation of such a system with a case study to provide orientation for practitioners by implementing such a system for one particular task. This results in two concrete research goals. On the one hand, the components we implement for the system need to be evaluated. On the other hand, the reception of the feedback by learners also needs evaluation. Accordingly, we address the following research questions in this study.

RQ1: To what degree can we automate the analytic scoring procedures (segmentation and labelling) needed to provide highly informative feedback to learners concerning the content of their essays?

RQ2: How do students perceive highly informative feedback?

The Learning Activity

The learning activity we used to implement and evaluate our system gives students the task of writing essays. It was developed for educational psychology lectures at Goethe University, Frankfurt, Germany. These lectures are mainly attended by teacher students. In their lecture, students were taught the basic concepts of learning, assessment and diagnostics from the perspective of educational psychology. E.g., this involves concepts related to learning such as long-term

Table 1 The ten learning tips presented in the video the learning activity is using

Learning tip	Description
Tip 1	Start learning early enough
Tip 2	Make systematic notes to stay organised
Tip 3	Reserve enough time for learning
Tip 4	Do not use the smartphone directly after learning
Tip 5	Use different rooms than your bedroom to learn
Tip 6	Learn and repeat in front of a mirror
Tip 7	Turn your smartphone off or place it in another room during learning
Tip 8	Stay hydrated
Tip 9	Let other people query you about the learning content
Tip 10	Stay motivated

memory, motivation or learning strategies n. For the learning activity, they were tasked to evaluate the learning tips given in a video by a German high school student. For the essays, they were asked to connect the learning tips presented in the video to these concepts and explain their choices. The individual learning tips presented in the video are shown in Table 1.

The exact task description given to students (translated from its original German version into English using ChatGPT):

> The psychological foundations of learning are an important starting point for systematically supporting students' learning. Please watch the video by It's Leena (2017) with the player available here and analyse the ten tips provided there. Please evaluate each of the tips one by one in terms of the following aspects:
>
> 1. Does the tip actually promote learning, in your opinion?
> 2. Please justify your statement and, where possible, assign it to one or more of the following lecture contents:
>   a. Learning strategies (repeating, organising, elaboration)
>   b. Metacognition, Attention, working memory, long-term memory
>   c. Motivation
> 3. In view of the lecture contents, is there an important tip that you believe should be added? Please state this if you see one.
>
> In your explanations, please use a separate paragraph for each tip. Please refer to at least two sources in your explanations. These can be scientific sources mentioned in the previous lectures or other scientific sources. Please cite exclusively according to APA guidelines; see Excursus: APA guidelines [LINK]. It is expected that you write the text yourself. Therefore, it must

not have any significant textual similarity to texts from other students or sources (slides, textbook text).

Datasets & Coding Rubrics

The dataset used to train the machine learning components is a novel one and was collected from two cohorts of students. The data stems from teacher students who had to solve the task as a compulsory assignment during the lecture. It consists of $N = 698$ essays written in German. The learners had to hand in their essays as PDF documents on *Moodle*. The PDFs were downloaded from Moodle, and the text was then extracted via *PyPDF*. The title page and reference list were removed using heuristics. The texts were then further segmented into sentences and tokens via the *SoMaJo* tokeniser using the *de_CMC* model (Proisl & Uhrig, 2016).

Table 2 and Fig. 1 show the general distributional properties of the resulting corpus of texts. It was used to generate training corpora for segmentation and the assignment of final codes, conducted by two separate teams of coders.

Coding Sentence to Learning Tip Correspondence

While the students were advised to write a paragraph for each learning tip they addressed, a closer inspection of the provided data revealed that many did not follow this advice. Consequently, it was impossible to simply split the essays into paragraphs for downstream processing. For this reason, we decided to introduce an additional segmentation step for preprocessing the texts. This step was operationalised as a sentence classification problem where the correspondence between given input sentences and learning tips is determined.

INCEpTION (Klie et al., 2018) was used to code the data accordingly. For this purpose, all essays were pre-tokenized and segmented into sentences with the help of *SoMaJo* (Proisl & Uhrig, 2016) and then imported into the annotation tool. Following this step, three coders were trained to annotate text sections with the corresponding learning tips (see Table 3 for the exact codes). The internal recommender feature of INCEpTION was activated.

First, in a pilot phase, all three coders coded the same 100 essays. Following this, the average agreement between all coders was measured. For this purpose, we calculated the individual agreement between all pairs of coders with the help of Cohen's Kappa (McHugh, 2012). Then we calculated the global average which was determined to be 0.96, indicating an 'almost perfect' agreement between all three coders.

Table 2 The distributional properties of the essay corpus

Number of essays	Average # of sentences per essay (Std)	Average # of tokens per essay (Std)	Average # of tokens per sentence per essay (Std)	Average type-token ratio (Std)
698	36.68 (9.14)	855.21 (148.44)	24.31 (6.11)	0.41 (0.04)

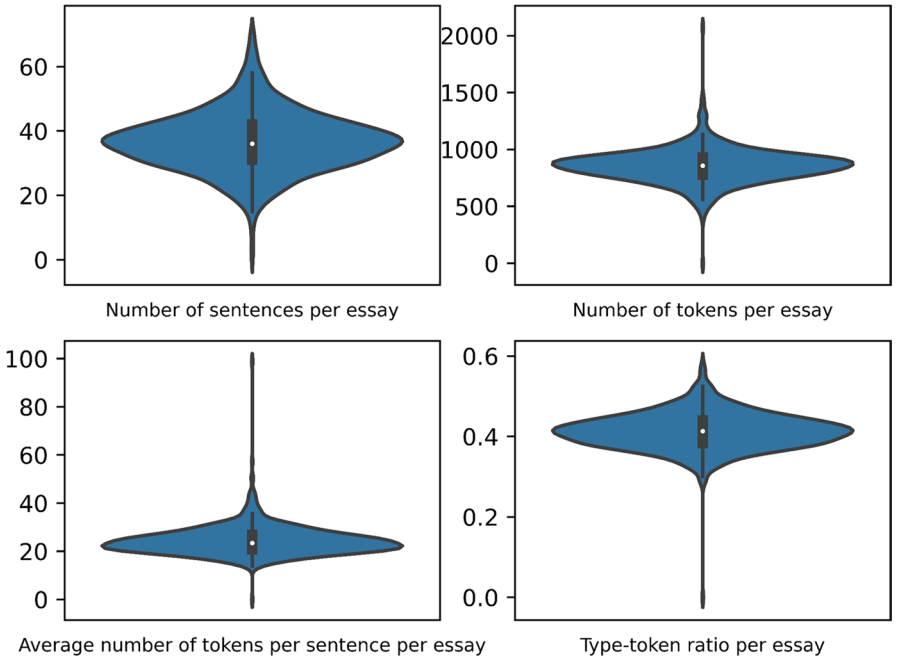

Fig. 1 Violin plots depicting the distributional properties of the essay corpus

For this reason, it was decided that no further parallel coding was needed, and the remaining essays were distributed between the coders.

Only 434 of the 698 essays were entirely coded during this step, which resulted in a satisfactory number of 22,581 sentences. The remainder of the essays were not coded for the segmentation step. Finally, the coded essays were exported from INCEpTION as UIMA XMI files and then processed with the help of *DKPro-Cassis* (Klie & Eckart de Castilho, n.d.) and *Pandas* to transform them into a tabular data format. Table 3 also shows the distribution of codes for the resulting data set.

Coding Lecture Topic to Learning Tip Correspondence

The coding of the correspondences between learning tips and lecture topics within essays was conducted by a team of seven coders on a shared *Excel* sheet. This sheet contained 100 columns, one for each combination of a learning tip and a topic from the lecture (see Table 4), and rows for each essay. The coders were tasked to read the essays individually and enter a "1" into the respective cell if a lecture concept was linked to a given learning tip within a given essay and a "0" otherwise.

In a pilot phase, 100 essays were coded in parallel with each of them being coded by two out of the seven coders. Following this, Cohen's Kappa (McHugh,

Table 3 The coding rubric applied for coding the sentence to learning tip correspondence

Code	Description	Absolute #	Relative #
Tipp 1	A sentence addresses learning tip 1 Example: "1.Tipp: Frühzeitig lernen, ich halte es für einen guten Tipp, da die zu erfüllenden Lerninhalte konstant wiederholt werden müssen." (1st tip: Learn early, I think it's a good tip, since the learning content to be fulfilled must be repeated constantly)	2563	0.11
Tipp 2	A sentence addresses learning tip 2 Example: "Der zweite Tipp von Annalena ist, dass man sich den ganzen Lernstoff nochmal ordentlich aufschreibt, da man geschriebenes besser einprägen kann, als sich den ganzen Stoff durchzulesen (It's Leena, 2017)." (The second tip from Annalena is that you write down all the learning material properly, as it is easier to memorize what you have written than to read through the whole material (It's Leena, 2017))	2312	0.10
Tipp 3	A sentence addresses learning tip 3 Example: "Der dritte Tipp zielt auf das Zeitmanagement: Man solle darauf achten, nicht den ganzen Tag zu lernen, sondern sich die Lernzeit so einteilen, dass man nur so lange lernt, wie man sich auf den Lernstoff konzentrieren kann." (The third tip is aimed at time management: you should make sure that you don't study all day, but divide your learning time in such a way that you only study for as long as you can concentrate on the subject matter)	2143	0.09
Tipp 4	A sentence addresses learning tip 4 Example: "Danach geht es um Handykonsum direkt nach dem Lernen." (Afterwards it's about mobile phone use directly after learning)	1850	0.08
Tipp 5	A sentence addresses learning tip 5 Example: "Als nächstes erzählt die YouTuberin, dass es förderlich sein könnte, in anderen Räumen zu lernen." (Next, the YouTuber tells that it could be beneficial to study in other rooms)	1968	0.08
Tipp 6	A sentence addresses learning tip 6 Example: "Wer in der Interaktion mit anderen Menschen lernt, vertieft bekanntlich sein Wissen ." (As is well known, those who learn through interaction with other people deepen their knowledge)	1808	0.08
Tipp 7	A sentence addresses learning tip 7 Example: "Der siebte Tip, das Handy beim lernen beiseite zu legen, wiederholt sich, denn dort führt sie genau die gleiche Thematik an, wie in ihrem vierten Tip, weshalb dieser für mich ein wenig redundant wirkt." (The seventh tip, to put the cell phone aside while studying, is repeated, because there she mentions exactly the same topic as in her fourth tip, which is why it seems a bit redundant to me)	1660	0.07

Table 3 (continued)

Code	Description	Absolute #	Relative #
Tipp 8	A sentence addresses learning tip 8 Example: "Der Tipp fördert das Lernen, denn Trinken verhindert eine Dehydrierung." (The tip promotes learning because drinking prevents dehydration)	1521	0.06
Tipp 9	A sentence addresses learning tip 9 Example: "Der neunte Tipp trägt erheblich zum Lernen bei." (The ninth tip contributes significantly to learning)	1887	0.08
Tipp 10	A sentence addresses learning tip 10 Example: "Man muss sich für die Handlung und Inhalte des Lernens und in Hinblick auf das Ziel motivieren, damit das Lernen leichter fällt." (One must motivate oneself for the action and content of learning and in view of the goal in order for learning to be easier)	2613	0.11
No Tipp	A sentence does not address a learning tip Example: "Abschließend lässt sich sagen, dass jeder Mensch anders lernt." (In conclusion, everyone learns differently)	1773	0.07
Additional Tipp	A sentence introduces or refers to a learning tip not presented in the video Example: "Ein für mich ergänzender Tipp, welcher das Lernen fördert, stellt das soziale Lernen dar." (An additional tip for me, which promotes learning, is social learning)	483	0.02

In addition, the absolute and relative (proportion of the total) number of sentences per assigned code is listed

Table 4 The coding rubric applied for coding topic to learning tip correspondence

Code	Description	Absolute #	Relative #
Beneficial	A learning tip is perceived as beneficial Example: "Das Angucken im Spiegel, welches in dem sechsten Tipp angesprochen wird (It's Leena, 2017), ist nicht für jeden sinnvoll, denn die soziokonstruktivistischen Ansätze beschreiben die Interaktion mit anderen Personen und nicht mit sich selbst (Kunter & Trautwein, 2013)." (Looking in the mirror, which is addressed in the sixth tip (It's Leena, 2017), does not make sense for everyone, because the socio-constructivist approaches describe the Interaction with other people and not with oneself (Kunter & Trautwein, 2013))	6064	0.89
Learning strategy	A learning tip is mentioned to be connected to the pedagogical concept of learning strategies Example: "Der zweite im Video genannte Tipp ist es, alles ordentlich aufzuschreiben (It's Leena, 2017). [SEP] Dieser Tipp kann einigen Personen helfen, da der Stoff durch das Aufschreiben nochmals wiederholt wird. [SEP] Zudem ist diese Lernstrategie ein Beispiel für Metakognition, beispielweise von deklarativem Metawissen, da sie weiß, wofür sie das Wissen einsetzt (Kunter & Trautwein, 2013)." (The second tip mentioned in the video is to write everything down properly (It's Leena, 2017). [SEP] This tip may help some people because by writing it down, the content is repeated again. [SEP] This learning strategy is also an example for metacognition, e.g. of declarative meta-knowledge, since she knows what she uses the knowledge for (Kunter & Trautwein, 2013))	1064	0.15
Repetition	A learning tip is mentioned to be connected to the pedagogical concept of repetition Example: "Als nächsten Tipp bezieht sich It's Leena (2017) darauf, dass man vor einem Spiegel lernen solle, also sich am besten im Spiegel anschauen und die Lerninhalte im Monolog vortragen solle. [SEP] Diese Aussage stimmt nur zum Teil. [SEP] Die Strategie fördert das Lernen, aber Schüler*innen sind nicht zwangsläufig auf den Spiegel angewiesen. [SEP] Auch ein anderer beliebiger Ort genügt zum Vortragen, denn es geht schließlich um das Wiederholen und Elaborieren. [SEP] Weiterhin auch um das Verankern im Langzeitgedächtnis; durch die tiefe, semantische Verarbeitung (Gold & Hasselhorn, 2017)." (As the next tip, It's Leena (2017) refers to learning in front of a mirror, i.e. look at yourself in the mirror and present the learning content in a monologue. [SEP] This statement is only partly true. [SEP] The strategy encourages learning, but students are not necessarily dependent on the mirror. [SEP] Any other location is also sufficient to lecture, because it is ultimately about repetition and elaboration. [SEP] Furthermore also about anchoring in long-term memory; through the deep, semantic processing (Gold & Hasselhorn, 2017))	1560	0.22

Table 4 (continued)

Code	Description	Absolute #	Relative #
Organisation	A learning tip is mentioned to be connected to the pedagogical concept of organisation Example: "Der zweite Tipp ist der, dass alle Informationen ordentlich aufgeschrieben werden sollen (It's Leena, 2017). [SEP] Ob dies ein guter Tipp ist, hängt damit zusammen, wie man die Informationen aufschreibt. [SEP] Wenn man den Text nicht nur sauber aufschreibt, sondern auch anfängt ihn zum Beispiel durch eine Mind-Map zu veranschaulichen, den Inhalt hinterfragt und versteht, dann spricht dies die Lernstrategie Organisieren an. [SEP] Durch das Beschäftigen mit dem Text und auch durch das Zusammenhänge herstellen unterstützt das Organisieren dabei, eine Verbindung zwischen den Informationen herzustellen (Gold & Hasselhorn, 2017)." (The second tip is that all information should be written down properly (It's Leena, 2017). [SEP] Whether this is a good tip has to do with how you get the information writes down [SEP] If you not only write down the text neatly, but also start it with To illustrate an example with a mind map, to question and understand the content, then this addresses the learning strategy organizing. [SEP] By engaging with the text and also by making connections, organizing helps create a connection between the information (Gold & Hasselhorn, 2017))	691	0.10
Elaboration	A learning tip is mentioned to be connected to the pedagogical concept of elaboration Example: "Als weiterer Tipp wird das Abfragen genannt (It's Leena, 2017). [SEP] Dies ist sinnvoll, da man Lerninhalte erklären muss und somit das eigene Verständnis kontrollieren kann. [SEP] Durch gezielte Fragen kommen hier das Elaborieren und das Langzeitgedächtnis zum Tragen. [SEP] Durch das häufige Abfragen der Verbindungen innerhalb des semantischen Netzwerkes kann leichter auf die Lerninhalte zurückgegriffen werden (Kunter & Trautwein, 2013)." (Another tip is to ask questions (It's Leena, 2017). [SEP] This makes sense since one has to explain learning content and thus be able to check your own understanding. [SEP] Avg targeted questions, elaboration and long-term memory come into play here. [SEP] By frequently querying the connections within the semantic network easier access to the learning content (Kunter & Trautwein, 2013))	440	0.06

Table 4 (continued)

Code	Description	Absolute #	Relative #
Metacognition	A learning tip is mentioned to be connected to the pedagogical concept of metacognition Example: "Im neunten Tipp bezieht sich Annalena auf die Abfrage der erlernten Inhalte (It's Leena, 2017). [SEP] Dies kann sehr hilfreich sein, da Gelerntes frei wiederholt werden und das Wissen abgerufen werden muss. [SEP] Hierbei wirkt auch die metakognitive Lernstrategie der Überwachung mit, denn der aktuelle Lernstand kann dadurch überprüft werden (Gold & Hasselhorn, 2017)." (In the ninth tip, Annalena refers to the query of the learned content (It's Leena, 2017). [SEP] This can be very helpful, since what has been learned must be repeated freely and the knowledge must be retrieved. [SEP] The metacognitive learning strategy of monitoring is also involved here, because the current learning status can be checked (Gold & Hasselhorn, 2017))	567	0.08
Attention	A learning tip is mentioned to be connected to the pedagogical concept of attention Example: "Nach dem Lernen nicht direkt ans Handy gehen: Hier schließe ich mich der Lena an, da ich der Meinung bin, dass nach dem Lernen eine kleine Pause eingelegt werden soll, damit das Gelernte nicht schnell vergessen wird. [SEP] Dies spricht den Vorlesungsinhalt „B)" an, da die gelernten Informationen durch die Ablenkung mit dem Handy beeinträchtigt werden können." (Don't go straight to the cell phone after learning: I agree with Lena here, because I believe that after learning, there should be a short break so that what has been learned is not quickly forgotten. [SEP] This addresses lecture content "B)" as the information learned can be affected by cell phone distraction)	1377	0.20
Working memory	A learning tip is mentioned to be connected to the pedagogical concept of working memory Example: "It's Leena (2017) teilt bei Tipp Nummer vier mit, dass Schüler*innen nach dem Lernen nicht direkt an das Handy gehen sollten. [SEP] Sie meint, dass sonst alles wieder vergessen gehe (It's Leena, 2017). [SEP] Dieser Aspekt fördert das Lernen, da die Aufmerksamkeit des Gelernten sonst direkt verdrängt wird und man sich noch einmal Gedanken zu seinem gelernten machen sollte. [SEP] Ferner würden dabei die Lerninhalte nicht im Arbeitsgedächtnis verbleiben, sondern die Reize, welche dem Handy entnommen wurden (Kunter & Trautwein, 2013)." (It's Leena (2017) states in tip number four that students do not after learning should go straight to the phone. [SEP] She says that otherwise everything will be forgotten again (It's Leena, 2017). [SEP] This aspect promotes learning, since the attention of what has been learned is otherwise direct is repressed and you should think again about what you have learned. [SEP] Also the learning content would not remain in the working memory, but the stimuli, which taken from the mobile phone (Kunter & Trautwein, 2013))	1368	0.20

Table 4 (continued)

Code	Description	Absolute #	Relative #
Long-term memory	A learning tip is mentioned to be connected to the pedagogical concept of long-term memory Example: "Als ersten Tipp erläutert It's Leena (2017) den Aspekt, dass Schüler*innen früh genug anfangen sollten zu lernen. [SEP] Dieser Tipp fördert das Lernen. [SEP] Durch frühzeitiges Anfangen mit dem Lernen wird das semantische Gedächtnis weiter ausgeprägt. [SEP] Außerdem ist es von der Reizaufnahme bis zum Langzeitgedächtnis ein langer Weg, welcher auch in die umgekehrte Richtung abweichen kann. [SEP] Folglich ist das frühe und wiederholende Lernen ein sehr wichtiger Faktor, um sich neues Wissen ausreichend einprägen zu können (Kunter & Trautwein, 2013)." (As a first tip, It's Leena (2017) explains the aspect that students* start early enough should learn. [SEP] This tip encourages learning. [SEP] By starting learning early the semantic memory is further developed. [SEP] Also, it's from stimulus reception long-term memory is a long way, which also goes in the opposite direction may differ. [SEP] Consequently, early and repetitive learning is a very important factor in order to be able to sufficiently memorize new knowledge (Kunter & Trautwein, 2013))	1 588	0.23
Motivation	A learning tip is mentioned to be connected to the pedagogical concept of motivation Example: "Auch die Bedeutung von Motivation erkennt die YouTuberin. [SEP] Sie will in der Schule gute Noten erzielen und ist damit extrinsisch motiviert und arbeitet leistungsorientiert. [SEP] Sie deutet aber auch heraus, dass sie manchmal trotz großen Lernaufwands schlechte Noten erhält, was damit in Verbindung gebracht werden kann, dass ihre Motivation nicht auf einer intrinsischen Basis beruht und sie somit nicht freiwillig lernt. [SEP] Würde sie aus reinem Interesse für ein Schulfach lernen, könnte sie noch mehr und übergreifendere Informationen aufnehmen (Sansone & Harackiewicz, 2000)." (The YouTuber also recognizes the importance of motivation. [SEP] She wants to go to school achieve good grades and is therefore extrinsically motivated and works in a performance-oriented manner. [SEP] But she also points out that sometimes despite a great deal of learning effort gets bad grades, which may be related to her Motivation does not have an intrinsic basis and is therefore not learned voluntarily. [SEP] If she were to study a school subject purely out of interest, she could do even more and include more overarching information (Sansone & Harackiewicz, 2000))	1247	0.18

In addition, the absolute and relative numbers of text segments per assigned code is listed

2012) was calculated. We then used these individual values to determine the overall agreement (0.87). After this step, fundamental disagreements found during coding were resolved through discussion. As the interrater reliability was determined to be high, no additional parallel coding was conducted, and the remaining 598 essays were distributed among the coders.

After the coding, the sentences corresponding to the same learning tips within the same essays were joined into overall text sections. This resulted in a total of 6789 individual segments. These were combined with the codes from the shared *Excel* sheet using *Pandas*, resulting in ten unique codes per section, to acquire a final tabular dataset for multilabel classification.

System Architecture

The system was implemented in a pipeline composed of four components. These are 1) Preprocessing, 2) Segmentation, 3) Analytic scoring and 4) Feedback generation. The input essays are extracted from PDF files which are uploaded to Moodle by the learners and segmented into sentences in the first step. Following this, the essays are segmented into sections addressing the different learning tips. These sections are then scored with regard to the lecture content learners associated a respective tip with. Finally, the predicted scores trigger feedback rules to assemble the final feedback. Finally, thefeedback texts are displayed in a custom Moodle plugin. Figure 2 depicts this pipeline schematically.

Preprocessing

The preprocessing procedures are identical to the steps described in the *Datasets & Coding Rubrics section*. It is intended that learners upload their essays as PDF files. From these, the text is extracted using *PyPDF*. Following this, title page and reference section are removed using heuristics so that only the text remains. The text is then segmented into sentences and tokens using *SoMaJo* (Proisl & Uhrig, 2016). The reason for choosing this tokeniser addresses a comparably novel challenge in processing German texts: dealing with gender-sensitive language. Within our data set, many students used gender-sensitive language. This practice has resulted from academic and societal debates on the role of the grammatical gender of German nouns. It is suspected that "grammatical gender doesn't just refer to gender per se, it does much more: It refers to social expectations to the sexes (gender) and thus on gender in a broader sense" (Nübling, 2018, p. 49). Moreover, empirical results suggest that the grammatical gender of nouns referring to people can evoke stereotypical thinking if no explicit reference to the gender of the described person is provided (Gygax et al., 2008).

For this reason, gender-sensitive language has become an established practice in academic writing to explicitly highlight gender diversity in cases where nouns are intended to address people of multiple genders. It is usually conducted by adding a so-called "gender asterisk" in combination with the suffix

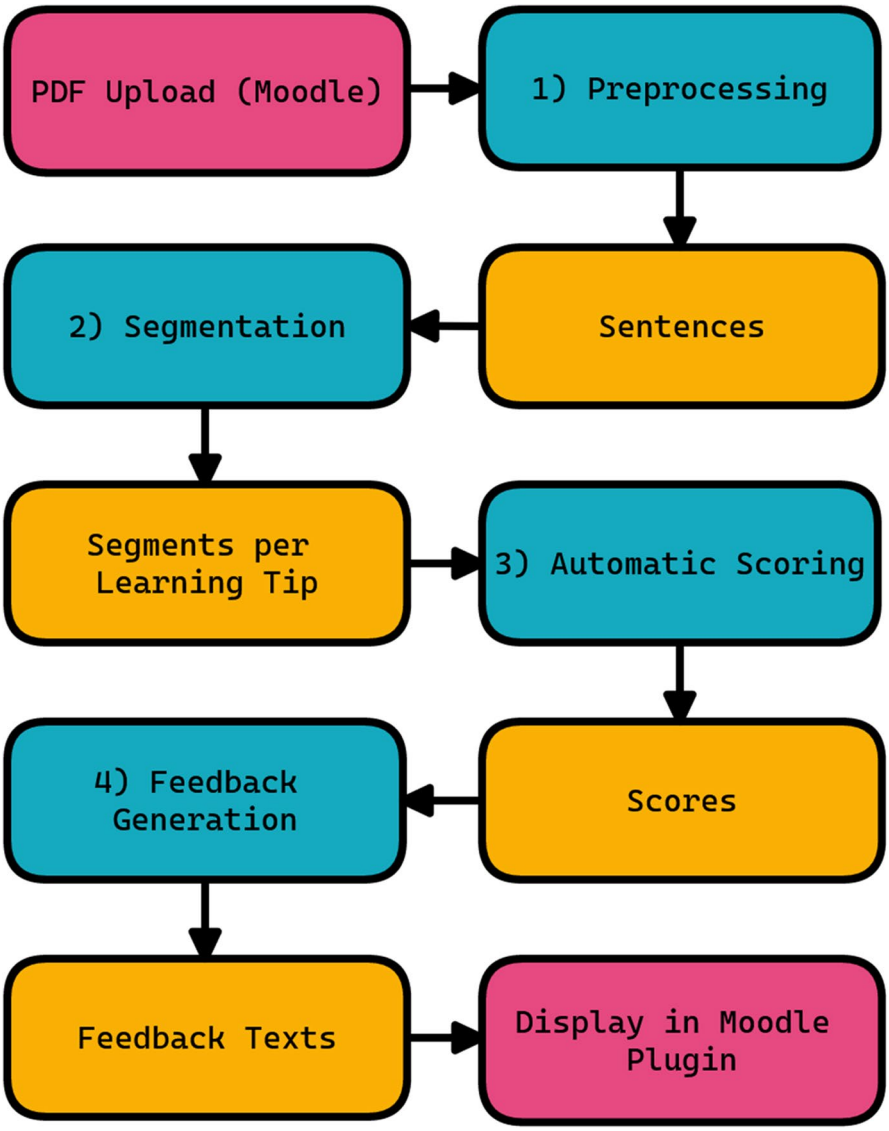

Fig. 2 A schematic illustration of the pipeline steps of our systems

"-innen" to gendered nouns. This is meant to highlight that women and diverse people are included in these nouns. For, following this interpretation, the noun "Lehrer*innen" would refer to teachers of all genders, while the use of "Lehrer" would only refer to male teachers.

When testing different tokenisers for preprocessing the data, we observed problems in correctly processing such nouns. A non-systematic sample analysis

revealed that *SoMaJo* was the only tokeniser that showed no problems processing gender-sensitive language in German. Other tokenisers, such as those provided by *OpenNLP* or *Stanza* (Qi et al., 2020), often interpreted the asterisk as a sentence border, which resulted in badly split sentences.

Semantic Segmentation

Following the preprocessing step, the essays are segmented into multiple sections by the learning tips these individual sections address. This is conducted in the form of a sentence-in-context classification task, where, for each sentence of an essay, it is determined whether it corresponds to a learning tip and, if yes, to which one; if it does not correspond to a tip at all, or if it introduces an additional tip instead (see Table 3).

It is possible that a sentence does not contain any direct information about which tip it corresponds to. However, valuable information is likely hidden in the surrounding sentences. Many of the essays addressed the learning tips in a similar order. This means that surrounding sentences referring to other tips might also give essential hints on the tip a sentence of interest refers to. Therefore,, while a sentence might not contain any direct information on which learning tip it addresses, its context maybe does. In such cases, sliding windows can provide this valuable information to a model. For this reason, we adopted the approach of Zehe et al. (2021) to use a sentence-level sliding window classifier for semantic segmentation so that models can access contextual information from neighbouring sentences during classification. The following equation illustrates the respective input for a sentence s_t at the position t for a context window size of$m$:

$$inp = s_{t-m} \oplus \cdots \oplus s_t \oplus \cdots \oplus s_{t+m} \tag{1}$$

To acquire the resulting final text segments, all sentences corresponding to a particular learning tip are joined into individual text segments in chronological order. The following equation illustrates this step for a set of sentences with $s_{t_k,n}$ being the n -th sentence referring to the k-th tip t_k:

$$inp = s_{t_k,1} \oplus \cdots \oplus s_{t_k,n} \tag{2}$$

GBERT-based Models

We implemented classifiers based on the pre-trained *GBERT* transformer language model (Chan et al., 2020) to segment our learner essays. *GBERT* is a *BERT* model (Devlin et al., 2019) specifically pre-trained for processing German data. *BERT* is a discriminative transformer model which uses a transformer encoder to provide contextual embeddings of the tokens within a given input sentence. *BERT* models are first pre-trained with a large text corpus for masked language modelling and next-sentence prediction, which equips the model with an abstract understanding of the distributional properties of language. For GBERT, this step was carried out with exclusively German data. Depending on the task, the model is then fine-tuned in a second step using

a smaller corpus. For this purpose, input texts are fed into the model to acquire output embeddings. Afterwards, these are pooled in various ways and fed into a final prediction layer, e.g., a linear maximum entropy layer for multiclass classification tasks. The layers learn to encode different formal and semantic aspects of languages (Tenney et al., 2019). This can be leveraged by end users who fine-tune the model for specific tasks.

In our approach we combined GBERT with a sliding window, similar to Zehe et al. (2021). It is implemented via the *BertForSequenceClassification* class of the *Huggingface Transformers* framework (Wolf et al., 2020) and fine-tuned with the regular *Trainer* class this framework provides. *AdamW* (Loshchilov & Hutter, 2019) was used as the optimisation algorithm. As an input prompt, it is given *m* sentences to the left and right in addition to the sentence of interest, all separated by *[SEP]* tokens. This allowed the model to use signals from surrounding sentences to classify a sentence of interest.

BoW Baseline Models

We implemented multiple baseline models combining a *tf-idf*-encoded bag-of-words feature set with different classification algorithms. Each sentence, including context sentences, wasrepresented by a *tf-idf* vector for these baseline classifiers. These were concatenated to form an overall feature vector. We trained three models on top of this feature encoding using three different algorithms: *Random Forests* (Ho, 1995), *Gradient Boosting* (Mason et al., 1999) and *Logistic Regression* classification. We used *Scikit-Learn* (Pedregosa et al., 2011) and *CatBoost* (Prokhorenkova et al., 2018) to implement these systems.

Automatic Scoring

During this step, we predict scores for each learning tip addressed in given input essays. This step consists of two sub-steps. First, we predict which concepts from the lecture are connected to a given learning tip. This is carried out in the form of a multi-label classification problem, as learners might connect multiple concepts to a learning tip. The models are trained to predict the encountered codes for a given input segment corresponding to a learning tip (see Table 4). The inputs for training are generated during the preceding segmentation step. Following this, the predictions are used to calculate the final scores.

Code Prediction: GBERT-based Models

One of the architectures we implemented for predicting codes for input segments is based on *GBERT,* implemented via the *BertForSequenceClassification* class from *Huggingface Transformers* (Wolf et al., 2020) which was fine-tuned for this purpose using the regular *Trainer* class from the same framework. It was trained in *multilabel* mode using the *AdamW* optimiser (Loshchilov & Hutter, 2019). As input, the model used the text segments produced from the previous step. The input sentences of a given text segment were separated by [SEP] tokens.

Code Prediction: T5-based Models

Inspired by the findings from Filighera et al. (2022) that string-to-string mapping encoder-decoder transformers such as *T5* (Raffel et al., 2020) are capable of educational scoring tasks, we also evaluated *T5* for this problem. In contrast to discriminative transformer models such as *BERT*, *T5* is an *encoder-decoder transformer model* consisting of both an *encoder* and a *decoder* component. The model was trained to generate output strings for a given input string.

For this purpose, a given input string is tokenised and transformed into contextualised embeddings using the *encoder* component. The autoregressive decoder component then uses these to generate an output string while attending to the embeddings of the input string. This architecture is usually aimed at tasks such as machine translation or summarisation. However, fitting T5-based models for multi-label classification is possible by simply letting them generate output strings containing labels corresponding to given input strings.

For this reason, in our case, the model wastrained to translate an input segment into an output string listing all detected codes in an arbitrary order, i.e., if a given input segment connects a learning tip to the topics of motivation and long-term memory, the resulting output string of the model would be 'motivation long-term memory </s>'. We used the implementation provided by *Huggingface Transformers* (Wolf et al., 2020) in the form of the *T5ForConditionalGeneration* class. We trained the model using the *AdaFactor* optimiser (Shazeer & Stern, 2018).

Code Prediction: BoW Baseline Models

As for the segmentation step, we implemented baseline systems based on *tf-idf* scores. The rationale is that a range of keywords and phrases signifies the final codes assigned to a respective text segment. *Tf-idf* scores allow models to identify them and identify their respective connections automatically. We used the full input text segments to calculate a feature vector. These were fed to three machine learning algorithms we evaluated for this purpose: *Random Forests* (Ho, 1995), *Gradient Boosting* (Mason et al., 1999) (in the form of one vs. many classifiers) and *Logistic Regression* classification. We used *Scikit-Learn* (Pedregosa et al., 2011) and *CatBoost* (Prokhorenkova et al., 2018) to implement these systems.

Calculation of Scores

For calculating the scores, the codes predicted for a given input segment are matched against a ground truth table denoting the correct concept assignments for each learning tip. This matching process is conducted separately for the three general topic areas addressed by the learning task:

A. Learning strategies (repeating, organising, elaboration)
B. Metacognition, attention, working memory, long-term memory
C. Motivation

This process can be illustrated by the following formulas which calculate the scores as proportion of correctly and incorrectly assigned codes within each content area. A refers to one of the three content areas, T to the set of all learning tips, and t to a learning tip. $C\prime(A,t)$ is the set of codes from the content area A that were correctly assigned to a given learning tip t, with $c\prime$ denoting a given code from this set. Conversely, $C\prime\prime(A,t)$ denotes the set of codes from the content area A that were incorrectly assigned to a given learning tip, with $c\prime\prime$ denoting one of these codes:

$$Corr(A) = \sum_{t}^{T} \sum_{c\prime}^{C\prime(A,t)} 1 \tag{3}$$

$$Incorr(A) = \sum_{t}^{T} \sum_{c\prime\prime}^{C\prime\prime(A,t)} 1 \tag{4}$$

$$Score(A) = \frac{Corr(A)}{Corr(A) + Incorr(A)} \tag{5}$$

This results in scores in the range of $0 \leq Score(A) \leq 1$.

Feedback Generation

Similar to the approach of the *OnTask* system (Pardo et al., 2018), the feedback is generated from templates following hand-crafted rules. The reasoning here is that the feedback we provide addresses a narrow domain and needs to be reliable, which rendered out more experimental approaches of using generative language models for feedback provision. We implemented a custom *Moodle* plugin for this purpose. The feedback was implemented using the *mustache* templating engine[1] packed with the LMS and assembled by rules.

Depending on the scores for the three content areas, students are shown differing messages concerning which parts of the lecture content they understood well and which parts they need to repeat. Students are presented with four general texts, depending on their performance in the content areas. The performance is measured in three levels (low/medium/high), which depend on threshold values limiting the allowed discrepancies between the ground truth table and the predicted labels. A performance for a given content area is counted as low if the respective score is below 0.5. It is counted as medium if this number sits between 0.5 and 0.8 and as high if it sits above. The feedback texts address the following cases:

1. Overall negative feedback: A learner demonstrated a low performance in all three content areas. The learner is advised to revisit all lecture topics.
2. Overall medium feedback: A learner demonstrated a medium performance in all three content areas. The learner is advised to revisit all lecture topics.
3. Overall positive feedback: A learner performed well in all three content areas.

[1] https://mustache.github.io/

4. Mixed feedback: A learner showed a high performance in one or two content areas and a low performance in one or two content areas. The learner is advised to revisit the contents of the content areas for which they demonstrated a low or medium performance.

Figure 3 shows an example of a generated mixed feedback text.

Evaluation

RQ1: Evaluation of the Machine Learning Models

In the first step, we evaluated different segmentation models based on *GBERT-base* and *GBERT-large*, which were fine-tuned for three epochs against the three *TF-IDF* baseline models. The evaluation was done via 5×5 cross-validation, and F1 scores were used as the evaluation metric. It was measured for each of the 12 labels. Table 5 lists the individual results.

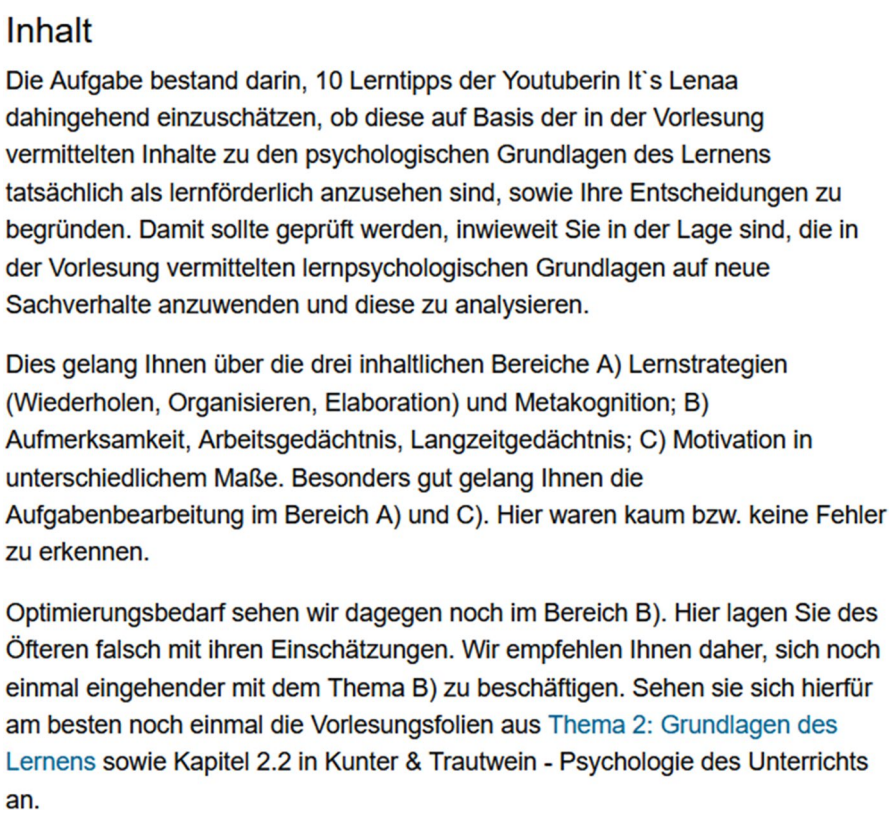

Fig. 3 An example of mixed feedback generated by our method as displayed by our custom Moodle plugin

Table 5 The F1 evaluation results for the different segmentation approaches

Code	BoW logistic regression	BoW random forests	BoW CatBoost	GBERT-b	GBERT-l
Tipp 1	86.83	88.36	90.85	97.75	**98.23**
Tipp 2	83.82	83.11	88.13	97.95	**98.30**
Tipp 3	82.34	82.26	86.94	97.05	**97.54**
Tipp 4	77.54	80.49	84.52	94.80	**95.67**
Tipp 5	81.01	83.18	86.98	96.20	**96.80**
Tipp 6	81.59	82.33	87.40	96.97	**97.50**
Tipp 7	77.89	81.40	84.25	95.11	**96.04**
Tipp 8	81.92	80.87	87.77	97.48	**97.88**
Tipp 9	83.72	83.30	87.86	97.06	**97.57**
Tipp 10	87.27	87.49	90.39	96.15	**96.71**
No Tipp	81.58	83.86	85.13	90.72	**91.14**
Add. Tipp	61.81	62.13	66.36	78.17	**80.12**
Overall (Macro F1)	80.61	81.56	85.55	94.62	**95.32**

The bold entries mark the best result achieved in a respective category

As is visible, all approaches achieved overall high F1 scores > 80.00 for the segmentation step, demonstrating the general feasibility of our segmentation approach. The *Additional Tip* category is an exception, for which all models achieved results significantly worse than the other categories. A possible explanation might be that this category addresses additional learning tips introduced by participants. This means that, compared to the other categories, which address pre-defined learning tips, this category is expected to be more diverse concerning the observed input as the learning tips brought up might be diverse. In addition to this, this category contains by far the fewest examples out of all.

The performance of the baseline models wasstill significantly worse than the performance of the ones based on *GBERT*. This result is expected given the general developments in natural language processing where *BERT*-based models achieved state-of-the-art results for various tasks and that the baseline models operate solely on *TF-IDF*-encoded bag-of-word features. The model based on *GBERT-large* achieved the best results for all 12 labels, although they are very close to the ones achieved by the *GBERT-base* model.

In the next step, we evaluated the approaches for the final coding procedure. For this purpose, we again used 5×5 cross-validation and measured the performance of the models using F1 scores. For this evaluation step, the essays were segmented using the best model from the previous step, which was based on GBERT-large. Table 6 shows the respective results.

The baseline based on logistic regression wassignificantly outperformed by all other models, which reached overall satisfactory F1 scores between 80 and 90. As for the segmentation step, the most potent approach is based on *GBERT-large*. This is followed by the approach based on FLAN-T5-large, which performed slightly worse

Table 6 The F1 evaluation results for the different approaches explored for the final coding step

Code	BoW logistic regression	BoW random forests	BoW CatBoost	FLAN-T5-b	GBERT-b	FLAN-T5-l	GBERT-l
Beneficial	95.09	94.49	95.00	95.97	95.70	95.58	**96.87**
Learning strategy	60.18	60.60	73.68	76.63	76.17	77.37	**78.04**
Repetition	82.46	87.58	88.65	90.71	90.76	91.15	**92.00**
Organisation	59.13	78.67	79.16	87.42	87.05	**91.30**	86.92
Elaboration	42.08	64.37	74.28	87.74	87.88	88.63	**91.83**
Metacognition	41.67	81.25	65.66	88.37	85.92	**93.02**	90.40
Attention	73.10	79.69	80.19	82.93	83.57	83.34	**84.51**
Working memory	78.26	86.55	86.31	88.78	89.76	89.71	**90.36**
Long-term memory	81.93	87.28	87.48	86.24	**89.37**	87.10	89.02
Motivation	79.13	91.02	89.95	91.28	90.96	91.20	**94.14**
Overall (Macro F1)	69.30	81.15	82.04	87.61	87.71	88.38	**89.41**

The bold entries mark the best result achieved in a respective category

and outperformed *GBERT-large* in two categories. The following best approaches based on *GBERT-base* and *FLAN-T5-base* showed a similar performance with an overall F1 difference of only 0.10 points.

The gap between the transformer-based and the baseline models based on *Random Forests* and *CatBoost* was small for the categories with many examples. However, the baselines falled short in the categories with fewer examples such as *Organisation*, *Elaboration* and *Metacognition*. Because the transformers operate on embeddings, i.e. word vectors that encode different task-dependent properties, and have been pre-trained on different corpora, they can better model the semantic relationship between words and phrases, which bag-of-words cannot represent because they rely on indicator words and phrases observed during training. Consequently, transformers learn to better generalise for unseen examples, which likely was the factor that put them ahead in categories with fewer examples in our case.

It is also visible that the predictions for the code *Learning Strategies* deviate from the other results, as the results for this code suggest that it was the most difficult for the models to predict. We could not find a clear reason for this, as the code is not less frequent than the other codes within the training set. A possible explanation could lie in the fact that learning strategies as a general concept is more broad than most of the other concepts addressed by the codes. While other codes address comparably narrow concepts, such as *Organisation*, or were predominately signified

by a small range of keywords- and phrases, as for example for the code *Motivation*, Learning Strategies refers to a broader range of concepts.

RQ2: Evaluation of the Feedback

To evaluate the quality of the provided feedback, we carried out a randomised controlled trial with an additional cohort of students ($N = 257$) who were given the task as a compulsory assignment for the lecture *Introduction to Educational Psychology* at the *Goethe University Frankfurt* in the winter term 2022/2203. Participants were divided into two randomised groups to measure the quality of the feedback. The treatment group ($N = 148$) received highly informative feedback generated by our system and essential feedback on formalities (the *correct usage of the APA citation style*, the *quality of the orthography*, and the *quality of the format*) manually provided by student assistants. In contrast, the control group ($N = 109$) received only this essential feedback on formalities but no feedback related to the content of their assignments.

The feedback was provided with the help of a combination of the strongest models for segmentation and coding determined during the previous evaluation steps, which were the *GBERT-large* models. Both models were retrained on the complete data set for this purpose. All participants received feedback two days after the assignment submission deadline. Table 7 lists the distribution of feedback the participants from the treatment group received. As shown, most students from the treatment group received mixed feedback, i.e., they scored high in some content areas and low in others. No student who handed in an essay failed the content-related part of the task as scored by our system.

For assessing what factors influenced the perception of the feedback, we applied a survey instrument consisting of six different items. These measure comprehensiveness, helpfulness, progress, opportunities for reflection, opportunities for regulation and motivational effects of the feedback. The items reflect the general criteria for highly informative feedback as specified by Wisniewski et al. (2020). The scales for all six items were set to four (*strongly disagree*, *disagree*, *agree*, *strongly agree*), and the survey results for the individual items are used as dependent variables. To reduce noise in the data and to only take serious answers into account, we excluded all participants who filled out the questionnaire in under 10 s ($N = 9$).

In the next step, we defined a set of independent variables whose relationship to the feedback perception we aimed to test. Among them, the central treatment variable is a binary one that encodes whether a participant received the treatment with highly informative feedback. On the other hand, we included three variables encoding whether participants passed the different formal requirements, namely APA,

Table 7 The distribution of feedback types that were generated for the participants from the focus group

Overall negative feedback	Mixed feedback (2 Content areas low)	Mixed feedback (1 Content area low)	Overall mediocre feedback	Overall positive feedback
0	26	99	3	20

Table 8 An overview of the six perception items used to evaluate the provided feedback

Variable	Description	Mean (Std)
Dependent variables		
Comprehensiveness	Item: The feedback is understandable	3.27 (0.76)
Helpfulness	Item: The feedback is helpful for my further learning	3.09 (0.80)
Progress	Item: The feedback makes me aware of my current learning progress	3.10 (0.77)
Reflexion	Item: The feedback helps me to reflect on my learning behaviour so far	3.01 (0.75)
Regulation	Item: The feedback helps me to regulate my learning behaviour if needed	2.91 (0.76)
Motivation	Item: The feedback motivates me to continue learning	3.06 (0.83)
Feedback-related indepentent variables		
Highly informative feedback	The treatment variable, i.e., whether or not a participant was provided with highly informative feedback	0.57 (0.49)
Pass APA	A binary variable encoding whether participants passed the APA requirements. This was manually assessed	0.83 (0.36)
Pass writing	A binary variable encoding whether participants passed the writing-quality-related requirements.This was manually assessed	0.94 (0.23)
Pass formal	A binary variable encoding whether participants passed the formal requirements.This was manually assessed	0.93 (0.23)
Independent control variables		
Usefulness	A scale that measures whether usefulness was a driving factor in choosing teacher studies as a subject	0.61 (0.21)
Pedagogical interest	A scale that measures whether pedagogical interest was a driving factor in choosing teacher studies	0.89 (0.14)
Social factors	A scale that measures whether social factors were a driving factor in choosing teacher studies as a subject	0.56 (0.24)
Academic interest	A scale that measures whether academic interest was a driving factor in choosing teacher studies	0.78 (0.18)
Belief in one's own abilities	A scale that measures whether the belief in one's abilities was a driving factor in choosing teacher studies as a subject	0.75 (0.17)
Easiness	A scale that measures whether easiness was a driving factor in choosing teacher studies as a subject	0.22 (0.20)

format and writing quality, as these also contributed to the feedback texts. Table 8 lists all variables together with their mean and standard deviation. All variables except for the dependent ones are scaled to a range between 0 and 1.

We supplemented the items with control variables measured by the FEMOLA instrument. This instrument was introduced by Pohlmann and Möller (2010). The rationale behind this is that reasons that motivate participants to choose their programme of study might also influence how they perceive the feedback. For example, it is imaginable that participants who chose the teaching profession primarily for financial security might perceive the provided feedback differently than participants who chose it rather for reasons of academic interest.

We then applied ordinary least squares regression analyses to assess the relationship between the dependent and independent variables. Collinearity was assessed by calculating the variance inflation factor of all independent variables. Variables that caused high collinearity (VIF > 1.5) were *Pass Writing*, *Belief in own Abilities,* and *Easiness*. As a consequence, we excluded these variables from the final analysis. Following this, we conducted the main analysis with the help of the *Statsmodels* library (Seabold & Perktold, 2010). Table 9 lists the summative results.

The treatment with highly informative feedback effected the perception of Helpfulness and Motivation. On the one hand, for Helpfulness, the results show a comparably strong positive effect, meaning that the highly informative feedback was perceived as significantly more helpful than the basic feedback. On the other hand, the results show a significant negative effect for Motivation, meaning that the highly informative feedback is perceived as less motivating compared to the basic one.

However, the strongest effects could not be observed for the treatment variable. Instead, they were visible for the variable encoding whether a student passed the APA requirements. Passing the APA requirements strongly affected the perception of Comprehensiveness, Motivation, Regulation, Progress and Helpfulness. These effects were all positive, meaning that passing the APA requirement has an overall stronger influence on the dependent variables than the treatment variable, i.e., participants were likely more influenced by whether they passed the APA requirements than by the treatment variable.

Discussion

RQ1

Concerning RQ 1 (To what degree can we automate the analytic scoring procedures needed to provide highly informative feedback to learners concerning the content of their essays?), it can be stated that the approaches tested performed well in nearly all evaluation categories. The best approaches for the segmentation and coding steps were based on GBERT-large, an expected result given that transformer-based models are state-of-the-art for many NLP tasks. A more interesting result is that the encoder-decoder-based *T5* architecture, which, in our case, translates an input string containing an essay segment into an output string containing a list of the predicted codes, performed close to the purely discriminative BERT-based models.

Table 9 The results of the ordinary least squares analysis

Variable	R-squared	*F*	Coefficient	Std Err	ANOVA	[0.025	0.975]
Comprehensiveness	0.105	***4.058******	-	-	-	-	-
Constant			2.6077	0.419	***6.226*****	1.783	3.433
HIF			−0.1864	0.096	−1.943	−0.375	0.003
Pass APA			0.3066	0.116	***2.638****	0.078	0.536
Pass formal			0.5582	0.199	***2.803****	0.166	0.951
Usefulness			−0.3126	0.235	−1.327	−0.776	0.151
Pedagogical interest			0.3957	0.356	1.111	−0.306	1.097
Social factors			0.1885	0.202	0.932	−0.210	0.587
Academic interest			−0.3327	0.278	−1.198	−0.880	0.214
Helpfulness	0.168	***7.045*****	-	-	-	-	-
Constant			1.4602	0.422	***3.458*****	0.628	2.292
HIF			0.3352	0.097	***3.473*****	0.145	0.525
Pass APA			0.6482	0.117	***5.535*****	0.418	0.879
Pass formal			0.3660	0.201	1.825	−0.029	0.761
Usefulness			0.4190	0.237	1.770	−0.047	0.885
Pedagogical interest			0.5347	0.358	1.492	−0.171	1.241
Social factors			−0.1436	0.203	−0.709	−0.543	0.255
Academic interest			−0.0795	0.279	−0.285	−0.629	0.470
Progress	0.094	***3.570****	-	-	-	-	-
Constant			1.7906	0.426	***4.207*****	0.952	2.629
HIF			0.0538	0.098	0.551	−0.138	0.246
Pass APA			0.4853	0.118	***4.111*****	0.253	0.718
Pass formal			0.3246	0.202	1.605	−0.074	0.723
Usefulness			0.1175	0.239	0.492	−0.353	0.589
Pedagogical interest			0.2931	0.362	0.810	−0.420	1.006
Social factors			0.2467	0.205	1.201	−0.158	0.651
Academic interest			0.1614	0.282	0.572	−0.394	0.717
Reflexion	0.060	***2.188****	-	-	-	-	-
Constant			2.3577	0.422	***5.585*****	1.526	3.189
HIF			0.1351	0.097	1.391	−0.056	0.327
Pass APA			0.3408	0.117	***2.910*****	0.110	0.571
Pass formal			0.3348	0.200	1.670	−0.060	0.730
Usefulness			0.2177	0.239	0.911	−0.253	0.688
Pedagogical interest			−0.4625	0.360	−1.286	−1.171	0.246
Social factors			0.0561	0.205	0.274	−0.347	0.459
Academic interest			0.3145	0.281	1.120	−0.238	0.867
Regulation	0.043	1.549	-	-	-	-	-
Constant			2.2321	0.434	***5.143*****	1.377	3.087
HIF			0.1073	0.100	1.074	−0.090	0.304
Pass APA			0.2346	0.120	1.949	−0.003	0.472
Pass formal			0.3936	0.206	1.910	−0.012	0.800
Usefulness			0.1291	0.246	0.525	−0.355	0.613

Table 9 (continued)

Variable	R-squared	*F*	Coefficient	Std Err	ANOVA	[0.025	0.975]
Pedagogical interest			−0.3788	0.370	−1.025	−1.107	0.349
Social factors			0.1667	0.210	0.792	−0.248	0.581
Academic interest			0.3053	0.289	1.058	−0.263	0.874
Motivation	0.143	***5.700******	-	-	-	-	-
Constant			1.5736	0.449	***3.501****	0.688	2.459
HIF			−0.2366	0.103	***−2.289****	−0.440	−0.033
Pass APA			0.5547	0.125	***4.452*****	0.309	0.800
Pass formal			0.3556	0.213	1.667	−0.065	0.776
Usefulness			0.5035	0.254	***1.978****	0.002	1.005
Pedagogical interest			0.2845	0.382	0.744	−0.468	1.037
Social factors			0.0859	0.218	0.395	−0.343	0.515
Academic interest			0.3134	0.298	1.051	−0.274	0.901

*: $p < 0.05$; **: $p < 0.01$; ***: $p < 0.001$

As Filighera et al. (2022) showed that T5 is, to a certain degree, able to generate individual feedback texts for short responses, it is imaginable to use it for such a feedback generation step in future work. I.e., one could use T5 to generate short texts on why linking certain concepts from the lecture to a specific string was wrong. These could then be combined with the overall feedback to provide participants with more detailed information on their performance. However, before purely automatically generated feedback can be used in a practical context, an open question needs to be answered: the question of alignment. Lim et al. (2021) claim that feedback should clarify performance, facilitate self-assessment, deliver high-quality information, encourage dialogue, promote motivational beliefs and illustrate the gap between achieved and desired performance. Consequently, it needs to be assessed for purely generative feedback models what is needed in terms of architecture, prompting and training data to meet these requirements.

RQ2

Concerning RQ2 (How is the provided highly informative feedback perceived by the learners?), it can be stated that our template-based feedback was perceived positively, as indicated by the results in Table 9. We conducted a randomised controlled trial to find significant effects that contribute to the perception of the feedback, which was measured through six single items that were used as dependent variables. For this purpose, we conducted an ordinary least squares analysis. In our case, the treatment variable was whether or not a participant received highly informative. The provision of highly informative feedback had a positive effect on Helpfulness and a negative effect on Motivation.

Regarding motivation, the highly informative feedback is perceived as less motivating than the basic feedback. A possible explanation could lie in the *self-determination theory* (SDT) (ten Cate, 2013; Fong & Schallert, 2023). *Self-determination*

theory was introduced by Ryan and Deci (2000). It states that the intrinsic motivation of humans is linked to three core psychological needs: feelings of competence, autonomy, and relatedness. As feedback points out discrepancies between the current performance of a learner and a desired goal, "*SDT predicts that it will not help in developing a feeling of competence, pride, and consequently, intrinsic motivation. At best, the addition of strengths has a balancing effect to make trainees not feel depressive about their overall competence*" (ten Cate, 2013).

The highly informative feedback gives participants more details on task performance than the minimal feedback. This also bears the possibility that more discrepancies between participants' actual and perceived performance will be pointed out. If these discrepancies are too large, and the resulting feedback is unexpectedly negative, this might hurt participants' feelings of competence, decreasing their intrinsic motivation. This assumption is also in line with empirical results. Fong et al. (2018), who conducted a meta-analysis on the effect of feedback on intrinsic motivation, found that, in many cases, feedback highlighting improvement areas can hurt learners' intrinsic motivation.

Interestingly, the question of whether participants passed the formal APA requirements of the assignment had a stronger overall effect on the different dependent variables compared to the highly informative feedback. While we cannot causally explain this effect, this likely stems from our participants being predominately first-semester students. This might have been one of the first assignments requiring many of them to cite sources properly. Since plagiarism is highly sanctioned throughout their studies, positive reinforcement regarding whether they cited properly might lead to relief, which might explain the stronger effects.

Conclusion

To summarise, we presented and evaluated an essay writing learning activity coupled with an automated feedback system that provided learners with feedback on the correctness of the content of their writing. For this purpose, we also collected and coded a custom data set composed of 698 essays written in German. This dataset was used for evaluating the performance of the individual machine-learning components constituting this pipeline. This evaluation was highly successful, demonstrating the feasibility of the proposed approaches. The results are in line with the general success of transformer language models.

Moreover, we evaluated how learners perceived the generated feedback by setting up a randomised control study with a cohort of learners. This revealed overall positive effects regarding highly informative feedback and that our participants were, on average, more influenced by whether they passed the formal APA requirements. Given that the participants were students from an introductory lecture, this is an expected result as they do not have extensive experience in paper writing. Nonetheless, the highly informative feedback positively affected helpfulness and reflection.

Limitations and Future Work

A clear limitation of our approach is that the feedback is generated solely through templates. While this guarantees stable feedback texts compared to purely generative models as used, for example, by Filighera et al. (2022), it also limits the possibilities for personalising the feedback. Ideally, feedback should address learners' mistakes in more detail and give them individual explanations of what they did wrong instead of just directing them to the appropriate lecture content. This is only feasible to a certain degree with purely template-based feedback and requires practitioners to consider all kinds of possible mistakes to prepare appropriate templates. This, in turn, hinders the scalability of such feedback. For this reason, our future work will mainly address how we can stabilise transformer-based feedback generation methods to allow for more individualised feedback. These could then be used to either populate placeholders within feedback templates or completely generate the feedback from scratch.

To release the full potential of highly informative feedback, the teaching concept, the course's intended learning outcomes and the design of the digital learning environment need to be closely aligned (Biggs, 1996). That means that after specifying a course's learning outcomes, it is not only necessary to define the assessment of these outcomes but also think about potential learning indicators from the data that provide valuable insights into the learner's state (Schmitz et al., 2022). Examples of well-designed learning environments that provide learning activities that directly send relevant indicators of learning are manifold (Ahmad et al., 2022). A future direction of work could thus also involve research on integrating these indicators into feedback generation models to provide feedback on learning processes in combination with feedback on the outcome. Integrating learning indicators into transformer-based feedback systems could provide additional information that could be leveraged for feedback generation.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability The data is available from the authors on request after signing an NDA agreement for purely scientific, non-commercial purposes. Any attempts at de-annonymizing the data are strictly forbidden.

Declarations

Conflicting Interests The authors have no relevant financial or non-financial interests to disclose.

References

Ahmad, A., Schneider, J., Griffiths, D., Biedermann, D., Schiffner, D., Greller, W., & Drachsler, H. (2022). Connecting the dots – A literature review on learning analytics indicators from a learning design perspective. *Journal of Computer Assisted Learning*. https://doi.org/10.1111/jcal.12716

Alobaidi, O. G., Crockett, K. A., O'Shea, J. D., & Jarad, T. M. (2013). Abdullah: An intelligent Arabic conversational tutoring system for modern Islamic education. In *Proceedings of the World Congress on Engineering* (Vol. 2).

Andersen, N., & Zehner, F. (2021). shinyReCoR: A shiny application for automatically coding text responses using R. *Psych, 3*(3), 422–446. https://doi.org/10.3390/psych3030030

Andersen, N., Zehner, F., & Goldhammer, F. (2022). Semi-automatic coding of open-ended text responses in large-scale assessments. *Journal of Computer Assisted Learning, 39*(3), 841–854. https://doi.org/10.1111/jcal.12717

Bai, X., & Stede, M. (2022). A survey of current machine learning approaches to student free-text evaluation for intelligent tutoring. *International Journal of Artificial Intelligence in Education*. https://doi.org/10.1007/s40593-022-00323-0

Barbaresi, A. (n.d.). German-NLP: Curated list of open-access/open-source/off-the-shelf resources and tools developed with a particular focus on German. *GitHub*. Retrieved September, 2023, from https://github.com/adbar/German-NLP

Beseiso, M., & Alzahrani, S. (2020). An empirical analysis of BERT embedding for automated essay scoring. *International Journal of Advanced Computer Science and Applications*, *11*(10), 204–210.

Bexte, M., Horbach, A., & Zesch, T. (2022). Similarity-based content scoring - How to make S-BERT keep up with BERT. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)* (pp. 118–123). https://doi.org/10.18653/v1/2022.bea-1.16

Biggs, J. (1996). Enhancing teaching through constructive alignment. *Higher Education, 32*(3), 347–364.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research, 3*(Jan), 993–1022.

Burrows, S., Gurevych, I., & Stein, B. (2015). The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education, 25*, 60–117. https://doi.org/10.1007/s40593-014-0026-8

Cahill, L., & Gazdar, G. (1999). German noun inflection. *Journal of Linguistics, 35*(1), 1–42.

Camus, L., & Filighera, A. (2020). Investigating transformers for automatic short answer grading. In I. I. Bittencourt, M. Cukurova, K. Muldner, R. Luckin, & E. Millán (Eds.), *Artificial intelligence in education* (pp. 43–48). Springer International Publishing. https://doi.org/10.1007/978-3-030-52240-7_8

Cavalcanti, A. P., Barbosa, A., Carvalho, R., Freitas, F., Tsai, Y.-S., Gašević, D., & Mello, R. F. (2021). Automatic feedback in online learning environments: A systematic literature review. *Computers and Education: Artificial Intelligence, 2*, 100027. https://doi.org/10.1016/j.caeai.2021.100027

Chan, B., Schweter, S., & Möller, T. (2020). German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics* (pp. 6811–6822). International Committee on Computational Linguistics. https://doi.org/10.18653/v1/2020.coling-main.598

Chen, H., & He, B. (2013). Automated essay scoring by maximizing human-machine agreement. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing* (pp. 1741–1752). Association for Computational Linguistics. https://aclanthology.org/D13-1180/

Choi, F. (2000). Advances in domain independent linear text segmentation. In *1st Meeting of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics. https://aclanthology.org/A00-2004/

Clahsen, H., Marcus, G., Bartke, S., & Wiese, R. (1996). Compounding and inflection in German child language. *Yearbook of morphology 1995*, 115-142.

Cozma, M., Butnaru, A., & Ionescu, R. T. (2018). Automated essay scoring with string kernels and word embeddings. In I. Gurevych & Y. Miyao (Eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* (Volume 2: Short Papers) (pp. 503–509). https://doi.org/10.18653/v1/P18-2080

Crossley, S. A., Roscoe, R. D., & McNamara, D. S. (2011). Predicting human scores of essay quality using computational indices of linguistic and textual features. In *Proceedings of the 15th*

International Conference on Artificial Intelligence in Education, AIED 2011, Auckland, New Zealand, June 28 - July 2011 (pp. 438–440). https://doi.org/10.1007/978-3-642-21869-9_62

Crossley, S. A., Allen, L. K., Snow, E. L., & McNamara, D. S. (2015). Pssst... textual features... there is more to automatic essay scoring than just you! In *Proceedings of the Fifth International Conference on Learning Analytics and Knowledge* (pp. 203–207). https://doi.org/10.1145/2723576.2723595

Deeva, G., Bogdanova, D., Serral, E., Snoeck, M., & De Weerdt, J. (2021). A review of automated feedback systems for learners: Classification framework, challenges and opportunities. *Computers & Education, 162*, 10. https://doi.org/10.1016/j.compedu.2020.104094

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). 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* (pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

Di Mitri, D., Gombert, S., & Karademir, O. (2022). Reflecting on the actionable components of a model for augmented feedback. In *Proceedings of the Second International Workshop on Multimodal Immersive Learning Systems (MILeS 2022) at the Seventeenth European Conference on Technology Enhanced Learning (EC-TEL 2022)* (pp. 45–50). CEUR Workshop Proceedings. http://ceur-ws.org/Vol-3247/paper8.pdf

Durrell, M. (2006). Germanic Languages. *In Encyclopedia of Language & Linguistics* (pp. 53–55). Elsevier.

Dzikovska, M., Nielsen, R., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., Clark, P., Dagan, I., & Dang, H. (2013). SemEval-2013 Task 7: The joint student response analysis and 8th recognizing textual entailment challenge. In *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)* (pp. 263–274). Association for Computational Linguistics.

Filighera, A., Parihar, S., Steuer, T., Meuser, T., & Ochs, S. (2022). Your answer is incorrect... Would you like to know why? Introducing a bilingual short answer feedback dataset. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 8577–8591). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.587

Firoozi, T., Mohammadi, H., & Gierl, M. J. (2023). Using active learning methods to strategically select essays for automated scoring. *Educational Measurement: Issues and Practice, 42*(1), 34–43.

Fong, C. J., Patall, E. A., Vasquez, A. C., & Stautberg, S. (2018). A Meta-analysis of negative feedback on intrinsic motivation. *Educational Psychology Review, 31*(1), 121–162. https://doi.org/10.1007/s10648-018-9446-6

Fong, C. J., & Schallert, D. L. (2023). "Feedback to the future": Advancing motivational and emotional perspectives in feedback research. *Educational Psychologist, 0*(0), 1–16. https://doi.org/10.1080/00461520.2022.2134135

Gold, A., & Hasselhorn, M. (2017). Pädagogische Psychologie (4th ed.) [PDF]. Kohlhammer.

Gombert, S., Di Mitri, D., Karademir, O., Kubsch, M., Kolbe, H., Tautz, S., …, & Drachsler, H. (2022). Coding energy knowledge in constructed responses with explainable NLP models. *Journal of Computer Assisted Learning, 39*(3), 767–786. https://doi.org/10.1111/jcal.12767

Grünewald, S., Friedrich, A., & Kuhn, J. (2021). Applying Occam's razor to transformer-based dependency parsing: What works, what doesn't, and what is really necessary. *IWPT, 2021*, 131–140.

Gygax, P., Gabriel, U., Sarrasin, O., Oakhill, J., & Garnham, A. (2008). Generically intended, but specifically interpreted: When beauticians, musicians, and mechanics are all men. *Language and Cognitive Processes, 23*(3), 464–485.

Haiden, M. (1997). Verbal Inflection and the Structure of IP in German. *GAGL: Groninger Arbeiten zur germanistischen Linguistik*, (41), 77–106.

Hattie, J. A. (2009). *Visible learning: A synthesis of 800+ meta-analyses on achievement*. Routledge.

Hattie, J., & Timperley, H. (2007). The Power of Feedback. *Review of Educational Research, 77*(1), 81–112. https://doi.org/10.3102/003465430298487

Hearst, M. A. (1997). Text tiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, *23*(1), 33–64. https://aclanthology.org/J97-1003.pdf

Ho, T. K. (1995). Random decision forests. In *Proceedings of the 3rd International Conference on Document Analysis and Recognition* (Vol. 1, pp. 278–282). IEEE. https://doi.org/10.1109/ICDAR.1995.598994

Horbach, A., Scholten-Akoun, D., Ding, Y., & Zesch, T. (2017). Fine-grained essay scoring of a complex writing task for native speakers. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics. https://doi.org/10.18653/v1/w17-5040

Horbach, A., Laarmann-Quante, R., Liebenow, L., Jansen, T., Keller, S., Meyer, J., …, & Fleckenstein, J. (2022). Bringing automatic scoring into the classroom - measuring the impact of automated analytic feedback on student writing performance. In *Proceedings of the 11th Workshop on NLP for Computer Assisted Language Learning* (pp. 72–83). https://aclanthology.org/2022.nlp4call-1.8

It's Leena. (2017). Zehn Lerntipps für die Schule [Video]. YouTube. Retrieved January, 2023, from https://www.youtube.com/watch?v=wqCe5KQqhxs

Ke, Z., & Ng, V. (2019). Automated essay scoring: A survey of the state of the art. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence* (pp. 6300–6308). https://doi.org/10.24963/ijcai.2019/879

Klie, J.-C., Bugert, M., Boullosa, B., Eckart de Castilho, R., & Gurevych, I. (2018). The INCEpTION platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of System Demonstrations of the 27th International Conference on Computational Linguistics* (pp. 9–13). International Committee on Computational Linguistics. https://aclanthology.org/C18-2002

Klie, J.-C., & Eckart de Castilho, R. (n.d.).*DKPro Cassis - Reading and Writing UIMA CAS Files in Python*. Retrieved August 2023, from https://doi.org/10.5281/zenodo.3994108

Kunter, M., & Trautwein, U. (2013). *Psychologie des Unterrichts*. Stuttgart, Germany: UTB.

Landauer, T. K., Laham, D., Foltz., P. (2003). Automated scoring and annotation of essays with the intelligent essay assessor. *In Shermis, M. D., & Burstein, J. (Eds.): Automated essay scoring: A cross-disciplinary perspective*. Routledge.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes, 25*(2–3), 259–284. https://doi.org/10.1080/01638539809545028

Li, J., & Hovy, E. (2014). A model of coherence based on distributed sentence representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* (pp. 2039–2048). Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1218

Li, J., & Jurafsky, D. (2017). Neural net models of open-domain discourse coherence. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 198–209). Association for Computational Linguistics. https://doi.org/10.18653/v1/D17-1019

Lim, L. A., Dawson, S., Gašević, D., Joksimović, S., Pardo, A., Fudge, A., & Gentili, S. (2021). Students' perceptions of, and emotional responses to, personalised learning analytics-based feedback: An exploratory study of four courses. *Assessment & Evaluation in Higher Education, 46*(3), 339–359.

Livingston, S. A. (2009). Constructed-response test questions: Why we use them; How we score them. In *R&D Connections. Number 11*. Educational Testing Service. https://eric.ed.gov/?id=ED507802

Loshchilov, I., & Hutter, F. (2019). Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6–9, 2019*. OpenReview.net.

Mason, L., Baxter, J., Bartlett, P., & Frean, M. (1999). Boosting algorithms as gradient descent. In S. A. Solla, T. K. Leen, & K.-R. Müller (Eds.), *Advances in neural information processing systems 12* (pp. 512–518). MIT Press.

Mayfield, E., & Black, A. W. (2020). Should you fine-tune BERT for automated essay scoring? In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications* (pp. 155–164). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.bea-1.15

McDonald, J., Knott, A., Stein, S., & Zeng, R. (2013). An empirically-based, tutorial dialogue system: Design, implementation and evaluation in a first year health sciences course. In *ASCILITE-Australian Society for Computers in Learning in Tertiary Education Annual Conference* (pp. 562–572). Australasian Society for Computers in Learning in Tertiary Education.

McHugh, M. L. (2012). Interrater reliability: The kappa statistic. *Biochemia Medica, 22*(3), 276–282.

McNamara, D. S., Crossley, S. A., Roscoe, R. D., Allen, L. K., & Dai, J. (2015). A hierarchical classification approach to automated essay scoring. *Assessing Writing* (Vol. 23, pp. 35–59). Elsevier BV. https://doi.org/10.1016/j.asw.2014.09.002

Misra, H., Yvon, F., Cappé, O., & Jose, J. (2011). Text segmentation: A topic modeling perspective. *Information Processing & Management, 47*(4), 528–544. https://doi.org/10.1016/j.ipm.2010.11.008

Mousavinasab, E., Zarifsanaiey, N., Niakan Kalhori, S. R., Rakhshan, M., Keikha, L., & Ghazi Saeedi, M. (2021). Intelligent tutoring systems: A systematic review of characteristics, applications, and evaluation methods. *Interactive Learning Environments, 29*(1), 142–163. https://doi.org/10.1080/10494820.2018.1558257

Nachtigall, V., Serova, K., & Rummel, N. (2020). When failure fails to be productive: Probing the effectiveness of productive failure for learning beyond stem domains. *Instructional Science*. https://doi.org/10.1007/s11251-020-09525-2

Narciss, S. (2008). Feedback strategies for interactive learning tasks. In J. M. Spector, M. D. Merrill, J. van Merriënboer, & D. M. Driscoll (Eds.), *Handbook of research on educational communications and technology* (pp. 125–144). Routledge.

Nelson, T. O., & Narens, L. (1994). Why investigate Metacognition. In J. Metcalfe & A. P. Shimamura (Eds.), *Metacognition. Knowing about knowing* (pp. 1–25). Cambridge University Press.

Nguyen, H., & Litman, D. (2018). Argument mining for improving the automated scoring of persuasive essays. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*. https://doi.org/10.1609/aaai.v32i1.12046

Nicol, D. J., & Macfarlane-Dick, D. (2006). Formative assessment and self-regulated learning: A model and seven principles of good feedback practice. *Studies in Higher Education, 31*(2), 199–218. https://doi.org/10.1080/03075070600572090

Nübling, D. (2018). Und ob das Genus mit dem Sexus. Genus verweist nicht nur auf Geschlecht, sondern auch auf die Geschlechterordnung. *Sprachreport, 34*(3), 44–50.

Ortmann, K., Roussel, A., & Dipper, S. (2019). Evaluating Off-the-Shelf NLP Tools for German. *Proceedings of the 15th Conference on Natural Language Processing, KONVENS 2019, Erlangen, Germany, October 9–11, 2019*. https://corpora.linguistik.uni-erlangen.de/data/konvens/proceedings/papers/KONVENS2019_paper_55.pdf

Page, E. B. (1967). Grading essays by computer: Progress report. In *Proceedings of the Invitational Conference on Testing Problems* (pp. 87–100).

Pardo, A., Bartimote, K., Shum, S. B., Dawson, S., Gao, J., Gašević, D., ..., & Vigentini, L. (2018). OnTask: Delivering data-informed, personalized learning support actions. *Journal of Learning Analytics*, *5*(3), 235–249.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., …, & Duchesnay, É. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Phandi, P., Chai, K. M. A., & Ng, H. T. (2015). Flexible Domain Adaptation for Automated Essay Scoring Using Correlated Linear Regression. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 446–452). Association for Computational Linguistics. https://doi.org/10.18653/v1/d15-1049

Pohlmann, B., & Möller, J. (2010). Fragebogen zur Erfassung der Motivation für die Wahl des Lehramtsstudiums (FEMOLA). *Zeitschrift für pädagogische Psychologie, 24*(1), 73–84.

Proisl, T., & Uhrig, P. (2016). SoMaJo: State-of-the-art tokenization for German web and social media texts. In *Proceedings of the 10th web as corpus workshop* (pp. 75–78). Association for Computational Linguistics. https://doi.org/10.18653/v1/w16-2607

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased boosting with categorical features. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems 31* (pp. 6639–6649). Curran Associates Inc.

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th annual meeting of the association for computational linguistics: System demonstrations* (pp. 101–108). https://doi.org/10.18653/v1/2020.acl-demos.14

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., …, & Liu, P. J. (2020). Exploring the Limits of transfer learning with a unified text-to-text transformer. *Journal of Machinge Learning Research*, *21*(1), 1–67.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using siamese BERT-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3982–3992. https://doi.org/10.18653/v1/D19-1410

Riedl, M., & Biemann, C. (2012). TopicTiling: A text segmentation algorithm based on LDA. In *Proceedings of ACL 2012 Student Research Workshop* (pp. 37–42). Association for Computational Linguistics. https://aclanthology.org/W12-3307

Riedl, M., & Biemann, C. (2016). Unsupervised compound splitting with distributional semantics rivals supervised methods. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 617–622. https://doi.org/10.18653/v1/N16-1075

Rodriguez, P. U., Jafari, A., & Ormerod, C. M. (2019). Language models and automated essay scoring. *arXiv preprint* arXiv:1909.09482. *https://doi.org/10.48550/arXiv.1909.09482*

Ryan, R. M., & Deci, E. L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist, 55*(1), 68. https://doi.org/10.1037/0003-066X.55.1.68

Sansone, C., & Harackiewicz, J. M. (Eds.). (2000). Intrinsic and extrinsic motivation: The search for optimal motivation and performance. Academic Press.

Schmitz, M., Scheffel, M., Bemelmans, R., & Drachsler, H. (2022). FoLA2 — a method for co-creating learning analytics-supported learning design. *Journal of Learning Analytics, 9*(2), 265–281. https://doi.org/10.18608/jla.2022.7643

Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with python. In *Proceedings of the 9th python in science conference* (Vol. 57, No. 61, pp. 10–25080).

Shazeer, N., & Stern, M. (2018). Adafactor: Adaptive learning rates with sublinear memory cost. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10–15, 2018* (pp. 4603–4611). PMLR.

Song, X., Salcianu, A., Song, Y., Dopson, D., & Zhou, D. (2021). Fast WordPiece tokenization. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2089–2103. https://doi.org/10.18653/v1/2021.emnlp-main.160

Sung, C., Dhamecha, T., Saha, S., Ma, T., Reddy, V., & Arora, R. (2019). Pre-training BERT on domain resources for short answer grading. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 6071–6075). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1628

Taghipour, K., & Ng, H. T. (2016). A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on empirical methods in natural language processing* (pp. 1882–1891). Association for Computational Linguistics. https://doi.org/10.18653/v1/d16-1193

ten Cate, O. T. J. (2013). Why receiving feedback collides with self determination. *Advances in Health Sciences Education, 18*, 845–849. https://doi.org/10.1007/s10459-012-9401-0

Tenney, I., Das, D., & Pavlick, E. (2019). BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 4593–4601). Association for Computational Linguistics. https://doi.org/10.18653/v1/p19-1452

Vajjala, S. (2018). Automated assessment of non-native learner essays: Investigating the role of linguistic features. *International Journal of Artificial Intelligence in Education, 28*(1), 79–105. https://doi.org/10.1007/s40593-017-0142-3

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., …, & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 30) (pp. 1–11).

Winne, P. H., & Hadwin, A. F. (2008). The weave of motivation and self-regulated learning. In D. H. Schunk & B. J. Zimmerman (Eds.), *Motivation and self-regulated learning: Theory, research, and applications* (pp. 297–314). Lawrence Erlbaum Associates Publishers.

Wisniewski, B., Zierer, K., & Hattie, J. (2020). The power of feedback revisited: A meta-analysis of educational feedback research. *Frontiers in Psychology, 10*, 1–14. https://doi.org/10.3389/fpsyg.2019.03087

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., …, & Rush, A. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: System demonstrations* (pp. 38–45). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-demos.6

Yannakoudakis, H., Briscoe, T., & Medlock, B. (2011). A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th annual meeting of the association for computational*

linguistics: Human language technologies (pp. 180–189). Association for Computational Linguistics. https://aclanthology.org/P11-1019

Yannakoudakis, H., & Briscoe, T. (2012). Modeling coherence in ESOL learner texts. In *Proceedings of the seventh workshop on building educational applications using NLP* (pp. 33–43). Association for Computational Linguistics. https://aclanthology.org/W12-2004

Zehe, A., Konle, L., Dümpelmann, L. K., Gius, E., Hotho, A., Jannidis, F., …, & Wiedmer, N. (2021). Detecting scenes in fiction: A new segmentation task. In *Proceedings of the 16th conference of the European chapter of the association for computational linguistics: Main volume* (pp. 3167–3177). https://doi.org/10.18653/v1/2021.eacl-main.276

Zesch, T., Wojatzki, M., & Scholten-Akoun, D. (2015). Task-independent features for automated essay grading. In *Proceedings of the tenth workshop on innovative use of NLP for building educational applications* (pp. 224–232). Association for Computational Linguistics. https://doi.org/10.3115/v1/w15-0626

Zesch, T., Horbach, A., & Zehner, F. (2023). To score or not to score: Factors influencing performance and feasibility of automatic content scoring of text responses. *Educational measurement: Issues and practice.* Wiley. https://doi.org/10.1111/emip.12544

Authors and Affiliations

Sebastian Gombert[1] · Aron Fink[2] · Tornike Giorgashvili[2] · Ioana Jivet[1,2] · Daniele Di Mitri[1] · Jane Yau[1] · Andreas Frey[2] · Hendrik Drachsler[1,2,3]

✉ Sebastian Gombert
s.gombert@dipf.de

Aron Fink
a.fink@psych.uni-frankfurt.de

Tornike Giorgashvili
giorgashvili@sd.uni-frankfurt.de

Ioana Jivet
i.jivet@dipf.de

Daniele Di Mitri
d.dimitri@dipf.de

Jane Yau
j.yau@dipf.de

Andreas Frey
frey@psych.uni-frankfurt.de

Hendrik Drachsler
h.drachsler@dipf.de

[1] DIPF | Leibniz Institute for Research and Information in Education, Frankfurt am Main, Germany

[2] Goethe University, Frankfurt am Main, Germany

[3] Open University of the Netherlands, Heerlen, Netherlands