Int J Artif Intell Educ
https://doi.org/10.1007/s40593-023-00383-w

ARTICLE

Short-Answer Grading for German: Addressing the Challenges

Ulrike Padó[1] · **Yunus Eryilmaz**[1] · **Larissa Kirschner**[1]

Accepted: 19 November 2023

Abstract

Short-Answer Grading (SAG) is a time-consuming task for teachers that automated SAG models have long promised to make easier. However, there are three challenges for their broad-scale adoption: A **technical** challenge regarding the need for high-quality models, which is exacerbated for languages with fewer resources than English; a **usability** challenge in adapting high-quality research prototypes to the needs of non-expert users, and a **trust** challenge in communicating the abilities and limitations of the tools. We propose to meet the technical challenge for German with a robust Transformer-based SAG model. We address the usability challenge with an easy-to-use graphical user interface for the SAG model, and the trust challenge with a workflow that allows teachers to evaluate the model on their own data, to choose on the basis of this evaluation which model predictions to trust, and in consequence to stay in control of grading their students while saving grading effort.

Keywords Short-answer grading · Grading support · German · Transformers

Introduction

Natural Language Processing for Education is an active research area aiming to support learning and teaching. It promises to do so very tangibly for Short-Answer Grading (SAG), where the task is to grade the content of one- to three-sentence constructed student answers (cf. the overview in Burrows et al., 2015). Hours of human effort could be saved on any single test by easy-to-use, reliable machine support that teachers and students can trust.

Yunus Eryilmaz and Larissa Kirschner both contributed equally to this work.

✉ Ulrike Padó
ulrike.pado@hft-stuttgart.de

[1] Hochschule für Technik, Schellingstr. 24, 70174 Stuttgart, Germany

Published online: 07 December 2023

Since research prototypes show promising performance on standard benchmark data sets, the research community is increasingly investigating the use of automated SAG models in the classroom (e.g., Zhu et al., 2020; Nazaretsky et al., 2022). This requires applying the research models to teachers' individual data sets and supporting teachers in tool adoption. The next challenge is preparing tools for independent use by teachers outside a study context.

In order to facilitate broad adoption of grading support in classrooms around the world, several things are needed: First, from a purely *technical* point of view, SAG tools have to be sufficiently reliable to be used in a wide variety of classrooms and on texts from varying domains, as well as texts written in many languages.

Currently, SAG research focuses on English, which is the language with most available training resources: English Wikipedia, which is often used for model development in Natural Language Processing, contains almost three times more words than French or German Wikipedia (Meta, 2023), and almost 50% of HTML documents archived by the Common Crawl project are in English (Common Crawl Project , 2023). Not surprisingly, the picture is similar for task-specific annotated corpora as well as models derived from them. Therefore, lack of resources is a continuing challenge for non-English SAG models.

Even for English texts, however, applying a SAG model to new data sets is challenging given that machine learning models perform best on data sets that are similar to the data they were trained on - therefore performance is expected to degrade for test data from different topic domains, or written by students of different ages or language backgrounds. For broad application, teachers therefore cannot rely on benchmarking results to estimate model reliability, but need to know the expected performance of the SAG model on their specific data.

The second challenge for accessible grading support is *usability* for a broad user base: The SAG tools have to be easy to use for all teachers regardless of their programming skills or even general computer expertise. This is necessary for seamless integration into teachers' workflows (Yuen and Ma, 2008; Burstein et al., 2012).

Third, the existence of a reliable and easy-to-use tool does not automatically mean that the intended users *trust* it. General distrust in or exaggerated expectations of AI tools can be overcome by (1) facilitating users' understanding of the capabilities and limitations of the tool in order to build teachers' trust, and (2) preserving their control over the grading process and the final grades communicated to students (Nazaretsky et al., 2022).

We will address these challenges for the case of German as an example. We solve the technical challenge by training and evaluating an automated SAG model. By harnessing a Transformer-based architecture, we reduce the need for task-specific training data to counter lack of data, while demonstrating that the model gracefully deals with properties of German that differ from English (section "Addressing the Technical Challenge: A German SAG Model"). In section "The Trust Challenge: The Automated Grading Work", we will explicitly demonstrate how much model performance deteriorates when transferring a trained model to other German data sets, and we will investigate whether the resulting less reliable grade predictions can still be useful to support human graders.

We counter the usability challenge with a straightforward user interface for the SAG model which is being specifically designed to be intuitively usable even for occasional computer users (section "The Usability Challenge: ASYST, an easy-to-use Automated Grading System").

We propose to address the trust challenge by a grading process[1] that facilitates a hybrid approach: Automated grade predictions are seen as suggestions subject to human review. This way, teachers retain agency in grading, but can save effort by skipping the review step whenever the automated predictions are reliable enough. From this perspective, two questions remain: First, how much disagreement between automated and human grades is any individual teacher ready to accept and still trust the system, and second, how to tell which grade predictions are in fact trustworthy and which require human attention. The first question is up to the users. The second question can be answered by a detailed analysis of the SAG model on data that is comparable to that of the intended use case (cf. Mieskes and Padó, 2018). We demonstrate results of the workflow for four German corpora and find that by focusing human effort where it is most needed, even imperfect SAG models can still be used to save grading effort and even greater overall grading consistency.

Plan of the Paper

We begin by reviewing literature relevant to the three challenges ("Related Work") section. We then address the technical challenge in "Addressing the Technical Challenge: A German SAG Model" section. After further specifying the task by describing some properties of German that differ from those of English, we document the training and evaluation of a SAG model for German, as well as analyzing its appropriateness for German.

"The Usability Challenge: ASYST, an easy-to-use Automated Grading Syst" section addresses the usability challenge and describes the current state of the ASYST tool, our user-friendly front-end for the German Transformer model.

We discuss the workflow that we propose for countering the trust challenge in detail in "The Trust Challenge: The Automated Grading Workflow" section and demonstrate its results on four SAG data sets for German. Finally, "Conclusions" section concludes.

Related Work

We begin by briefly reviewing the literature relevant to the three challenges identified above.

The Technical Challenge: Automated Short-Answer Grading

Researchers have traditionally met the technical challenge of building well-performing SAG models with feature-based, non-neural algorithms (Burrows et al., 2015). Since these algorithms cannot process text input as-is, features are derived to represent

[1] Thus expanding on our earlier work in Padó (2022).

relevant properties of the text, such as important words. A common SAG strategy is to create these feature sets for a student answer and a correct reference answer, and to train the model to recognize correct answers by their feature overlap with the reference answer.

Feature-based approaches allow fine-grained control over which aspects of the student answer are considered relevant to grading (Ding et al., 2020). Since the simple bag-of-words baseline, which uses just the words in an utterance as features, is strong for SAG (Dzikovska et al., 2013), some approaches have successfully relied on string-level representations of student and reference answers and their similarity (e.g., Jimenez et al., 2013; Ramachandran et al., 2015). On the other end of the spectrum, language-specific features have been constructed which require complex pre-processing and represent, e.g., syntactic and semantic relationships between input words or the whole input (Ott et al., 2013; Zesch et al., 2013).

Recently, features based on embedding representations have proven very successful (e.g., Sultan et al., 2016; Saha et al., 2018; Kumar et al., 2019; Steimel and Riordan, 2020; Vittorini et al., 2021). Embeddings are representations for words (or whole sentences) that were derived from the words' distribution in context, often by a deep neural network. To some extent, embeddings capture semantic properties of the modelled words without requiring deep syntactic and semantic pre-processing of the input. Embeddings can then be used as input for non-neural machine learning algorithms.

Alternatively, neural models can be used to both learn word embeddings and, in the same model, use these representations to solve the overarching task, in this case grade prediction for SAG. Riordan et al. (2017) analyze different fully neural approaches to SAG. Bai and Stede (2022) discuss embedding-based and fully neural approaches to SAG and essay grading.

When embeddings or fully neural models are used, intensive pre-processing for feature induction (like deep syntactic and semantic analysis) is no longer required, and yet models achieve much better results than feature-based non-neural approaches. However, this strength comes at a cost: Neural models are data hungry and, if trained from scratch, need much more training input than standardly available for SAG (where corpora contain in the thousands of answers).

In this situation, transfer learning for Transformer-based deep neural models like BERT (Devlin et al., 2019) is a promising solution. Transformers (Vaswani et al., 2017) are a deep neural network architecture that encodes the input into an abstract internal representation and is able to decode it again into the desired output. Applied to text data, Transformers learn a detailed language model from large amounts of input text during the pre-training stage. At this point, no task-specific, annotated training data is needed, as the models typically learn to predict words that were masked from the input text (Devlin et al., 2019). After pre-training, Transformer-based models are fine-tuned on small amounts of task-specific data, with good success (Howard and Ruder, 2018). This means that the models' encoding portion is topped with one or more new classification layers, after which the whole architecture (or just the new layers) are trained on the task-specific annotated data. A successful example of this strategy for SAG is presented in Ghavidel et al. (2020). In addition, it is possible to further improve results by pre-training for several rounds, first on un-annotated training data, then on

annotated data for a related task[2], and finally on the task-specific data (Sung et al., 2019; Camus and Filighera, 2020; Willms and Padó, 2022).

BERT can be trained directly to solve the SAG task, but Reimers and Gurevych (2019) introduced the SBERT architecture specifically for tasks that rely on sentence comparison. The architecture uses BERT to create sentence-level embeddings for two input sentences in parallel. These are concatenated and used as input for a final classification layer. The architecture is designed for any text classification task that requires the comparison of two portions of text on the semantic level. For SAG, the student and reference answers are passed into the SBERT model and the classification layer is trained to predict the grade based on the representations of the two inputs. Condor et al. (2021) use SBERT for SAG, but in their best-performing setting provide the model with the student answer and question text, since no reference answers are provided in their data (scoring rubrics exist, but are not helpful in their experiments). Bexte et al. (2022) propose an alternative: In a classic memory-based learning approach, they compare the SBERT representations of the input to those of reference answers (or training examples of student answers) and assign the grade of the closest answer (or set of answers). Beyond training a classification layer and using a memory-based strategy, Bexte et al. (2023) also experiment with ensembles of Transformer-based models.

Beyond the SAG task proper, researchers have also been interested in identifying content dimensions (Mizumoto et al., 2019; Gombert et al., 2023). Another highly active area of current research into Transformers for SAG is the analysis of model behavior from the point of view of robustness towards unorthodox input (e.g., Ding et al., 2020; Willms and Padó, 2022; Filighera et al., 2023), explainability of model decisions (e.g., Poulton and Eliëns, 2021; Törnqvist et al., 2023) as well as giving feedback (Filighera et al., 2022).

Research into automated SAG routinely finds that the same models perform differently on different data sets. Zesch et al. (2023) put this performance difference down to variance in the student answers, where questions with high variance in the answer texts are harder to score automatically. They suggest that would-be users of SAG models first determine the expected amount of variance in their data (stemming from conceptual, linguistic or nonconformity variance in student answers), and to expect best results from automated SAG for low-variance data.

Several recent publications describe the use of automated grading systems in real-world examinations. They share the assumption that all automated SAG models will be somewhat flawed, and discuss several strategies to address these flaws.

Condor (2020) suggests using a BERT-based grading model alongside a human grader to save human effort in a standard *double-grading* situation where two graders work independently of one another and any answers with disagreeing grades are reviewed by a third grader. This is similar to existing real-world applications of machine grading as a replacement for one human grader such as the ETS e-rater[3] (Attali and Burstein, 2006).

[2] For example, the MNLI corpus for Natural Language Inference, Wang et al. (2018).

[3] see https://www.ets.org/erater/about.html

Vittorini et al. (2021) use a *human-review* setting in high-stakes situations where all automated grades are standardly reviewed (and possibly revised) by a human grader. Since reviewing a pre-assigned grade can be easier than independent grading, they report a time-saving effect of 43% compared to independent grading. For lower-stakes formative settings, Vittorini et al. deem their system's grading accuracy of 89% sufficient.

Mieskes and Padó (2018) suggest involving human graders only for those automated grades that are *least likely to be correct* in order to save human time and effort. Candidate grades for revision can be identified using the results of a priori automated grader evaluation (which yields information about general weaknesses of the grader for specific grade labels) or the agreement in an ensemble of automated graders (which indicates ensemble confidence in individual assigned grades). Schneider et al. (2022) follow a similar strategy. They identify questions that are hard to grade based on the automated grader's similarity estimate for each student and correct reference answer. In both approaches, the teacher can adjust the expected correctness of the automated grader predictions and the human workload by adjusting the parameter of which questions should be assumed to be hard to grade.

Azad et al. (2020) use an automated grader with an accuracy of 89% for *stand-alone grading* even in a high-stakes setting. Their focus is on adapting the grading process in order to mitigate the effect of automated grader errors: They allow several answer attempts in case the automated grader rejects a student's answer, as well as offering an appeal to manual grading. They report student satisfaction with the process and find that less than 1% of students are incorrectly denied points by the system. However, they also note that the multiple-attempt strategy allows students to try out conceptually different answers, which leads to an increase of incorrectly awarded points. Interestingly, the option to appeal to human re-grading mostly seems to affect student satisfaction with the process; the offer does not noticeably change the outcome of the exam.

In sum, there are different strategies to address flaws in automated grading in order to benefit from reduced human grading effort: Human involvement can be limited by taking a reviewing role rather than grading from scratch or by focusing grading effort where it is most needed, but it is also feasible to make changes to the testing situation to ensure points are not unfairly withheld.

The Usability Challenge: Software Usability

Usability is defined as the "extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" (ISO Technical Committees, 2018). Effectiveness means that the user's intended task can be completed successfully, while an efficient system allows the user to reach this goal quickly. User satisfaction with the process is measured through user feedback. Systems with high usability are easy to understand for users from many backgrounds, which also facilitates learning how to use the system and avoiding errors during use (ISO Technical Committees, 2018).

In Software Engineering, usability is in focus in User Interaction (UI) and User Experience (UX) design. Shneiderman (1987) formulates eight "golden rules" for user interface design (cited here from Shneiderman et al., 2016):

1. Strive for consistency
2. Seek universal usability
3. Offer informative feedback
4. Design dialogs to yield closure
5. Prevent errors
6. Permit easy reversal of actions
7. Keep users in control
8. Reduce short-term memory load

Rules 2, 4 and 5 resonate directly with the extended ISO specification where it requires usability for users from many backgrounds, efficient and satisfying handling of user tasks and the resulting reduction of errors. Rule 8 points to the fact that all UI design principles used to implement these guidelines are rooted in cognitive psychology: The software tools should be designed for the human mind (Johnson, 2013).

This means that in implementing general rules like 2 or 5, software design can profit from properties of human cognition: By conforming to common design practices and navigation strategies shared with other software GUIs, UI designers rely on users' previous experience and expectations to navigate a new application without instruction. On the other hand, specific properties of human visual perception prompt UI designers to place related items closely together, highlight them in similar colors and use icons rather than text wherever possible to make their GUIs comprehensible "at a glance" or highlight important information (Johnson, 2013).

In addition to these design principles, the recommended development process includes careful study of the expectations, abilities and goals of the intended users, which helps define detailed requirements. GUI development is iterative, starting with mockups and progressing to implemented prototypes while involving the users early on to ensure timely feedback at multiple stages during development (Galitz, 2007).

The Trust Challenge: Adopting Automated Tools in the Classroom

A number of recent studies look at the impact of SAG and writing feedback systems in the classroom, on both teachers and students, and factors that influence tool adoption. They rely on concepts from the Technology Acceptance Model (Davis, 1989), which posits that two factors influence actual system use: Perceived usefulness and perceived ease of use of the system. On top of these, additional factors like the social norm (i.e., the socially implied desirability or undesirability of using the system) can be defined (Venkatesh and Davis, 2000).

Yuen and Ma (2008) look at what convinces teachers to use eLearning software for their classes and find that teachers' intention to use the software depends (directly or indirectly) on their perception of their own computer abilities, the ease of use of the software and the perceived social norm. From a software development perspective, this shows the importance of a hassle-free user interface, especially for teachers who feel that their computer abilities are limited.

Another relevant finding is the importance of trust in the system by both students and teachers: Zhai and Ma (2022) present a meta-study on the perceived usefulness of automated writing evaluation by Chinese students. They identify students' trust in the automated system as an important factor explaining the perceived usefulness of the system. Other influential variables were the perceived social norm, and feedback quality, in that students valued feedback that supports reflection on the writing process.

Addressing the question of teachers' trust in an automated SAG tool specifically, Nazaretsky et al. (2022) recommend providing teachers with a basic understanding of the technology as well as focusing on concrete applications that teachers deem useful. They also recommend to give teachers unrestricted agency in integrating the tools into their teaching, as well as giving them the ability to override grading proposals.

Addressing the Technical Challenge: A German SAG Model

We now turn to the practical part of the paper, addressing the three challenges in turn. Our solution will use German data as a specific problem instantiation, but is not limited to German. We begin with the technical challenge, namely the need for a robust SAG model in German. In order to specify the problem setting, we first describe some syntactic and morphological properties of German that differ from those of English. We then describe the training and evaluation of an automated SAG model for several German corpora. Since some corpora are annotated with partial credit and others just as *correct-incorrect*, we normalize all grades to the *correct-incorrect* case to make evaluation results more comparable (see also "Corpora"). We also specifically verify the chosen model's appropriateness for dealing with German.

Properties of German

German is in principle a well-represented language in terms of publicly available written text: It has the third-largest resources in both Wikipedia (Meta, 2023) and the Common Crawl (Common Crawl Project , 2023), a large web data collection project. However, there are only four SAG corpora for German[4] with a grand total of roughly 6,500 student answers, as opposed to seven freely available standard corpora for English[5] with a grand total of roughly 44,500 human-graded student answers. Therefore, the need for methods that perform well in data-poor environments is still pressing when working with German.

German has some properties that make the use of the traditional simplifying assumptions of NLP harder than for English (Krumm et al., 2011): On the morphological level, the presence of verb, adjective and noun inflection calls for normalization strategies like lemmatization (or splitting words into sub-tokens in the case of embeddings).

[4] The German SAG corpora are: CREG (Meurers et al., 2011), CSSAG (Padó and Kiefer, 2015), ASAP-DE (Horbach et al., 2018) and SAF-DE (Filighera et al., 2022)

[5] The English SAG corpora are Mohler et al. (Mohler et al., 2011), CREE (Meurers et al., 2011), ASAP (www.kaggle.com/c/asap-sas), SciEntsBank and Beetle (Dzikovska et al., 2013), Powergrading (Basu et al., 2013) and SAF-EN (Filighera et al., 2022)

German nouns and adjectives are morphologically marked for four cases and take one of three grammatical genders (see the top of Table 1 for examples). In addition, there is a variety of plural paradigms for nouns, including Ablaut (vowel changes inside the word stem as for some irregular verbs in English). Verbs are marked for person and number in all tenses and can show Ablaut formation in the word stem as well, across tenses. This means stemming strategies are less likely to succeed in German than in English for all word classes.

Highly productive noun compounding adds complexity in lexical semantic analysis for the human and machine learner alike. Compound nouns express meaning that would often be realized as a noun plus noun or a noun-of-noun construction in English (see examples in line 5 of Table 1).

On the syntactic level, word order is more flexible than in English: In German, only the verb has a fixed position in the sentence (either in the second or in the last place), while other phrases are much freer to move (cf. the final part of Table 1) – the order variants differ with regard to information structure (see, e.g. Wöllstein, 2014). This means surface-level strategies for inferring syntactic relationships are less reliable than in English.

Table 1 Some examples of German morphology and syntax

Phenomenon	German	English translation
Noun case	der Tisch, des Tisches	the table
	dem Tisch(-e), den Tisch	
Noun plural	der Tisch, die Tische	the table, the tables
	das Dach, die Dächer	the roof, the roofs
	die Sache, die Sachen	the thing, the things
Verb person	ich gehe, du gehst, er geht	I go, you go, he goes
Verb tense	ich ging, du gingst, er ging	I went, you went, he went
Noun compounding	Schreibtisch	desk (lit. writing table)
	Hausdach	roof of the house
Word order	Gestern biss der Mann den Hund.	Yesterday, the man bit the dog.
	Der Mann biss gestern den Hund.	The man bit the dog yesterday.
	Der Mann biss den Hund gestern.	
	Den Hund biss der Mann gestern.	
	Den Hund biss gestern der Mann.	
	..., dass gestern der Mann den Hund biss.	... that the man bit the dog yesterday.
	..., dass der Mann gestern den Hund biss.	
	..., dass der Mann den Hund gestern biss.	
	..., dass den Hund gestern der Mann biss.	

Taken together, these morphological and syntactic variants for German words and sentences mean that similar meaning can be expressed in many different surface strings in German. Therefore, representing a sentence by patterns in word distributions (and sub-word sequence distributions) is harder for German than for English. This in turn makes it harder to automatically extract the content of a student answer in SAG for modelling approaches that rely on distributional modelling of semantics.

Method

To create a SAG model for German, we employ a successful strategy from the literature (Camus and Filighera, 2020; Willms and Padó, 2022) and use a fine-tuned Transformer approach. Transformer-based models require very little annotated training data for fine-tuning, which helps in the face of data sparseness. Also, Transformer-based models do not require input pre-processing for feature generation, which is a time-consuming and somewhat brittle step in feature-based grading models. Note that our goal is not to create the best possible German SAG model: Instead, we want to demonstrate the usefulness of a robust, relatively off-the-shelf model, knowing that it could be replaced by an even better-performing model in the future.

Model

Specifically, we use the SBERT architecture (Reimers and Gurevych, 2019), which is designed for tasks that require the comparison of whole sentences with regard to their meaning (such as paraphrase detection or inference). We apply it to SAG by generating representations for the student answer and the reference answer for the question provided in the corpus.

SBERT has been used for English SAG before by Condor et al. (2021), but their best-performing setup compares the student answer to the question text, while we compare to a correct reference answer (which is not available to Condor et al.) Our approach appears more natural in terms of the SBERT architecture, since it was designed to enable exactly the direct sentence comparison on the semantic level that is required in our setting.

SBERT derives fixed-length representations u and v of the input sentences from a BERT language model by first splitting words into (subword-)-tokens and then deriving embedding representations for these tokens. These embeddings are concatenated and fed into a final classification layer that learns to predict grades from the combined representations. The SBERT model was pre-trained for a paraphrase detection task, which also involves comparing sentences according to the similarity of their content.

We parametrize the architecture for use on German as follows: We employ the `sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2` BERT-based tokenizer and language model (Reimers and Gurevych, 2020, available on https://huggingface.co). This model covers 50 world languages, among them German,

but (originally) not English[6]. We only evaluate the model for German, but hope to eventually make the Automated Grading Workflow available to as many users as possible.

We use mean pooling for sentence embedding creation, combine the resulting embeddings as $|u - v| + |u \circ v|$ and use a logistic regression classifier for the classification layer. These parameters were chosen by cross-validation on the Computer Science Short Answers for German (CSSAG) corpus training set. Note that only the classification layer is ever trained in our experiments; the language model for embedding creation remains untouched. We will refer to this model as the German Transformer model below.

The steps of the processing pipeline thus become:

1. Preparation of the input strings using the publicly available tokenizer
2. Generation of a fixed-length representation of the input using the publicly available language model
3. Combination of the representations for the student and reference answer into one input vector
4. Classification of this input vector by a logistic regression learner inferred from the training data

Evaluation

We report the standard evaluation scores of weighted Precision, Recall, and F-Score. Precision shows which percentage of the predictions of a specific label were correct, that is, how trustworthy the grader's predictions are. Recall shows how many instances of a specific class were labelled correctly by the grading model, i.e., how many instances of interest were identified. These measures are computed on the basis of single labels and then the weighted average across labels is computed to characterize the performance of the grading model as a whole. The F-Score is the harmonic mean of Precision and Recall and serves as a single measure for model comparison.

For comparison with the literature, we also show Accuracy, the number of instances that were labelled correctly across the whole test set.

To make predictions, we leave one question out in turn and train on the others, making predictions for the one question that was unseen in training. This evaluation setting is called "unseen question" (Dzikovska et al., 2013). It is the hardest formulation of the SAG task, since the models have no information about question-specific answer patterns for the test question and instead have to learn general signals of semantic similarity between student and reference answers. It is also the only realistic setting of the task – except for the rare case of large-scale, standardized testing, where previous student answers are available for question-specific training (Padó, 2016).

Corpora

We will work with the four German SAG corpora that are currently available. They were collected in different contexts and with different motivations: CREG contains

[6] Reimers and Gurevych (2020) give the languages as ar, bg, ca, cs, da, de, el, es, et, fa, fi, fr, fr-ca, gl, gu, he, hi, hr, hu, hy, id, it, ja, ka, ko, ku, lt, lv, mk, mn, mr, ms, my, nb, nl, pl, pt, pt-br, ro, ru, sk, sl, sq, sr, sv, th, tr, uk, ur, vi, zh-cn, zh-tw

data from learners of German as a second language, while the other corpora focus on content assessment. ASAP-DE was created through crowdsourcing, the other two corpora were collected in the course of teaching. Table 2 lists the corpus sizes - ASAP-DE is the smallest at 903 answers, and the German section of SAF is the largest at 2407 answers.

Table 2 also shows the label distribution for the binary correct-incorrect decision. Note that only CREG is annotated with binary grades. The other corpora have question-specific grading scales, which makes automated labelling and comparison of results more difficult. Therefore, we normalize all grade annotations to the binary case, where any answer with $\geq$ 50% of the maximum available points counts as correct. After normalization, CREG and CSSAG are (approximately) balanced, ASAP-DE has more incorrect than correct answers and SAF-DE is strongly biased towards correct answers.

SAG for German: Results

Table 3 shows the results of training and evaluating the German Transformer model separately on each of the four German corpora using leave-one-question-out evaluation to test on unseen questions. For comparison, we also give the state-of-the-art result from the literature for each corpus in Table 3. Since we make predictions for the binary *correct-incorrect* case, no literature results for ASAP-DE and SAF-DE exist (published figures are for the multi-label, partial credit case).

The literature benchmark models for CSSAG and CREG are systems relying on carefully designed features derived from several different layers of linguistic analysis of the input, for example syntactic structure and deep semantic representations.

It is quite clear from Table 3 that the German Transformer model does not beat the literature results. It does, however, approximate the performance of the carefully hand-crafted model for CSSAG. Performance on SAF-DE, where direct comparison to the literature is missing, is comparable to CREG, the best result. For ASAP-DE, results are noticeably poorest. The reason is probably that the ASAP-DE data covers only three questions, so that the model has difficulty in generalizing sufficiently from the seen two questions in the training set to the unseen question currently tested.

These results highlight two important points: First, Transformer-based models, data efficient as they are, do not necessarily outperform other types of grading model if only small amounts of training data are available. This underscores the undiminished value of crafting language-specific models that rely on linguistic knowledge in data-poor settings, if possible.

Table 2 German SAG corpora: Source, size and label distribution for binary correct-incorrect decisions

Source	# answers	% correct-incorrect
CREG (Meurers et al., 2011)	1032	50 – 50
CSSAG (Padó and Kiefer, 2015)	1926	56 – 44
ASAP-DE (Horbach et al., 2018)	903	25 – 75
SAF-DE (Filighera et al., 2022)	2407	80 – 20

Table 3 Evaluation results: Our German Transformer model and literature results on German corpora. Unseen question evaluation. Weighted F-Scores and Accuracy

Data Set	Model	F-Score	Accuracy
CREG	German Transformer	72.1	72.1
CREG	Hahn and Meurers (2012)	–	86.3
CSSAG	German Transformer	65.0	65.1
CSSAG	Padó (2016)	66.6	69.3
ASAP-DE	German Transformer	59.8	64.7
ASAP-DE	–	–	–
SAF-DE	German Transformer	71.7	70.6
SAF-DE	–	–	–

Second, we see clearly that for the same machine learning model architecture and underlying language model, prediction performance can vary quite widely across corpora. This point is especially relevant for the real-world applicability of automated grading: The performance on any new data set cannot be reliably predicted from existing benchmarks. Note that in these experiments, models were tested on unseen data originally from the same source as the training data. This is an optimal setting for machine learning, and model deterioration is expected when test data from a different source is used. We quantify the loss in model performance when switching to different test sets when we demonstrate our Automated Grading Workflow in "The Trust Challenge: The Automated Grading Workflow" section.

Model Appropriateness for German

Some of the performance difference of our German Transformer model to the literature models that rely on deep linguistic analysis could be caused by difficulty dealing with the syntactic and morphological complexities of German (cf. "Properties of German)" section. We therefore investigate how well the tokenization step in our German Transformer model deals with German morphology, since this is the crucial step that prepares the student answers for further processing by the language models. Given the extremely wide coverage of languages from different language families and writing systems of the pre-trained Transformer model we use, it is not self-evident that the multilingual language model appropriately covers German.

Tokenization in our chosen language model relies on the WordPiece approach (Wu et al., 2016) which allows words to be broken into substrings and derives a set of tokens from its training data such that it concisely covers the vocabulary: A representation of common word stems plus endings is more concise in this sense than a list of all encountered strings. Any words in later input that were unseen in model training can be represented by (subword) tokens from this vocabulary. Wu et al. (2016) state that between 8000 and 32000 word pieces yield good results – this is much less than the vocabulary size of corresponding corpora.

In our case, the multilingual language model was trained on data in 50 languages, which all had to be represented by the same inventory of word pieces. This inventory is also trained from data and without explicitly considering linguistic rules. Despite the complexity of this task, we do find word representations that are reminiscent of German lemmatization. Table 4 shows examples. Tokenizations that match linguistic analysis are set in bold face. Underscores in front of letter sequences mark the beginning of a new word in the original text.

Lines 1 and 2 show sample text written by native speakers on the topic of Java programming. We find that tokens mostly correspond to morphemes, although frequent words such as the determiners *diese* and *eine* are tokenized as complete words instead of inflected forms, probably due to their frequency. Content words, except for *Methode (method)* and *Konstante (constant)* are analyzed correctly – *prüft* is split into the stem and the person affix, and *Inhaltsgleichheit* is separated into its components *Inhalt (content)* and *gleich (equal)* as well as the linking element *s* and the derivation marker *heit*. *Schnittstelle (interface)* is represented by its components *Schnitt (here: nominalization of 'to intersect')* and *Stelle (place)*, while the latter keeps its plural marker *n* (possibly due to overlap with the verb *stellen (to put, to place)*).

The prefix *ver* of the verb *vererben (to pass on by inheritance)* is represented as one token, but the stem *erben (to inherit)* shows non-standard segmentation. This is probably due to the cross-lingually high frequency of the sequence *er*, which prompts misrepresentation (the canonical lemmatization would be stem *erb* and infinite ending *en*).

Learner language (line 3 in Table 4) is more of a challenge for the tokenizer: Missing capitalization causes the word *Kenntnis (knowledge)* to be (only somewhat correctly) represented as *kennt (knows)* plus derivation morpheme *nis* (a morphological analysis would yield *kenn–t–nis*, with the stem *kenn* followed by linking element *t* and *nis*). Capitalized versions of this word are represented as a single token in the data. This representation difference between capitalized and uncapitalized versions of the same word may cause the grading model to miss lexical overlap. The spelling error in *österreichische (Austrian)* also causes a complete, but linguistically unfounded representation. These observations may explain why the German Transformer model so clearly stays below the literature results for the learner data in CREG.

Besides morphology, which the Transformer-based model captures in broad strokes, we mentioned the relatively free word order in German as a potential issue for the

Table 4 Automatic tokenization of German text, samples from CSSAG (native speakers, lines 1 and 2) and CREG (learners, line 3)

'**_Da**', '**_dies**e', '_Method', 'e', '**_auf**', '**_Inhalt**', '**s**', '**gleich**', '**heit**', '**_prüf**', '**t**', '.',
Because this method checks for content equality.
'**_Man**', '**_nimm**t', '**_ein**e', '**_abstrakt**', '**e**', '**_Klasse**', ',', '**_da**',
'**_Schnitt**', '**stell**en', '**_nur**', '_Konst', 'anten', '**_ver**', 'er', 'ben', '**_könn**en', '.'
One uses an abstract class, since interfaces can only pass on constants [by inheritance].
'**_kennt**', '**nis**', '**_in**', '**_der**', '_öst', 'er', 'rich', 'en', '**_Küche**'
knowledge (sp) of the Austrian (sp) cuisine

automated grader. As our examples in Table 1 show, adverbs and arguments can take several positions in the sentence, while the verb position is either in the beginning (in second position) or at the very end of the sentence. The challenge for any model is therefore to identify and represent verbs and arguments wherever they appear in the sentence. This task is made easier by the bidirectional training strategy employed by the BERT family of language models: The input is passed to the models simultaneously front-to-back and back-to-front, ensuring that, e.g., the German verb is encountered soon no matter what its position in the sentence.

Furthermore, BERT is quite insensitive to input shuffling on Natural Language Understanding tasks (Hessel and Schofield, 2021; Willms and Padó, 2022), which also implies that it should be robust to effects of (relatively) free word order.

Therefore, we conclude that even though the language model does not deeply analyze the input text on a linguistic level, it is well-equipped to deal with the complexities of (standard) German that we introduced.

Cross-Language Comparison to English

Cross-language model comparison is difficult, since evaluation results vary strongly by corpus even within one language, as we have seen. Table 5 does demonstrate, though, that evaluation results for our German Transformer model in terms of F-Score are numerically similar to state-of-the-art Transformer results for the SemEval-2013 English standard corpora (Dzikovska et al., 2013). (No benchmarks exist for the other English corpora on the *correct-incorrect* task.)

To adapt to standard procedure for the English corpora, we report evaluation results for a fixed train-test split of the German data sets that is stratified by question. This setting is called "unseen answers" and is an easier task than the "unseen questions" setting reported above, since the models have already encountered sample answers for each question during training. This explains the higher F-Scores for the German Transformer model on all four corpora compared to the unseen questions results in Table 3. This is especially noticeable for ASAP-DE, where switching to unseen answers evaluation removes the difficulty of generalizing over a small number of available questions in the training set. For each corpus, a set of approximately 200 answers is withheld for

Table 5 Evaluation results: German Transformer model on German corpora and literature results for English standard corpora (Beetle/SciEntsBank:Poulton and Eliëns (2021), model albert-large). Unseen answer evaluation. Weighted F-Scores

Data Set	Language	F-Score
CREG	German	81.7
CSSAG	German	84.8
ASAP-DE	German	86.2
SAF-DE	German	86.2
SciEntsBank	English	83.4
Beetle	English	91.2

testing (this is 20% of CREG and ASAP-DE, 10% of CSSAG, and the 270 instances of the unseen answers test set of SAF-DE).

The results do not directly prove the quality of the German Transformer model, but they do indicate that the model's performance is numerically comparable to reference models for English.

Discussion

In sum, we have seen that our Transformer-based SAG models for German data show comparable performance to English-language models despite the smaller amounts of available training data because of their smaller data requirements for fine-tuning existing language models. These models also appropriately deal with German morphology and syntax. The models however do not outperform hand-crafted, feature-based models using linguistic analysis on the harder "unseen questions" task that is closest to expected usage of the models in real-world classrooms. This highlights the power of utilizing linguistic knowledge in hand-crafted systems in data-poor settings. We have also seen variance in performance across different corpora, an argument for closely inspecting model performance on each new data set, especially in a practical setting.

The German SAG model we have built and evaluated is not fully optimized. There is sure to be a further margin for improvement using different pre-trained Transformer models, different classification models or even different classification approaches (e.g., as in Bexte et al., 2023). We have shown, however, that the model has robust performance, and we will demonstrate how teachers can profit from its predictions in a hybrid grading setting in "The Trust Challenge: The Automated Grading Workflow".

The Usability Challenge: ASYST, an easy-to-use Automated Grading System

We counter the challenge of making the German Transformer model easy to use for a broad user base by developing the ASYST (Automated Grading System) tool. In addition to the German Transformer model introduced in "SAG for German: Results" section, ASYST also wraps a SAG model for English, so that the input language can be freely chosen for each data set.

ASYST takes as input an Excel sheet containing student answers and correct reference answers. Question texts can also be provided at this step so they will be available for reference during later review. They are however not used for grade prediction. If available, teachers can also specify existing manual grades for some or all student answers – this helps with analyzing the reliability of ASYST predictions on one specific data set.

The Excel sheet containing student answers and correct reference answers is loaded into the tool in just two steps (see Fig. 1): First (top left), the user is asked to choose the language of the student answers, and then to specify the Excel sheet using a standard file chooser dialog that pops up upon clicking the "Input file" button. In the top right of Fig. 1, a file named `DE-Demo-Daten.xlsx` has been loaded successfully.

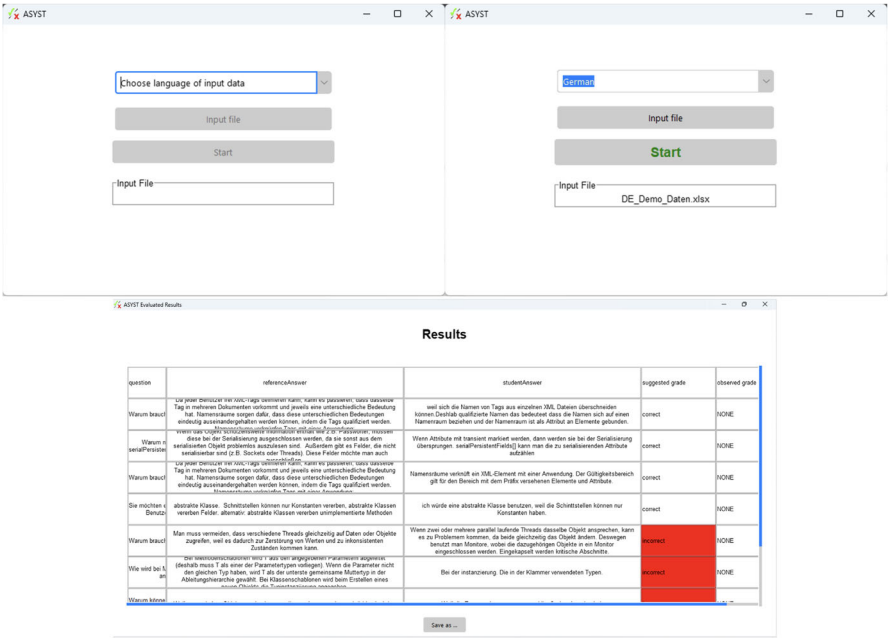

Fig. 1 The ASYST GUI. Top left: Initial state. Choose language and input file. Top right: Ready to start. Bottom: Result display and Save button

As the user progresses through these steps, the next relevant button (first "Input file", then "Start") is only activated in the GUI once the previous task is concluded in order to prevent errors (following Golden Rule 5) and feedback about user choices is given (according to Golden Rule 3). In the top right of Fig. 1, this is done by showing the name of the loaded input file for verification. The user can always go back to earlier steps if needed – this keeps the user in control as demanded by Golden Rule 5.

Finally, at the click of the *Start* button, the grading process begins, using the appropriate grading model for the chosen input language. Automated grading is fully local, which precludes data protection issues because the student answers never leave the teacher's computer.

After grading, a table opens with the results (see bottom of Fig. 1). The table shows all the columns provided in the input file, but centers on the reference and student answer and the ASYST grade prediction. The "observed grade" column at the far right is for existing manual grades, but the value `NONE` was entered instead by the user in the screenshot. Whenever the *incorrect* grade is encountered in the "suggested grade" or "observed grade" columns, the corresponding spreadsheet cell is highlighted in red.

The table content can then be saved as an Excel sheet for review and revision at the teacher's discretion before being distributed to students. If previously observed grades exist, model reliability can also now be evaluated (cf. "Analysis") by comparing the labels in the suggested and observed columns using Excel formulae.

Design Process and User Involvement

ASYST was designed with Shneiderman's Golden Rules in mind (see "The Usability Challenge: Software Usability") in order to achieve high usability as defined by the ISO. To this end, users were involved in two places in the design process: First, a small group of test users helped guide initial development. Second, a group of teachers experienced in short-answer grading participated in a more formal usability study.

Initial Development

Three test users of different ages and computer skills gave feedback with a focus on ease of use for anyone irrespective of their technical skills. One tester was a software developer, one had experience using office applications for work and one mostly used mobile phone apps for all digital tasks. These early testers were not experts on short-answer grading; the age range was 20-55 years. They used an early version of ASYST to load student answers in different file formats, generate grade predictions and save them.

The most important result of this test user involvement was the focus on Excel sheets as input and output format because this format was most familiar to the testers and is (in the LibreOffice incarnation) accessible to anyone. Another advantage of the format is that Learning Management Systems (LMS) like Moodle[7] or Ilias[8] that are commonly used in German universities offer Excel export of student answers from online tests, which allows integration with existing computerized test tools.

Usability for Target Group

Upon completion of a first full version of ASYST, we ran a second round of user tests with a dual goal: One, to more formally evaluate software usability, and two, to gather additional feedback from users from the intended target group.

We recruited a panel of teachers with experience in short-answer grading. In total, six people (three women and three men) participated. Four participants teach German as a second language, two teach computer science. All six have more than three years' experience grading short answer questions.

The testers were given an Excel sheet with sample student answers in individual test runs and were asked to create grade suggestions using ASYST and to save the results in a new Excel sheet. In order to evaluate usability in terms of the ISO definition (see "The Usability Challenge: Software Usability"), we registered whether users were able to complete the task (measuring *effectiveness*), how long they took to complete the task (measuring *efficiency*) and how satisfied they were with the ASYST tool, using the standard Software Usability Score (SUS, Brooke, 1995) to measure *satisfaction*.

Users were asked to think aloud during task completion, in order to help the observer understand the reason for any problems using the tool. Users were also asked for free feedback on the ASYST tool after the test. Feature requests and comments on difficulties were documented.

[7] www.moodle.org

[8] www.ilias.de

All participants were ready in principle to use a tool for grading assistance; the language teachers in particular were very interested in a tool to support their grading and help save them time.

Table 6 shows the numeric results of our small user study. All testers were able to complete the task successfully, leading to 100% effectiveness of the tool. The median time taken on the task (from setting the input language to completed saving of the results, and counting the run time of the tool) was 2:15 min. The minimum time was 1:24 min, the maximum was 6:14 min. However, in this longest trial, the participant carefully reviewed the grade suggestions before saving, which accounts for the additional time taken.

The average SUS score across the six participants was 88.8, with a minimum score of 67.5 and a maximum of 100 (which is a perfect score). SUS scores should be interpreted against the empirically observed score distribution. Bangor et al. (2009) in a meta study found the SUS mean score across all tested items to be 70, while the mean score for Graphical User Interfaces (GUIs) like ASYST is 76. This puts the SUS score of the ASYST system well above average, even for a GUI, and indicates good usability.

Even though our results should not be over-interpreted due to the small number of participants, they do indicate that short-answer grading experts were able to use the tool successfully and smoothly to generate grading predictions for further revision.

We also collected feedback and feature requests from the users. Three aspects were mentioned more than once: Users asked for clearer visual indication that grade prediction can be started once the input file is loaded. In the tested software version, this was indicated by the unlocking of the Start button. Users suggested to additionally highlight the button in a contrast color to the gray background. When inspecting the results, users also asked to see the question text in addition to the reference and student answers. These two requests have already been incorporated in the current version of ASYST (see the screenshots in Fig. 1).

While the ASYST results display is meant to give a quick overview before saving, users wanted to be able to immediately edit the output in the result display. This functionality is scheduled for future implementation.

ASYST Availability

The finished tool is available as an end-user application in the form of a compiled Windows 11 executable at https://transfer.hft-stuttgart.de/gitlab/ulrike.pado/ASYST. The Python source code is also available there under an Open Source license. It can be executed in a Python interpreter on all platforms.

Table 6 Evaluation of the usability of ASYST by six experts: Effectiveness (success on task), efficiency (median time taken in minutes) and satisfaction (average SUS score)

Effectiveness	Efficiency	Satisfaction
100%	2:15 min	88.8

Current goals for further development also include an evaluation mode to test model performance on one's own data (currently, performance evaluation has to happen manually using spreadsheet calculations in the output Excel sheet) as well as adding GUI support for more of the many languages that the SBERT model covers.

Other tools are available that are similar to ASYST, but have a somewhat different goal: One is the ESCRITO scoring tool (Zesch and Horbach, 2018), which takes users through multiple steps analyzing whether the use of SAG models on their data is promising. Another is ShinyReCoR (Andersen and Zehner, 2021), which supports human analysis and grading of data by creating semantic clusters of the input answers.

The Trust Challenge: The Automated Grading Workflow

In the literature, two keys to adoption for teaching support tools are ease of use and trust in the system (see "The Trust Challenge: Adopting Automated Tools in the Classroom"). We have discussed a tool to help with the former. The latter can be facilitated by users' understanding of the capabilities of the tool and their agency in its integration into teaching. Nazaretsky et al. (2022) argue specifically for SAG that it is key to both communicate that grading differences are to be expected even among two humans and to enable teachers to review and revise machine grades.

We propose an evaluation process that allows teachers to gauge the performance of a grading pipeline on their students' data, apply their own quality requirements and choose which automated grades to keep and which to review. In short, the evaluation process acknowledges the imperfection of today's SAG models, but allows teachers to actively make the most of what is available by choosing exactly how much to rely on the model in a hybrid human-machine grading setting.

The workflow integrates steps from the classical data mining and artificial intelligence model development cycle (e.g., CRISP-DM, Shearer, 2000) – data preparation, model training and result analysis – as well as an explicit step for setting requirements for the resulting model (cf. the reliability testing workflow proposed by Tan et al., 2021) and a step for deciding on the best integration of human and machine grading in each special case. In the context of winning teachers' trust, the step of defining requirements for model performance raises awareness for the expected level of grading disagreement in any grading process, be it manual or automated. It also allows teachers to adapt the maximum acceptable level of disagreement to their concrete use case, be it formative or summative testing. The decision step at the end emphasizes teachers' agency in using automated grading tools for their own goals in their own setting.

The process is as follows:

1. Teachers **define** the maximum affordable disagreement between machine and human grader depending on their test situation – sample human-human disagreement rates from real data are provided
2. Teachers **collect** a data set of manually annotated student answers (e.g., from last year's exam)

3. Optionally: Teachers **train or fine-tune** a grading model on this data to improve performance – a publicly available model can also be used
4. Teachers **analyze** the grading model's performance on the level of individual grade labels (like *correct*/*incorrect*)
5. Teachers **decide** on the basis of Step 4 which grade predictions to accept and which to revise in order to save effort, while grade disagreement with their own judgment stays below their required threshold

We will now discuss these steps in turn from a theoretical perspective and then apply the process to the available German data in "Sample Application" to demonstrate the results for several data sets.

Defining Requirements

An automated grader in teaching should save grading effort and yield a grading result at the same level of quality as human grading, or even better. However, these general expectations are quite imprecise – a concrete minimum requirement for model performance and effort saved are needed. We therefore discuss concrete examples from the literature to demonstrate which levels of grading quality have been deemed acceptable in the past and which amount of saved labor can realistically be expected.

Grading disagreement

We measure the performance of a SAG model in terms of its agreement with the gold standard grade annotations on the test data. Therefore, it is easy to conceptualize imperfect predictions as grading error on the part of the machine. While differences between a machine and the gold standard can be due to erroneous predictions just as differences between a human grader and the gold standard can be due to tiredness or oversight, differences between different annotators can also be caused by different interpretations of the student answer or gray areas in the scoring rubrics. Therefore, a certain amount of disagreement in the grade assignments of any two graders (human or machine) are to be expected.

Mieskes and Padó (2018) surveyed a number of publicly available SAG corpora with double human grade annotation. They determined that human-human grading disagreement of up to 15% has been accepted in the past for published data from ad-hoc, small- to medium scale testing like in-class tests. Nazaretsky et al. (2022) find similar average disagreement between different teachers grading the same answers. For standardized, large-scale testing, the observed human-human disagreement is at less than 10% in Mieskes and Padó's survey, due to grader training and quality control measures.

These values can serve as a guideline to the amount of grading disagreement we can accept from a machine grader. In fact, the decision made by Vittorini et al. (2021) to use their 89% correct model for stand-alone grading in low-stakes testing and to additionally review its grades in high-stakes testing fits well within these boundaries.

Distribution of disagreement

Another important aspect of grader disagreement is its distribution over the different grade labels. In a *correct-incorrect* grading task, any disagreement would ideally occur equally in both grade labels, but both humans and machine graders may well show a grading tendency by being too strict (rejecting correct answers) or too lenient (accepting wrong answers). Depending on the context, one tendency or the other may be more desirable.

Workload reduction

A desire to save grading time and effort is often the driving factor behind the use of automated graders, so users will require a substantial reduction in grading time in order to consider using a model. The minimum required reduction depends on any specific user's time budget. In terms of what can be realistically expected of a system, Vittorini et al. (2021) report a grading time reduction of about 40%.

Data Collection

In the context of our workflow, manually graded student answers are required in two places: As test data for the analysis step, and, optionally, for model training and fine-tuning to improve model quality (recall "Method")

As we have seen in "SAG for German: Results", the performance of the same model on different data sets can vary substantially, even after the optional model training step. Without model training, performance is expected to be even lower. Therefore, evaluation results from literature data cannot be expected to carry over identically to a new test set.

Teachers who wish to profit most from evaluating and analyzing model performance therefore should collect a context-specific test set. The data might for example consist of graded answers from old tests that are similar to the ones that the system will be grading. The test set should be as large as possible to make it robust to chance fluctuations; a size in the hundreds of answers is realistic.

Ideally, this data set would be graded by multiple teachers to provide an insight into the reliability of the human grades (in terms of annotator agreement) and into the acceptable error level for the setting (in terms of Accuracy).

Training or Fine-Tuning

If large amounts of data (in the thousands of manually graded answers) are available, the grading model can be optimized for this data set. The fastest option to do this for our German Transformer model is to train the logistic regression classifier in the SBERT architecture – this is the component that is doing grade prediction given input sentence representations generated by the earlier model steps.

It is also possible to further fine-tune the language model used for creating the sentence representations. This takes some technical expertise, but it is a plausible

improvement strategy which adds vocabulary and samples of the expected writing style to the language model. Note that we did not take this step for our experiments.

Analysis

A number of different standard measures exist for evaluating SAG. In our experiments, we take the view of grading as a classification task, where pre-defined labels are assigned to the test instances. In this case, standard measures are Accuracy, Precision, Recall and F_1 Score.

When speaking about grading disagreement between machine and human grader (as above in "Defining Requirements"), we are more closely aligned with the concept of grading as parallel annotation by several graders. The standard evaluation method for this operationalization of grading is Cohen's κ.

We nonetheless opt for the classification evaluation measures because they are easily interpreted as percentages in the different analysis steps: Overall grading differences between the SAG model and the human gold standard become visible in system Accuracy, which gives the percentage of answers where the model agrees with the gold standard. This means that a SAG model with Accuracy of 85% agrees with the human gold standard in 85% of cases and disagrees in the remaining 15%. Note that we have no way of determining which portion of the remaining disagreement is true error (and the resulting machine prediction would never agree with any human grader's) and which portion is due to the gray area of interpretation that mostly causes disagreement between humans.

Importantly, evaluation should be carried out not just for the whole set of answers, but also for subsets, in order to zoom in on grading tendencies. We can do this by inspecting the label-specific Precision. If 90% of all student answers that were graded *correct* by the machine also received this label by a human grader, the assignment precision of the grade label *correct* Prec_{corr} is 90.

A SAG model with a tendency to over-predict *correct* will likely have a lower Prec_{corr} score, since many of student answers with the overly frequent machine prediction *correct* will be graded *incorrect* by the human gold standard. A SAG model with a tendency to under-predict *correct* will have a low Prec_{incorr} performance instead, since it will assign *incorrect* in many cases where the human gold standard says *correct*.

Decision on Usage

After the performance analysis, the automated grader's fulfillment of the required standards for grading error, grading tendencies and workload reduction can be evaluated. This tells the teacher what usage settings are optimal given the observed performance and the stated requirements (for example, hybrid usage only).

In this way, teachers can evaluate their individual cost-benefit ratio of using an automated system over standard manual grading based on a realistic picture of what the automated grader is able to contribute in their classroom setting.

Sample Application

We now apply our process to the German data sets. We **define the requirements** to be in line with the literature findings (see "Defining Requirements"): The output of a grading cycle should have at most 15% grading disagreement between human and machine grader, in line with common human-human agreement. If the machine grades disagree with the human gold standard, machine grades should optimally be in favor of the students, i.e., accepting gold standard *incorrect* answers rather than rejecting gold standard *correct* answers. Projected workload reduction should be around 40%.

The **data** consists of the German corpora we used above, i.e., CSSAG, CREG, ASAP-DE and SAF-DE. These are treated as four separate data samples and stand in for teacher-collected data samples as described in "Data Collection".

For three of these corpora, κ values for double human annotation are provided in the literature. (CREG only contains student answers for which the graders agreed.) CSSAG annotators achieved Fleiss' κ of 0.54 and Accuracy of 89.2% for binary grading (Mieskes and Padó, 2018). ASAP-DE grade annotations agree "between .58 and .84 quadratically weighted κ" (Horbach et al., 2018) on three or four labels (depending on the prompt). SAF-DE uses five labels, and annotators reached 0.78 Krippendorff's α and 81.4% Accuracy (Filighera et al., 2022).

Even though each work reports a different κ formulation and κ is not always available for the binary grading done here, we conclude that the human grades reported for the four corpora are reliable. Our demonstration of the grading support process now aims at showing how human grading effort can be saved while still achieving grading outcomes within the requirements.

We **train** the classification step of the German Transformer model on a 90% split of CSSAG and apply this model to the unseen 10% CSSAG test split as well as the whole of CREG, ASAP-DE and SAF-DE (including both SAF-DE test sets). This mimics a situation for CREG, ASAP-DE and SAF-DE where a model trained on a different corpus is re-used without further adaption, which is realistic in practice due to lack of data or technical expertise. Note that we expect the model Accuracy on CREG, ASAP-DE and SAF-DE to be lower than in Table 3, where we reported the performance of models specifically trained on the individual data sets: In general, the more different the evaluation data is from the training data, the more the performance of Machine Learning models suffers.

We choose the CSSAG model as the prediction model because (1) it performs best of all models in comparison to the relevant literature and (2) it was trained on a relatively large corpus with balanced label distribution, which will hopefully enable it to generalize well.

Table 7 shows the results for **analysis**. First, we find that the Accuracy for CREG, ASAP-DE and SAF-DE is indeed lower than for the tuned models in Table 3, as expected. This underscores again the need to test existing models against samples of teachers' own data, since the size of the performance drop varies notably between corpora and is hard to predict. The drop in Accuracy is most drastic for ASAP-DE (loss of 22 percentage points) and SAF-DE (loss of 15 percentage points). On the face of it, this raises the question of how useful the model predictions can still be for these corpora.

Table 7 Applying the CSSAG German Transformer model to the CSSAG test set, CREG, ASAP-DE and SAF-DE: Overall Accuracy and label-specific Precision. Hybrid Accuracy and Workload Reduction are for hybrid machine-human grading, assuming that the labels with the higher label-specific Precision are accepted, and the others are manually reviewed

Data Set	Acc	Prec_{corr}	Prec_{incorr}	Hybrid Acc	Workload Reduction
CSSAG Test	84.8	83.2	86.5	0.94	46.4%
CREG	66.0	69.1	63.7	0.87	42.0%
ASAP-DE	43.2	28.3	85.9	0.96	25.9%
SAF-DE	55.0	85.3	24.7	0.93	50.0%

However, when looking at grade-wise Precision, predictions are still quite reliable for at least one grade: Strikingly, for ASAP-DE with an overall Accuracy of just 43%, 86% of all predictions of *incorrect* are still in agreement with the gold standard human grades. Similarly, SAF-DE has an overall Accuracy of 55%, but predictions of *correct* still agree with the gold standard in 85% of cases. CREG shows the least difference between grade-wise Precisions, probably because the data set is perfectly balanced between grade labels.

The CSSAG model on the CSSAG test set is the best case of a corpus-specific grading model and almost fulfills the workflow requirements as it stands: 84.8% of all predictions are accurate. Additionally, the remaining grading disagreement is quite well-balanced between grade labels, with a slight inclination towards leniency (the model more often accepts *incorrect* answers than rejecting *correct* ones). Accepting all grade predictions without review would of course save 100% of grading effort, so the workload reduction criterion would also be satisfied.

Hybrid grading

For CREG, SAF-DE and especially ASAP-DE, stand-alone use of the model predictions is not realistic. However, even these model predictions can still be used to reduce a teacher's workload while maintaining acceptable overall agreement between machine and human grader: In a hybrid setting, the teacher can accept all grade predictions made for the grade that the model predicts most reliably, and manually review the remaining predictions, correcting them as necessary. The two rightmost columns in Table 7 show the overall Hybrid Accuracy and the workload reduction due to hybrid grading given in percent of all students answers. The Hybrid Accuracy is the overall agreement with the gold standard human grade that would result from accepting the more reliable label prediction and manually reviewing the remaining predictions.[9]

This strategy would result in an overall percentage of remaining disagreement of at or below 15% for all four corpora and a workload reduction between 26 and 50% of answers. There will even be some additional speedup compared to pure manual grading, since it is faster to review grades than to assign them from scratch (Vittorini et al., 2021).

The outcome of the hybrid grading process satisfies our criteria for CSSAG, CREG and SAF-DE. Adding a human review step for some of the predicted grades would

[9] We (somewhat optimistically) assume that human-reviewed grades are always correct.

reduce disagreement on grades for CSSAG from 15 to 6% across all student answers, while still saving human grading of almost half of the student answers. This would however introduce a small bias to the detriment of students, since the remaining disagreement would stem from *correct* answers that were automatically labelled *incorrect* and therefore not revised.

For CREG, 13% of disagreement remain in the hybrid setting (this time, in favor of the students) and 40% of answers can be skipped in manual grading. For SAF-DE, the large label imbalance of the data set leads to a workload reduction of 50% at 7% remaining disagreement if all predictions of *correct* are accepted. Only answers that were predicted to be *incorrect* would need to be reviewed, and any remaining grading disagreement would again be in favor of the students.

Despite the poor model performance for ASAP-DE, hybrid grading would even bring the overall level of disagreement down to 4% when accepting all answers labelled as *incorrect* and manually revising all others. However, this means that the remaining prediction disagreement is to the detriment of students, since *correct* answers mislabelled as *incorrect* would not be reviewed. Also, due to the label distribution of the corpus, this translates to only 26% of annotation effort saved, which does not meet our labor saving criterion.

Based on this result pattern, users can now **decide** on whether and how to use the system. Users with data like the CSSAG test set can choose between fully automated and hybrid grading since both settings fulfil our requirements. Results like for CREG and SAF-DE invite hybrid grading, yielding substantial effort reduction and grading disagreement rates below our required threshold of 15% along with remaining disagreement in favor of the students.

Faced with a result like for ASAP-DE, a teacher might decide not to use the system and rely on full manual grading as before. Alternatively, given the very high Hybrid Accuracy, the annotation effort saved might even still be worthwhile.

Importantly, the decision for any of these strategies is fully in the hands of the teacher, and is based on a sample of their own data. Therefore, teachers have a clear expectation of how well machine grades will agree with their own judgment and how strong this agreement will be after hybrid grading. They can also communicate their assessment of the chosen strategy and its quality clearly to their students to build their trust in turn.

Conclusions

We have highlighted three challenges for the broad-scale application of automated Short-Answer Grading (SAG) to student answers, especially for languages other than English: A **technical** challenge regarding the need for automated SAG models of sufficient quality, given that fewer resources exist for training and model development; a **usability** challenge in adapting high-quality research prototypes to the needs of users who are not programming experts, and a **trust** challenge in communicating the abilities and limitations of the tools and keeping the intended users in control of grading in their classrooms.

We have addressed these challenges in turn, proposing an automated SAG model for German wrapped by a GUI that was designed with usability for all in mind, as well as a workflow that enables teachers to evaluate the model on their data and decide on the best usage for the tool in their context, while realizing workload reduction at acceptable rates of disagreement between the machine and human graders.

Focusing on German, we have demonstrated that our SAG model gracefully deals with syntactic and morphological properties of German that differ from English. Due to the nature of the multilingual language model used in model creation, our German Transformer model can be applied to texts in 49 more languages (as diverse as Arabic, Danish, Finnish, Chinese or Italian) as easily as to a new German data set. Analyses of performance and linguistic appropriateness for (some of) these languages are interesting avenues for future work on the technical challenge.

One limitation of the German Transformer model is our decision to work with the dichotomous grading case (the *correct-incorrect* decision). While technically, the German Transformer model can be trained to cover partial credit (as additional grade labels), the integration of machine grades and human review becomes much more complex in this case. The larger the number of available labels, the harder the task of consistently deciding between the classes. This means annotators (human or machine) will disagree more often than for the dichotomous case. Also, the interpretation of this disagreement becomes more challenging: Likely, practitioners will view disagreement about which of two adjacent grade labels to choose as less serious than disagreement about two grade labels that are far apart (like full and zero marks). This complexity would need to be considered in choosing the machine grades that a teacher is ready to accept as reliable in the partial credit case.

We addressed the usability challenge by presenting the ASYST GUI for easy application of the German Transformer model and demonstrating its usability for expert teachers. Development of ASYST is ongoing, with better support for grading model evaluation and analysis a major focus for future work. ASYST currently assumes that student answers are easily accessible in spreadsheet format. In Germany, this is plausible to assume at the university level, where learning management systems (LMS) are available and accessible for testing. However, at the level of primary and secondary education, availability and use of LMS varies widely among schools. For many teachers, this means a gap between handwritten student answers and digital automated grading that is not currently addressed by ASYST.

To demonstrate possible outcomes of using our proposed workflow, we have applied it to four German data sets and found that by focusing human effort where it is most needed, even imperfect SAG models can still be used to save grading effort and produce reliable grade predictions.

The hybrid nature of our proposed workflow and the focus on the teacher's freedom to set requirements for the German Transformer model's output avoids many of the ethical questions associated with automated grading in the classroom: The performance of the automated grading model is scrutinized in a realistic setting and unreliable grade predictions are rejected out of hand. Therefore, students are not graded by a black box model with unknown performance and biases. In hybrid grading, model predictions can even serve as a corrective to possible teacher grading biases, while the teacher can identify model bias in the grade predictions under human review. What remains

potentially problematic is the amount of disagreement between human and machine (small by design) on those grade labels that are accepted without review. In these cases, is not possible at this point to distinguish between true grading errors on the part of the machine and disagreement of interpretation as between humans, and the danger of introducing a machine bias remains.

However, machine involvement need not be the same across all settings – teachers choose the amount of grading support (and with it of possible machine bias and true error) that is acceptable to them for each individual test. This means that hybrid grading for high-stakes testing with its rigorous requirements for grading agreement and absence of bias will be different from low-stakes testing, for example in weekly feedback tests.

With an easily usable automated SAG tool and workflow available, we believe that everything is in place for future work on the trust challenge by inviting feedback from the field on both usability and workload reduction in real-life settings, and on teachers' views of the ethical consequences of hybrid grading, as well.

Author Contributions Ulrike Padó contributed the overall conception of the paper as well as the workflow process and the analysis of sample applications. Yunus Eryilmaz trained the German Transformer model and determined optimal parameters. Larissa Kirschner designed and implemented the ASYST GUI. The various drafts of the manuscript were written by Ulrike Padó and all authors commented on them. All authors read and approved the final manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL. Ulrike Padó acknowledges funding through Bundesministerium für Bildung und Forschung (BMBF), grant 16DHBKI072, project KNIGHT. Yunus Eryilmaz and Ulrike Padó acknowledge support by the state of Baden-Württemberg through the use of the bwHPC high-performance computing infrastructure.

Declarations

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

References

Andersen, N., & Zehner, F. (2021). shinyReCoR: A Shiny Application for Automatically Coding Text Responses Using R. *Psych, 3*(3), 422–446.

Attali, Y., & Burstein, J. (2006). Automated essay scoring with e-rater®v2. *Journal of Technology, Learning, and Assessment, 4*(3).

Azad, S., Chen, B., Fowler, M., West, M., & Zilles, C. (2020). Strategies for deploying unreliable AI graders in high-transparency high-stakes exams. *Proceedings of the International Conference on Artificial Intelligence in Education* Vol. 12163.

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*.

Bangor, A., Kortum, P., & Miller, J. (2009). Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *Journal of Usability Studies, 4*(3), 114–123.

Basu, S., Jacobs, C., & Vanderwende, L. (2013). Powergrading: A clustering approach to amplify human effort for short answer grading. *Transactions of the Association for Computational Linguistics, 1*, 391–402.

Bexte, M., Horbach, A., & Zesch, T. (2022). Similarity-Based Content Scoring - How to Make S-BERT Keep Up With BERT. *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)* pp. 118–123.

Bexte, M., Horbach, A., & Zesch, T. (2023). Similarity-based content scoring - a more classroom-suitable alternative to instance-based scoring? *Findings of the Association for Computational Linguistics: ACL, 2023*, 1892–1903.

Brooke, J. (1995). *SUS: A quick and dirty usability scale* (Vol. 189). Ind.: Usability Eval.

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.

Burstein, J., Shore, J., Sabatini, J., Moulder, B., Holtzman, S., & Pedersen, T. (2012). *The "Language Muse" System: Linguistically Focused Instructional Authoring*. Research Report. ETS RR-12-21. (Tech. Rep.). Princeton, NJ: Educational Testing Service.

Camus, L., & Filighera, A. (2020). Investigating Transformers for automatic short answer grading. *Proceedings of the International Conference on Artificial Intelligence in Education* pp. 43–48.

Common Crawl Project (2023). *Statistics of common crawl monthly archives- distribution of languages*. https://commoncrawl.github.io/cc-crawl-statistics/plots/languages [Online; accessed 10-November-2023]

Condor, A. (2020). Exploring automatic short answer grading as a tool to assist in human rating. *Proceedings of the International Conference on Artificial Intelligence in Education, 12164*, 74–49.

Condor, A., Litster, M., & Pardos, Z. (2021). Automatic short answer grading with SBERT on out-of-sample questions. *Proceedings of the 14th International Conference on Educational Data Mining*(EDM21) pp. 345–352. International Educational Data Mining Society.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly, 13*(3), 319–340.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional Transformers for language understanding. *Proceedings of the 2019 conference of the NAACL:HLT* pp. 4171–4186.

Ding, Y., Riordan, B., Horbach, A., Cahill, A., & Zesch, T. (2020). Don't take "nswvtnvakgxpm" for an answer - The surprising vulnerability of automatic content scoring systems to adversarial input. *Proceedings of the 28th International Conference on Computational Linguistics* p. 882–892.

Dzikovska, M., Nielsen, R., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., & Dang, H. T. (2013). SemEval-2013 task 7: The Joint Student Response Analysis and 8th Recognizing Textual Entailment Challenge. *Proceedings of SemEval, 2013*, 263–274.

Filighera, A., Ochs, S., Steuer, T., & Tregel, T. (2023). Cheating automatic short answer grading with the adversarial usage of adjectives and adverbs. *International Journal of Artificial Intelligence in Education*.

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. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics* pp. 8577–8591.

Galitz, W.O. (2007). *The essential guide to user interface design: An introduction to GUI design principles and techniques* (3rd ed.). Wiley.

Ghavidel, H.A., Zouaq, A., & Desmarais, M.C. (2020). Using BERT and XLNET for the automatic short answer grading task. *Proceedings of the International Conference on Computer Supported Education* pp. 58–67.

Gombert, S., Di Mitri, D., Karademir, O., Kubsch, M., Kolbe, H., Tautz, S.,.& Drachsler, H. (2023). Coding energy knowledge in constructed responses with explainable NLP models. *Journal of Computer Assisted Learning, 39*(3), 767–786.

Hahn, M., & Meurers, D. (2012). Evaluating the meaning of answers to reading comprehension questions: A semantics-based approach. *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP* pp. 326–336.

Hessel, J., & Schofield, A. (2021). How effective is BERT without word ordering? Implications for language understanding and data privacy. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (short papers)* pp. 204–211.

Horbach, A., Stennmanns, S., & Zesch, T. (2018). Cross-lingual content scoring. *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications* pp. 410–419.

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* pp. 328–339.

ISO Technical Committee, I.S.. (2018). Ergonomics of human-system interaction –part 11: Usability: Definitions and concepts. *International Organisation for Standardization.*

Jimenez, S., Becerra, C., & Gelbukh, A. (2013). Softcardinality: Hierarchical text overlap for student response analysis. *Proceedings of SemEval, 2013*, 280–284.

Johnson, J. (2013). *Designing with the mind in mind*. Morgan Kaufmann.

Krumm, H.-J., Fandrych, C., Hufeisen, B., & Riemer, C. (2011). *Deutsch als Fremd- und Zweitsprache. Ein internationales Handbuch*. Berlin, New York: De Gruyter Mouton.

Kumar, Y., Aggarwal, S., Mahata, D., Shah, R.R., Kumaraguru, P., & Zimmermann, R. (2019). Get IT scored using AutoSAS – an automated system for scoring short answers. *Proceedings of the AAAI Conference on Artificial Intelligence* p. 9662–9669.

Meta (2023). *List of Wikipedias–Meta, discussion about Wikimedia projects*. https://meta.wikimedia.org/w/index.php?title=ListofWikipedias&oldid=25452928 [Online; Accessed 10-November-2023]

Meurers, D., Ziai, R., Ott, N., & Bailey, S. (2011). Integrating parallel analysis modules to evaluate the meaning of answers to reading comprehension questions. *International Journal of Continuing Engineering Education and Life-Long Learning, 21*(4), 355–369.

Meurers, D., Ziai, R., Ott, N., & Kopp, J. (2011). Evaluating answers to reading comprehension questions in context: Results for German and the role of information structure. *Proceedings of the Textinfer Workshop on Textual Entailment* pp. 1–9. Edinburgh, Scotland, UK.

Mieskes, M., & Padó, U. (2018). Work smart - reducing effort in short-answer grading.*Proceedings of the 7th Workshop on NLP for Computer Assisted Language Learning* pp. 57–68.

Mizumoto, T., Ouchi, H., Isobe, Y., Reisert, P., Nagata, R., Sekine, S., & Inui, K. (2019). Analytic score prediction and justification identification in automated short answer scoring. *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications* pp. 316–325.

Mohler, M., Bunescu, R., & Mihalcea, R. (2011). Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* pp. 752–762. Portland, OR.

Nazaretsky, T., Ariely, M., Cukurova, M., & Alexandron, G. (2022). Teachers' trust in AI-powered educational technology and a professional development program to improve it. *British Journal of Educational Technology, 53*(4), 914–931.

Ott, N., Ziai, R., Hahn, M., & Meurers, D. (2013). CoMeT: Integrating different levels of linguistic modeling for meaning assessment. *Proceedings of SemEval, 2013*, 608–616.

Padó, U. (2016). Get semantic with me! The usefulness of different feature types for short-answer grading. *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical papers* pp. 2186–2195.

Padó, U. (2022). Assessing the practical benefit of automated shortanswer graders. *Proceedings of the International Conference on Artificial Intelligence in Education* p. 555–559.

Padó, U., & Kiefer, C. (2015). Short answer grading: When sorting helps and when it doesn't. *Proceedings of the Workshop on NLP for Computer-Aided Language Learning* p. 42–50. Vilnius, Lithuania.

Poulton, A., & Eliëns, S. (2021). Explaining Transformer-based models for automatic short answer grading. *Proceedings of the 5th International Conference on Digital Technology in Education* pp. 110–116.

Ramachandran, L., Cheng, J., & Foltz, P. (2015). Identifying patterns for short answer scoring using graph-based lexico-semantic text matching. *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications* pp. 97–106.

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* pp. 3982–3992.

Reimers, N., & Gurevych, I. (2020). Making monolingual sentence embeddings multilingual using knowledge distillation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing* pp. 4512–4525.

Riordan, B., Horbach, A., Cahill, A., Zesch, T., & Lee, C.M. (2017). Investigating neural architectures for short answer scoring. *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications* pp. 159–168.

Saha, S., Dhamecha, T.I., Marvaniya, S., Sindhgatta, R., & Sengupta, B. (2018). Sentence level or token level features for automatic short answer grading?: Use both. *Proceedings of the International Conference on Artificial Intelligence in Education* pp. 503–517.

Schneider, J., Richner, R., & Riser, M. (2022). Towards trustworthy autograding of short, multi-lingual, multi-type answers. *International Journal of Artificial Intelligence in Education*.

Shearer, C. (2000). The CRISP-DM model: the new blueprint for data mining. *Journal of Data Warehousing, 5*, 13–22.

Shneiderman, B. (1987). *Designing the user interface: Strategies for effective human-computer interaction* (1st ed.). Addison-Wesley.

Shneiderman, B., Pleasant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2016). *Designing the user interface: Strategies for effective human-computer interaction* (6th ed.). Pearson.

Steimel, K., & Riordan, B. (2020). Towards instance-based content scoring with pre-trained Transformer models. *workshop on artificial intelligence for education* (AI4EDU@AAAI).

Sultan, M.A., Salazar, C., & Sumner, T. (2016). Fast and easy short answer grading with high accuracy. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* pp. 1070–1075.

Sung, C., Dhamecha, T., Saha, S., Ma, T., Reddy, V., & Arora, R. (2019). Pre-training BERT on domain resources for short answer grading. *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.

Tan, S., Joty, S., Baxter, K., Taeihagh, A., Bennett, G.A., & Kan, M.-Y. (2021). Reliability testing for natural language processing systems. *Proceedings of ACL-IJCNLP* p. 4153–4169.

Törnqvist, M., Mahamud, M., Mendez Guzman, E., & Farazouli, A. (2023). ExASAG: Explainable framework for automatic short answer grading. *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications* (BEA 2023) pp. 361–371.

Vaswani, A., Jones, L., Shazeer, N., Parmar, N., Uszkoreit, J., Gomez, A.N., Kaiser, Ł., . . .& Polosukhin, I. (2017). Attention is all you need. *Proceedings of the 31st Conference on Neural Information Processing Systems* (NIPS 2017).

Venkatesh, V., & Davis, F. D. (2000). A theoretical extension of the technology acceptance model: Four longitudinal field studies. *Management Science, 46*(2), 186–204.

Vittorini, P., Menini, S., & Tonelli, S. (2021). An AI-based system for formative and summative assessment in Data Science courses. *International Journal of Artificial Intelligence in Education, 31*, 159–185.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., & Bowman, S. (2018). GLUE: A multi-task benchmark and analysis platform for natural language understanding. *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP* pp. 353–355.

Willms, N., & Padó, U. (2022). A Transformer for SAG: What does it grade?. *Proceedings of the 11th Workshop on NLP for Computer Assisted Language Learning* pp. 114–122.

Wöllstein, A. (2014). *Topologisches Satzmodell*. Heidelberg: Winter.

Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., . . . & Dean, J. (2016). *Google's neural machine translation system: Bridging the gap between human and machine translation*. arXiv:1609.08144 .

Yuen, A. H. K., & Ma, W.W.-K. (2008). Exploring teacher acceptance of elearning technology. *Asia-Pacific Journal of Teacher Education, 36*, 229–243.

Zesch, T., & Horbach, A. (2018). ESCRITO - an NLP-enhanced educational scoring toolkit. *Proceedings of the Eleventh International Conference on Language Resources and Evaluation* (LREC-2018).

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* (pp. 44–58). Educational Measurement: Issues and Practice.

Zesch, T., Levy, O., Gurevych, I., & Dagan, I. (2013). UKP-BIU: Similarity and entailment metrics for student response analysis. *Proceedings of SemEval, 2013*, 285–289.

Zhai, N., & Ma, X. (2022). Automated writing evaluation (AWE) feedback: a systematic investigation of college students' acceptance. *Computer Assisted Language Learning, 35*(9), 2817–2842.

Zhu, M., Liu, O.L., & Lee, H.-S. (2020). The effect of automated feedback on revision behavior and learning gains in formative assessment of scientific argument writing. *Computers & Education*, *143*.