Gombert, Sebastian; Di Mitri, Daniele; Karademir, Onur; ...

Coding energy knowledge in constructed responses with explainable NLP models

Journal of computer assisted learning 39 (2022) 3, S. 767-786

Quellenangabe/ Reference:
Gombert, Sebastian; Di Mitri, Daniele; Karademir, Onur; Kubsch, Marcus; Kolbe, Hannah; Tautz, Simon; Grimm, Adrian; Bohm, Isabell; Neumann, Knut; Drachsler, Hendrik: Coding energy knowledge in constructed responses with explainable NLP models - In: Journal of computer assisted learning 39 (2022) 3, S. 767-786 - URN: urn:nbn:de:0111-pedocs-284415 - DOI: 10.25656/01:28441; 10.1111/jcal.12767

https://nbn-resolving.org/urn:nbn:de:0111-pedocs-284415
https://doi.org/10.25656/01:28441

Received: 10 December 2021 | Revised: 11 September 2022 | Accepted: 18 November 2022

DOI: 10.1111/jcal.12767

ARTICLE

Journal of Computer Assisted Learning WILEY

Coding energy knowledge in constructed responses with explainable NLP models

Sebastian Gombert[1] | Daniele Di Mitri[1] | Onur Karademir[1] | Marcus Kubsch[2] | Hannah Kolbe[2] | Simon Tautz[2] | Adrian Grimm[2] | Isabell Bohm[3] | Knut Neumann[2] | Hendrik Drachsler[1,4,5]

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

[2]Physics Education, IPN: Leibniz Institute for Science and Mathematics Education, Kiel, Germany

[3]Educational Psychology and Technology, Ruhr University, Bochum, Germany

[4]Studiumdigitale, Goethe University, Frankfurt am Main, Germany

[5]Welten Institute - Research Centre for Learning, Open University of the Netherlands, Heerlen, Netherlands

Correspondence
Sebastian Gombert, Educational Technologies, DIPF: Leibniz Institute for Research and Information in Education, Frankfurt am Main, Germany
Email: s.gombert@dipf.de

Funding information
AFLEK Projekt, funded by Bundesministerium für Bildung und Forschung, Grant/Award Number: 01JD2008; ALICE project, funded by Leibniz-Gemeinschaft, Grant/Award Number: K365/2020

Abstract

Background: Formative assessments are needed to enable monitoring how student knowledge develops throughout a unit. Constructed response items which require learners to formulate their own free-text responses are well suited for testing their active knowledge. However, assessing such constructed responses in an automated fashion is a complex task and requires the application of natural language processing methodology. In this article, we implement and evaluate multiple machine learning models for coding energy knowledge in free-text responses of German K-12 students to items in formative science assessments which were conducted during synchronous online learning sessions.

Dataset: The dataset we collected for this purpose consists of German constructed responses from 38 different items dealing with aspects of energy such as manifestation and transformation. The units and items were implemented with the help of project-based pedagogy and evidence-centered design, and the responses were coded for seven core ideas concerning the manifestation and transformation of energy. The data was collected from students in seventh, eighth and ninth grade.

Methodology: We train various transformer- and feature-based models and compare their ability to recognize the respective ideas in students' writing. Moreover, as domain knowledge and its development can be formally modeled through knowledge networks, we evaluate how well the detection of the ideas within responses translated into accurate co-occurrence-based knowledge networks. Finally, in terms of the descriptive accuracy of our models, we inspect what features played a role for which prediction outcome and if the models pick up on undesired shortcuts. In addition to this, we analyze how much the models match human coders in what evidence within responses they consider important for their coding decisions.

Results: A model based on a modified GBERT-large can achieve the overall most promising results, although descriptive accuracy varies much more than predictive accuracy for the different ideas assessed. For reasons of comparability, we also evaluate the same machine learning architecture using the SciEntsBank 3-Way

benchmark with an English RoBERTa-large model, where it achieves state-of-the-art results in two out of three evaluation categories.

KEYWORDS
automated coding, constructed response assessment, energy didactics, energy transformation, knowledge networks, short answer scoring

1 | INTRODUCTION

In online learning, the data generated by learners within learning management systems can be utilized for monitoring and supporting them, for example, in the form of automatically provided feedback, an endeavor often referred to as Learning Analytics (Greller & Drachsler, 2012; LAK, 2011). To design and administer effective ways of supporting students throughout units, their active knowledge needs to be tested via formative assessments. Within such assessments, especially open-ended tasks require students to use and recombine acquired knowledge actively. For this reason, respective items can provide a good indication of what active knowledge students possess (Livingston, 2009; Lukhele et al., 1994). If one uses such assessments to monitor the development of students' knowledge as they progress through a unit, one can gain an insight into how student knowledge evolves over time, and where potential knowledge gaps develop. This can, in turn, be used to provide students with appropriate feedback and scaffolding. However, to assess the development of students' knowledge, their responses need to be coded first. Coding responses to open-ended tasks by hand is expensive and labor-intensive. Creating systems to automate this procedure promise to speed up this process, but this endeavor is far from trivial. The automation which is needed here can be described as a case of automated constructed response scoring. Scoring constructed responses in an automated fashion has a rich and vivid history and was approached with various methods from natural language processing, ranging from different forms of word- and pattern matching to machine learning (Burrows et al., 2015). For this reason, one can rely on a fundus of different methods which can be applied to the problem.

For this particular study, the core aim was to implement and evaluate systems for the automatic coding of seven different core ideas from the domain of energy physics within German K12 short responses. For this purpose, we implemented two transformer-based and five feature-based machine learning models, as well as several baselines. The responses used to train and evaluate these were taken from formative assessments within synchronous online units designed under the paradigms of project-based pedagogy (Krajcik & Shin, 2014) and evidence-centered design (Mislevy et al., 2003; Mislevy & Haertel, 2007; Pellegrino et al., 2016). The dataset we collected includes the responses of 305 German secondary school students from Schleswig-Holstein (school forms *Gemeinschaftsschule* and *Gymnasium*) to a set of 38 different constructed response items.

We evaluated the predictive performance of our models using F1 scores to test their reliability in predicting the correct codes for individual responses. Following this, we tested to which degree the individual codes translated into accurate representations of students' overall domain knowledge. To represent the latter in a formal manner, we used knowledge networks (Kubsch et al., 2019; Shaffer et al., 2016; M. S. C. Thomas & McClelland, 2001). We compared the networks generated from the predicted codes to gold standard ones generated from a human-coded ground truth. Overall, a model based on the transformer language model GBERT-large (Chan et al., 2020) achieved the best performance among all approaches tested by us, as it achieved the overall highest F1 scores and the derived knowledge networks were the closest to the gold standard. To demonstrate the general feasibility of our approach and to enable comparability to past work, we also evaluated this best-performing architecture on the *SciEntsBank 3-Way* (Dzikovska et al., 2013) dataset, an established benchmark, where it could achieve new state-of-the-art results. Moreover, we inspected our models for descriptive accuracy, that is, the question if they assign codes to responses for correct reasons. For this purpose, we evaluated if our models had learned undesired shortcuts and to which extent they considered similar signals in responses important as human coders. The models indeed learned a few impactful undesired shortcuts and overall differed in how they matched human coders in different categories. Nonetheless, we could also show that models.

2 | BACKGROUND

2.1 | Energy learning and integrated knowledge networks

Modern science education standards and other educational policy documents stress the importance of students being able to apply scientific ideas to make sense of the natural and engineered world (e.g., National Research Council, 2012; Konferenz der Kultusminister der Länder in der Bundesrepublik Deutschland., 2020). One prerequisite for such *competence* (Weinert, 2002), also referred to as knowledge-in-use (Pellegrino, 2013), is that students possess well-connected knowledge organized around the core ideas of a given domain. The structure of their individual integrated knowledge can be modeled using network models, for example, in the form of knowledge networks (Anderson, 2013; McClelland & Cleeremans, 2009; M. S. C. Thomas & McClelland, 2001). In a constructivist sense, learning can be modeled as a process during which students expand, modify and update their personal knowledge networks as they interact with given course material and construct new knowledge (diSessa, 1988; Kubsch et al., 2019); well-connected

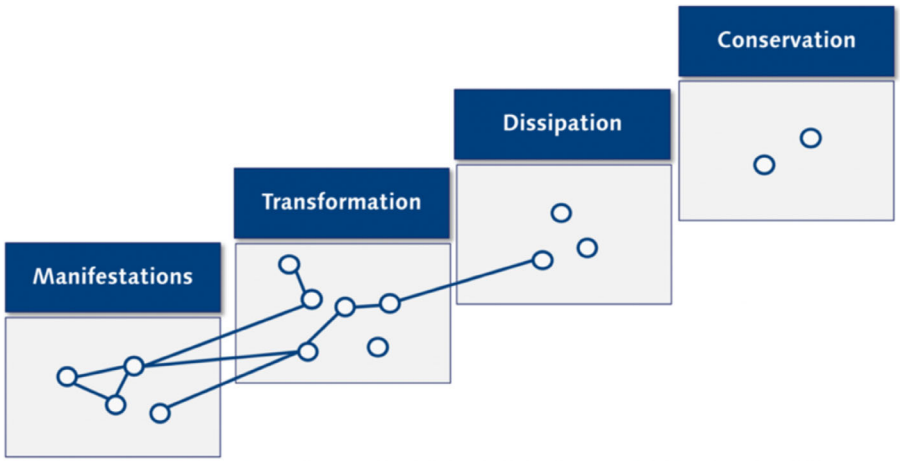

FIGURE 1 A schematic illustration of how knowledge networks modeling a learning progression in energy learning might look after the completion of a couple of tasks by a learner.

knowledge networks should, amongst other benefits, also allow for a fluent retrieval of information (Bransford, 2000; Linn, 2006).

A core concept across the sciences, and especially in the domain of physics, is energy. Research investigating students' optimal progression of them learning about energy in K-12 instruction (drawing on samples from the U.S., China, and Germany, among others) suggests that a typical progression in developing knowledge about energy is to first acquire ideas about the different ways in which energy manifests itself in the real worlds (e.g., that a moving object has kinetic energy). This is followed by an integration of ideas about how different forms of energy can be transformed from one form into another (e.g., the declining speed and increasing height of a pendulum as it moves up indicate kinetic energy being transformed into gravitational energy). Following this notion, students should integrate ideas about degradation and dissipation before being introduced to ideas around the principle of energy conservation (Duit, 2014; Herrmann-Abell & DeBoer, 2018; Liu - et al., 2015; X. Liu & McKeough, 2005; Neumann et al., 2013; Yao et al., 2017). Figure 1 schematically illustrates the interplay between knowledge networks and the described learning progression.

There have been multiple publications dealing with the assessment of students' integrated knowledge in energy didactics. For example, Lee and Liu (2010) proposed a holistic approach for which they first developed coding rubrics for open-ended tasks that specified different levels of connectedness and integration of ideas. They then conducted a study for which students had to solve multiple-choice questions about different energy-related ideas and then provide open-ended explanations for their choice. Afterwards, students' answers were hand-coded accordingly on a five-point scale, indicating how well they connected respective ideas. Liu et al. (2016) successfully automated the respective coding process using support vector machines.

Kubsch et al. (2019) proposed a network analytic approach for assessing the development of students' integrated knowledge throughout a unit. They collected a data set consisting of interview transcriptions and constructed responses from secondary school students working on a 10-week long unit about energy and hand-coded different pre-defined ideas about energy that students used in their responses. Then, based on the co-occurrence of identified ideas within responses, the authors computed network representations of students' demonstrated knowledge. Like related network analytical approaches such as epistemic network analysis (Shaffer et al., 2016), these networks could provide a detailed overview of what ideas students used in their explanations and how they connected them. Feeding such networks back to teachers could, in theory, provide them with valuable information on their students' performance concerning various aspects. Automating the assessment could help to conduct respective network analytic studies on energy learning on a larger scale and should allow for the implementation of feedback and scaffolding systems that send out feedback based on students' knowledge networks.

2.2 | Assessment of short constructed responses

Assessing constructed responses automatically is one of the oldest and most established use cases of natural language processing in educational contexts. In many publications, the methodology has also been referred to as automatic short answer assessment (ASAA) or short answer grading (ASAG). In most cases, authors addressed this problem with the intention to predict holistic scores or grades, ergo assigning a given input text a discrete or continuous value indicating its quality. Burrows et al. (2015) provide a comprehensive literature review addressing important earlier work in the field focusing on short free-text responses.

The earliest work in this field builds upon the notion that students' responses are combinations of different expressed key concepts. A response is graded as correct if enough concepts from a pre-defined concept lexicon are detected in a response. Significant contributions based upon this notion are Burstein et al. (1999), Callear et al. (2001), and Leacock and Chodorow (2003). Other approaches build upon different forms of pattern matching between students' responses and pre-defined sample solutions. Pattern matching can be conducted through a range of different methods such as bag-of-words matching (Cutrone & Chang, 2010; Siddiqi & Harrison, 2008), Boolean matching (Thomas, 2003), matching sub-segments of the parse trees of responses and provided sample solutions (Bachman et al., 2002; Mitchell et al., 2002), or formal semantics (Hahn & Meurers, 2012). Another methodology often applied to the task is latent semantic analysis (Landauer et al., 1998). Systems such as the ones proposed by Zehner

(2016) or Klein et al. (2011) use respective word vectors in combination with clustering algorithms to model the relationships between responses and sample solutions in an unsupervised manner. Andersen and Zehner (2021) introduced a system for conducting such cluster-based scoring with the help of a graphical user interface.

Much of the more recent work on the topic uses different supervised machine learning algorithms. Works such as Hahn and Meurers (2012), Meurers et al. (2011), and Horbach et al. (2013) applied *k*-nearest-neighbor classifiers trained on hand-crafted and semi-hand-crafted feature sets incorporating different lexical and semantic similarity features. Crossley et al. (2016) combined word frequencies, semantic similarity, and psycholinguistic norm features with linear discriminant analysis to score different chemistry tasks. Moreover, with *SemEval-2013 Task 7*, Dzikovska et al. (2013) held a shared task focusing on the topic, where participating systems were evaluated on multiple data sets. Here, an approach based upon an ensemble of support vector machines and logistic regression excelled and achieved the best results in most categories (Ott et al., 2013). Runners-up systems were based on hierarchical pattern matching, decision tree classifiers (Jimenez et al., 2013), and naïve Bayes classification on a mixed feature set combining bag-of-words, word *n*-grams, and different similarity and entailment metrics (Levy et al., 2013).

The corresponding data sets have since been adopted as standard benchmarks for comparing constructed response assessment systems. Sultan et al. (2016) achieved improved results for these data sets with a ridge regressor-based system trained on different semantic similarity features. Saha et al. (2018) applied logistic regression with a feature set combining word-level similarity scores and sentence embeddings to the problem. Another proposed system uses features based on similarity scores and clustering to augment the input with prototypical responses from respective training sets (Marvaniya et al., 2018). For designing such feature-based scoring systems, Zesch and Horbach (2018) introduced a Java framework called *Escrito*, which implements a pipeline of different dataset readers, preprocessing components, feature extractors, and machine learning algorithms on top of the *DKPro framework* (Eckart de Castilho & Gurevych, 2014).

The overarching success of various neural network architectures in natural language processing also led to their application in the automated assessment of constructed responses. Maharjan et al. (2018) and Uto and Uchida (2020) applied LSTM networks (Hochreiter & Schmidhuber, 1997) to the task. Gautam and Rus (2020) evaluated systems based on neural tensor networks, which incorporate structured data from knowledge graphs to improve predictions for the dataset from Maharjan et al. (2018). *Transformer language models* such as BERT (Devlin et al., 2019) were successfully applied to the task, too, and could be used to achieve the latest state-of-the-art results for the SemEval-2013 data (Camus & Filighera, 2020; Poulton & Eliens, 2021; Sung et al., 2019).

2.3 | Trusted learning analytics

Assessing students' knowledge through computational, data-driven methods can provide a basis for Learning Analytics (Greller & Drachsler, 2012; LAK, 2011). According to Greller and Drachsler (2012), Learning Analytics refers to varied data-driven research and engineering activities that focus on modeling, evaluating, and supporting human learning. This includes learner feedback and support systems that build upon the automated assessment of responses. It is important to state that, while predictive modeling through machine learning and natural language processing is an important part of the Learning Analytics toolbox (given that a lot of learner data comes in the form of text), the core aim of Learning Analytics is the support of learners and teachers. For this reason, respective systems should work in the best interest of them. Slade and Tait (2019) formulated concrete ethical guidelines for Learning Analytic work. According to these guidelines, "models used to analyze, interpret, and communicate learning analytics to stakeholders (support staff, advisers, faculties, students) should be sound, free from algorithmic bias; transparent where possible and clearly understood by the end-users." A related notion can be found in Drachsler and Greller (2016), who introduced the term Trusted Learning Analytics. While they primarily focus on issues related to data privacy, they also reflect upon the problem of asymmetrical power relationships found in learning scenarios. Predictive, data-driven methodology can replicate asymmetrical power relations if they are mirrored in data and has the potential that "[c]ertain patterns are made visible […] while other types are erased" (Birhane, 2021). Moreover, through distributional biases in training sets, models can learn unwanted shortcuts instead of accurate regularizations (Geirhos et al., 2020), a phenomenon sometimes also referred to as "clever Hans modeling" (e.g., Anders et al., 2022). Therefore, if such models are deployed as the basis of feedback- or scaffolding systems, it is of crucial importance to assess whether they function correctly and to guarantee that they do not negatively impact learners negatively through mispredictions.

2.4 | Explainable models

Glass-box algorithms enable us to gain insight into the inner workings of models to prevent such scenarios. Machine learning algorithms can be categorized into glass-box and black-box approaches, depending on how much respective insight they provide (e.g., Murdoch et al., 2019). Glass-box models can offer model-intrinsic explanations for their predictions (Søgaard, 2021). Typical examples of glass-box models are regression-based ones. That is, if input features are normalized to the same scale, regression coefficients can indicate the possible contribution of each feature to a respective outcome. Another example of interpretable models is tree-based ones. Here, practitioners can inspect and traverse individual learned trees to understand what features contribute to a given prediction.

However, a core problem in machine learning is that models with the highest predictive power, such as transformers (Vaswani et al., 2017), are often black-box models (Sun et al., 2021). It is hard to control if such models learn plausible regularities or unstable shortcuts (e.g., Geirhos et al., 2020). As black boxing prevents the usage of respective models in high-stakes scenarios where accountability is needed (Sun et al., 2021), there has been ongoing research on methods

for achieving transparency and interpretability so that the predictive power of respective methods becomes available in such scenarios, as well (Murdoch et al., 2019; Søgaard, 2021; Sun et al., 2021). According to Sun et al. (2021), methods for explaining the predictions of neural black box models can be grouped into three categories. These are training-based, test-based, and hybrid, with the latter referring to combinations of the prior two. Training-based methods are aimed at identifying training examples responsible for patterns found in the predictions, while test-based methods recognize which parts of an input are responsible for an according prediction outcome.

Bastings and Filippova (2020) argue in favor of saliency-based methods which can be categorized as test-based following the ontology by Sun et al. (2021). In their article, they further categorize such methods into gradient-, propagation- and occlusion-based. Gradient-based methods acquire model gradients through backpropagation to infer importance scores for sentences' words. Propagation-based methods such as layer-wise relevancy propagation (Binder et al., 2016) require custom backward passes to calculate relevancy scores over different neural network layers. These scores are then accumulated into final importance scores. Some methods combine gradient- and propagation-based methods. For example, *Transformer Explainability* (Chefer et al., 2021), a method aimed specifically at transformer language models that combines gradient-based weighting and scores acquired through layer-wise relevancy propagation, has been introduced in the context of transformer language models. Occlusion-based methods mask individual features or words (or, in certain use cases, groups of the same) and then monitor how this affects predictions. Occlusion can also be used to evaluate the reliability of other explainability methods. This is conducted through masking the parts of an input marked as important by a respective method and then measuring how this changes predictions.

As multiple methods for explaining models in different fashions have been developed, and notions of interpretability differ across the field, Murdoch et al. (2019) introduced the *PDR framework*, which is aimed at providing a general conceptual framework for approaching interpretable machine learning in a structured manner. *PDR* stands for *predictive accuracy*, *descriptive accuracy*, and *relevancy in this context*. *Relevancy* refers to the requirements of stakeholders, namely the aspects of descriptive and predictive accuracy about the models which are *relevant* to them, that is, how the influence of different features can be made visible, if explanations need to be model-intrinsic, or if the methods used need to be test- or training-based. *Predictive accuracy* refers to the well-established evaluation methods in machine learning that assess predictions' quality. *Descriptive accuracy* refers to whether what a model learns is plausible and faithful to the applied coding guidelines.

3 | RESEARCH QUESTIONS

The core aim of this study was to develop a method for automatically coding core ideas related to the manifestation and transformation of energy in students' constructed responses to derive knowledge networks from these codes. As NLP methodology for the automated assessment of constructed responses allows coding texts in an automated fashion, we wanted to assess how well the respective methodology worked for our use case. To comply with Slade and Tait (2019) and Drachsler and Greller (2016), the PDR framework (Murdoch et al., 2019) was used as an orientation for approaching this task by us. Following the framework, we first assessed was relevant for us in terms of predictive and descriptive accuracy.

For *predictive accuracy*, the relevant factor was that models should make correct predictions for as many test inputs as possible, that is, that the correct knowledge was coded within responses and that predictions for single responses translated well into accurate student knowledge networks. For *descriptive accuracy*, it was, in the broadest sense, relevant to know if the models we trained for this purpose picked up on plausible evidence matching what the human coders considered as such or if they learned undesired shortcuts (Geirhos et al., 2020). This can be conducted by inspecting model features. An intuitive way to test this is by using occlusion to examine how masking human-coded evidence affects predictions (Poulton & Eliens, 2021).

To summarize, we addressed the following research questions:

Research Question RQ1. : To what extent can core ideas from the energy physics domain be coded in students' free-text responses using NLP methodology for constructed response assessment?

Research Question RQ2. : What are the trade-offs between explainable feature-based approaches and transformer-based ones concerning predictive accuracy in this context?

Research Question RQ3. : What are the trade-offs between explainable feature-based approaches and transformer-based ones concerning descriptive accuracy in this context?

Research Question RQ4. : Do the features considered important by models for their predictions match human coding guidelines?

4 | METHOD

4.1 | Dataset

The data used in this study was collected from two approximately six 45 minutes class period long units on energy designed for middle school physics instruction. The units were designed following project-based pedagogy (Krajcik & Shin, 2014). Each unit starts with setting up a driving question on energy and related phenomena that motivates the following lessons (e.g., 'Why do laptops sometimes overheat?'). This driving question is then divided into three smaller sub-driving questions that students need to answer by engaging in numerous scientific practices, such as conducting investigations or constructing explanations. Finally, the units conclude by bringing together

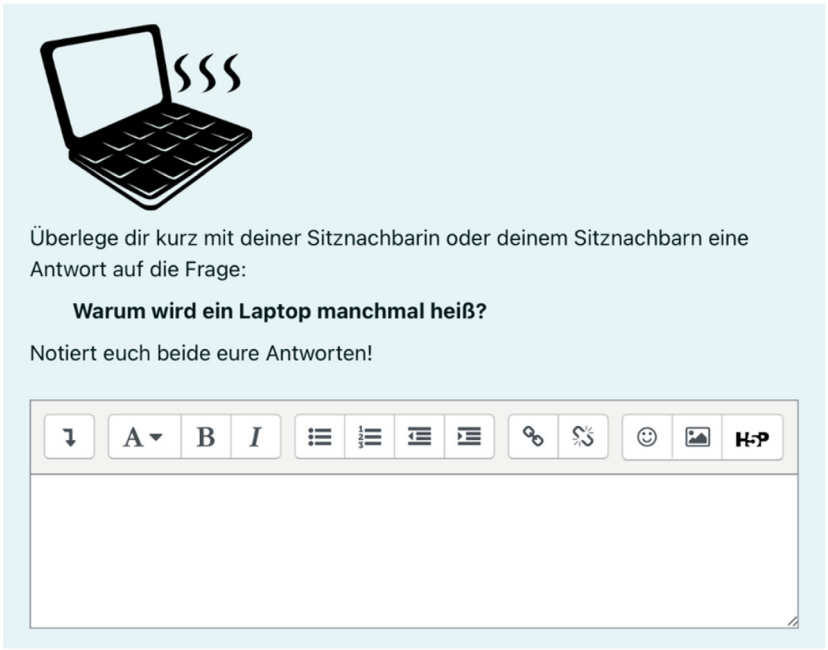

FIGURE 2 An open-ended question from the data set. In the task, students are prompted to discuss with a partner and then answer the question, "Why does a laptop heat up sometimes?".

the answers to the sub-driving questions to answer and reflect on the original driving question. The units were implemented in the learning management system Moodle (Dougiamas & Taylor, 2003), and all participating teachers received professional tutoring on Moodle and project-based pedagogy. They also were supported throughout the implementations of the units by the researchers. Figure 2 shows a typical open-ended task from one of the units.

We used a procedure grounded in evidence-centered design (ECD) (Mislevy et al., 2003; Mislevy & Haertel, 2007; Pellegrino et al., 2016) to develop rubrics for coding the core ideas within responses. It is based on existing research on students' learning about energy to formulate a competency model (e.g., Herrmann-Abell & DeBoer, 2018; Neumann et al., 2013). The core idea the units are meant to teach is that energy manifests itself in different forms, such as radiant or thermal energy and that each form can be observed through typical indicators. For example, a characteristic indicator of thermal energy is temperature. Moreover, the units also introduce the idea that energy can be transformed from one form into another; that is, the manifestations of energy change as phenomena unfold, for example, in the form of radiant energy converted into electric energy through a solar cell. Since energy degradation, dissipation, and conservation were not covered in the units, respective codes were not included in the rubric. Table 1 shows the complete list of codes we used in this work, grouped by aspect. It also provides criteria and examples for each of them.

In total, the data set we collected for this work comprises responses to 38 different constructed response items in German, for which we collected a total of 2835 responses from 305 students (see also Table 2). Each response was coded binarily for each idea indicating if it was present or not. Moreover, the coders also annotated the spans they saw as the corresponding evidence that a student knew and applied an idea correctly (e.g., if a student wrote about a light bulb that lights up after a switch was pressed, this could be considered as evidence for the student being aware of the idea that there exist indicators for electricity). We reference these parts of the responses as evidence spans.

Based on the data of 54 students (17.7% of the overall sample), interrater reliability was assessed using Cohen's Kappa (Landis & Koch, 1977) and was found to be within ranges with a minimum of 0.41 and an average of 0.96 across all tasks. In an iterative process, researchers and trained student workers first coded the responses independently, then checked for agreement, and finally resolved instances of disagreement through discussion. As some of the constructed response items addressed multiple ideas, responses could also be assigned multiple codes. Vice versa, we did not code all constructed response items for all ideas. Instead, we assigned all items only codes that were relevant for them.

Table 3 shows the corresponding distribution of *present*- and *not present*-cases for the different codes. It can illustrate that the present case is the minority case for all codes except *Thermal Indicator* and. *Radiant Indicator*. Table 4 shows the type-token rations of the human-coded evidence spans and the full responses for the different codes. This can reveal that the evidence spans for the codes corresponding to the manifestation of energy are limited to a small set of words, while the spans related to the indicator codes show a higher lexical diversity.

4.2 | System descriptions

As not all responses were coded with all codes from the rubric but rather only the ones that applied to the content of respective items, the problem at hand cannot be formulated as a regular multiclass classification problem but is rather a special case of a multi-labeling problem. In theory, responses can mention ideas that they were not explicitly coded for if students still brought them up.

TABLE 1 This rubric lists the different codes we applied to the data. Besides their names, the table lists corresponding descriptions and examples. Box brackets mark the human-coded evidence spans within the examples, the sections within the responses coders had used as evidence for positive coding decisions

Code	Criterion	Example
Electric Energy	A given response directly mentions the manifestation of electric energy or a synonym for the same.	Die Solarplatten erreichen die größte [elektrische Energie]. (The solar panels reach the highest electric energy).
Electric Indicator	A given response mentions at least one indicator for the manifestation of electric energy.	Man könnte überprüfen, ob man [einen Stromschlag bekommt]. (One could test if one gets an electric shock).
Thermal Energy	A given response directly mentions the manifestation of thermal energy or a synonym for the same.	[...] und das Thermometer zeigt eine Steigerung der [Wärmeenergie] an. ([...] and the thermometer shows an increase in thermal energy).
Thermal Indicator	A given response mentions at least one indicator for the manifestation of thermal energy.	Der Leiter [erhitzt sich]. (The conductor heats up).
Transformation Process	A given response mentions the transformation of energy from one form into another.	So kann an dem meisten Strahlungsenergie in elektrische Energie [umgewandelt werden]. (By this, the most radiant energy can be transformed into electric energy).
Radiant Energy	A given response directly mentions the manifestation of radiant energy or a synonym for the same.	Damit [die Energie von der Sonne] aufgefangen werden kann. (So that the energy of the sun can be collected).
Radiant Indicator	A given response mentions at least one indicator for the manifestation of radiant energy.	Weil dort am meisten [Sonnenlicht] hinkommt. (Because most sunlight reaches this point).

4.2.1 | Feature-based approaches

Therefore, for our feature-based models, we trained one distinct binary classification model per code for the feature-based classifiers. The subsets of responses coded for the involved idea were used for training and evaluating each respective model. With Explainable Boosting Machines, Random Forests, Ridge Regression, Logistic Regression, and Decision Trees, we tested five different explainable feature-based algorithms. These algorithms were chosen as they can all be explained through various methods for calculating feature importance scores. To our best knowledge, except for explainable boosting machines, all these algorithms were successfully applied for assessing constructed responses in past work. Except for Explainable Boosting Machines, for which we used an implementation provided by Nori et al. (2019), all models were implemented with Scikit-learn (Pedregosa et al., 2011) version 0.24.1.

While all other feature-based algorithms we applied are broadly known and part of the standard toolbox in machine learning, this is not the case for *Explainable boosting machines* (Lou et al., 2013). It is an ensemble method and an implementation of a *generalized additive model* as proposed by Hastie and Tibshirani (1987), a regression setup in which each of the different features is propagated through a corresponding scoring function aimed at modeling their contribution to a given final prediction outcome. As these are simply added up into a final score, the contribution of each feature to a prediction outcome can be explained ad hoc. Gradient-boosted regression tree ensembles (Mason et al., 2000) are used to fit the different scoring functions.

A criterion for all features we used to encode responses was that all of them should be easily interpretable by themselves, that is, it should be clear what information they encode so that respective feature importance scores would be informative of what the models had learned. For this reason, black box features such as raw latent word vectors were not used in the feature set. Instead, the following list of features was used:

Character n-*grams*

Ideas are, by large, signaled through certain words or combinations thereof, which function as respective evidence. For this reason, it is an intuitive choice to represent texts by the different terms they might contain. However, as the dataset consists of secondary school students' writing and, therefore, some responses have spelling mistakes, this approach cannot be considered robust and might fail to represent texts appropriately in such cases. Nonetheless, if a word contains spelling mistakes, it is likely that most of its constituting characters will still be correct. Therefore, when these words are encoded as lists of n-grams, there is a substantial overlap between misspelled and correctly spelled words. Thus, when encoding misspelled words through n-grams, they can still provide valuable signals. Therefore, we included character *n*-grams in our feature set with an *n* ranging from 1 to 5 for responses, sample solutions, and prompts. The n-grams were represented through *tf-idf* scores.

Word n-*grams*

In addition to character *n*-grams, we still included word *n*-grams using an *n* ranging from 1 to 3. We included these as character *n*-grams cannot appropriately model sequences of multiple words within a text, which might give helpful evidence on whether a specific idea is contained in a text, depending on the word order. Again, the *n*-grams were represented through *tf-idf* scores.

TABLE 2 This table lists the number of constructed response items, number of students, number of responses, and average number of responses per student

C. Response Items	Number of Students	Number of Responses	Avg. Number of Responses per Student	Avg. Number of Words per Response
38	305	2835	9.30	25.48

TABLE 3 The class distributions for the different codes within the data set

	Electric Energy	Electric Indicator	Thermal Energy	Thermal Indicator	Transformation Process	Radiant Energy	Radiant Indicator
Present	222	308	123	424	235	194	616
Not Present	904	629	347	169	891	653	231

TABLE 4 The type-token ratios of the human-coded evidence spans (parts of responses human coders marked as relevant for a given coding decision) and the full responses for the different codes

Corpus	Electric Energy	Electric Indicator	Thermal Energy	Thermal Indicator	Transformation Process	Radiant Energy	Radiant Indicator
Evidence Spans	0.121	0.176	0.152	0.162	0.095	0.195	0.101
Full Responses	0.134	0.135	0.168	0.120	0.131	0.127	0.086

N-grams of part-of-speech and dependency tags

We also included *n*-grams of part-of-speech and dependency tags. We used *Stanza* (Qi et al., 2020) to acquire these annotations automatically. The *Universal Dependencies* set of dependency relations[1] (Nivre et al., 2020) is used for dependency tags, while, for part-of-speech tags, the *Universal Dependencies* and *Stuttgart-Tübingen* tag sets[2,3] are used. The *n* was set to a range from 1 to 3. The *n*-grams were represented through *tf-idf* scores.

Similarity metrics

We included two different text similarity metrics with *Levenshtein* distance and cosine similarity. These were aimed at representing the superficial and semantic similarities between responses and the respective prompts and sample solutions. Cosine similarity is computed using German *fastText* word embeddings (Bojanowski et al., 2017) and word vectors generated through *latent semantic analysis* (Landauer et al., 1998). This is conducted on the levels of individual words as well as full responses. For the latter case, the centroid vectors of all individual word vectors from a response are used. We calculated the distances of all pairs of words between two texts for the word level. Following this, we selected the minimum, maximum, average, and median from the resulting scores and the range between minimum and maximum similarities as distinct features.

[1]https://universaldependencies.org/u/dep/
[2]https://universaldependencies.org/u/pos/
[3]https://homepage.ruhr-uni-bochum.de/stephen.berman/Korpuslinguistik/Tagsets-STTS.html

4.2.2 | Transformer-based multitask learning

Camus and Filighera (2020), Poulton and Eliens (2021), and Sung et al. (2019) used transformer language models to achieve state-of-the-art results for different SemEval-2013 data sets. For this reason, the application of transformer language models seemed like an approach worth testing for our purposes with regard to pure predictive accuracy. While transformers are black-box models, methods such as *Transformer Explainability* (Chefer et al., 2021) are promising to offer explanations that can be used to reveal shortcuts and gain insight into the models.

Transformer language models such as *BERT* (Devlin et al., 2019) are specialized *feed-forward neural networks* that process sequential data. They aim to solve a wide range of different natural language processing tasks and consist of so-called transformer encoders (Vaswani et al., 2017). These neural networks are based on what is called the self-attention mechanism. This neural network building block learns to represent words within a sequence as a learned weighted mean of the vectorial representations of their context words. By using multiple attention units, also called attention heads, models can learn to attend to different linguistic signals. Through this, transformer language models can produce contextual word embeddings. These vectorial representations of words encode their global distributional properties and respective local sentence contexts. Such representations can be used as input to neural networks for subsequent task-specific training and are especially useful in ambiguous cases where differences in contextual meaning can affect prediction outcomes.

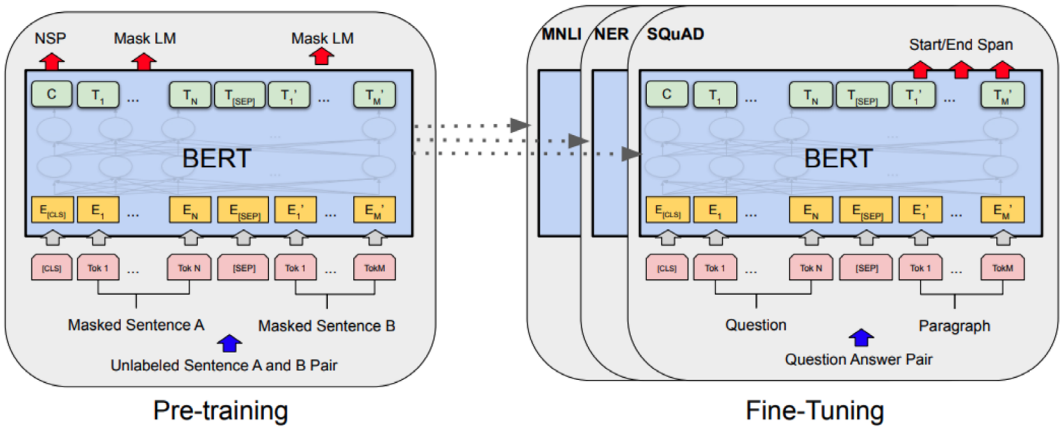

FIGURE 3 A schematic illustration of the training process of transformer language models taken from Devlin et al. (2019), licensed under Creative Commons Attribution 4.0. First, the model is trained with language modeling tasks such as masked language modeling and next sentence prediction in the pre-training stage. Afterwards, the model can be fine-tuned for different tasks such as question answering.

The training of transformer language models is divided into two distinct phases (see Figure 3). In the first one, a regular feed-forward layer is attached to the *transformer language model*. The resulting architecture is then trained with a language modeling task, usually *masked language modeling*. After this, the model can be used for downstream training. For this second phase, a smaller, task-specific neural network, in most cases a linear feed-forward layer, is attached to the *transformer model*. This layer is then passed the outputs of either the last or multiple intermediate layers of the pre-trained language model, and the overall network is trained to solve the downstream task in a supervised manner. As pre-training *transformer language models* is costly, while fine-tuning them is cheap, it has become standard practice not to carry out pre-training for each task but to rely on external pre-trained models. Especially the *Huggingface transformers* library (Wolf et al., 2020) and the accompanying model repository have become popular choices for this.

As not all responses were assigned codes for all ideas, the problem cannot be formulated as a regular multiclass or multi-label classification problem. However, training one distinct binary model per label as performed for the feature-based models becomes expensive quickly in terms of memory consumption for transformers. Consequently, we turned to the principle of multitask learning, used a single shared *transformer language model*, and initialized one binary linear classification layer per idea. The resulting output embeddings are then only fed to the binary layers corresponding to the codes relevant for a given item.

Moreover, we made a further adjustment to the model. As different attention heads in different layers of transformer language models tend to encode differing linguistic aspects, Liu et al. (2019) introduced the method of *scalar mixing* to let models make better use of these signals. Instead of just using the output of the last layer of a *transformer language model* (which is the default method as proposed by Devlin et al., 2019), a learned weighted mean of all layer outputs is calculated to allow the model to better use signals from the attention heads of intermediate layers. More specifically, the weights by which

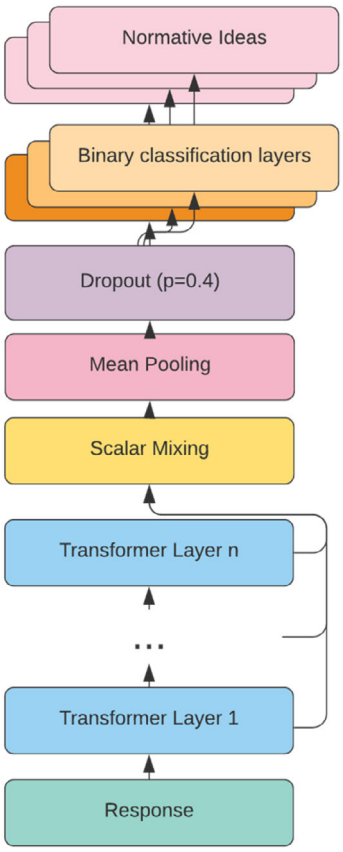

FIGURE 4 A visualization of the neural network architecture we use for our transformer-based models.

the different layer outputs are averaged are tuned during training. The resulting word vectors are then mean pooled to acquire a centroid vector representing the whole response. This pooling mechanism is inspired by findings from Reimers and Gurevych (2019), who found that mean pooling can help to improve predictive performance compared to the regular transformer pooling mechanism when

representing documents. Figure 4 shows the architecture we implemented schematically.

We used *GBERT* (Chan et al., 2020) as a basis for our models, a *BERT* variant pre-trained exclusively on German data. It comes in two different variants, *base* and *large*. The *large* model includes 24 transformer layers and consumes around double the memory compared to the *base* model with only 12 layers, but it also promises an increased performance. Therefore, we trained two systems based on each of the variants. The exact algorithm we used to train the transformer-based models can be formulated in the following way:

For each batch:

1. For each training example in a batch:
   a. Propagate the input response through the *transformer language model*.
   b. Acquire all word vectors for all transformer layers.
   c. Pass them through scalar mixing and mean pooling components to calculate a document embedding.
   d. For all codes, the example was coded for:
      i. First, pass the document embedding to the respective classification layer.
      ii. Then, calculate the individual loss and store it.
2. After calculations for all examples from the batch are finished: calculate the mean of all losses stored for the different samples.
3. Backpropagate the resulting mean loss to adjust the model parameters.
4. Delete the single losses stored for the items from this batch.

We used *AdamW* (Loshchilov & Hutter, 2019) as the optimization algorithm and a cosine-based learning rate scheduler with a continually decreasing learning rate. After conducting a hyperparameter search, we trained the models for 12 epochs using a learning rate of 2e-5, a weight decay of 1e-2, and a micro-batch size of 4. During hyperparameter search, we found that, after the 12th epoch, models started to overfit too much, and results started to decrease.

4.2.3 | Baselines

In addition to our machine learning-based models, we also implemented two keyword-based baselines. These function through keyword lexica, against which the lemmatized tokens within input responses are matched (like the early works discussed in the second paragraph of section 2.2). Per code, one lexicon is defined. A respective positive label is assigned to a response containing at least one of the keywords from a lexicon. For the first of these baselines, we assembled respective lexica by hand. We collected different nouns, verbs, adjectives, and adverbs present within human-coded evidence spans from a subset of the data set and selected appropriate words from this set. For the second one of these baselines, the keyword lexica were assembled automatically from the words within a response. This was achieved by ranking them according to their odds ratios using a foreground corpus of positive examples and a background corpus of all positive and negative examples from the training set combined to acquire the keywords which were most distinctive for the positive examples compared to the overall dataset.

5 | EVALUATION

5.1 | Predictive accuracy

For evaluating predictive accuracy, we trained and evaluated all different models in a 5×5 cross-validation setup which was also used for hyperparameter search. The same folds were used for all models to ensure that the results were perfectly comparable. Moreover, we did not group the folds by single data points but by different students. This is conducted as the systems are intended to classify the data of unseen students in the future, which can be tested best through this setup. We relied on *F1*, the harmonic mean between precision and recall, as the main evaluation metric to acquire separate scores for the positive and negative cases. Table 5 shows the respective results.

What becomes visible after observing these scores is that most machine-learning-based models show satisfactory performances for most labels. Out of all the different models, the one based on *GBERT-large* achieved the highest *F1 scores* for both cases and can, for this reason, be considered the best out of all models concerning *predictive accuracy*. However, the different feature-based models do not rank far below, with explainable boosting machines and random forests performing the best. The keywords-based baselines perform better than the random baseline but significantly worse than the machine learning models.

For the next evaluation step, we used the models to construct knowledge networks like Kubsch et al. (2019). This was conducted to observe how well codes for individual responses would translate into such networks. Nodes were used to represent the different ideas, while edge weights were used to indicate the number of co-occurrences of these ideas within students' responses. For the evaluation, we instantiated two networks per student. The first was constructed from model predictions, that is, if two ideas co-occurred in a response, the respective edge weight was increased by one. The second network was a gold standard one. These networks include all the ideas a student could have theoretically mentioned in the responses he gave, ergo all codes which were possible for these responses. Figure 5 shows an example of a student network from our data set.

We then measured Euclidean distances between the adjacency matrices of the predicted and gold standard networks to assess their similarity. A smaller Euclidean distance indicates a higher similarity, while a larger distance indicates a lower similarity. Moreover, we measured the percentage of cases where a predicted network completely matched a gold standard one.

As Table 6 shows, the resulting ranking matches the one from the F1-based evaluations. Models which reached higher F1 scores also achieved smaller mean Euclidean distances between predicted and gold standard networks, which is an expectable result. Moreover, the

TABLE 5 The results of the different approaches achieved for the corresponding codes. This table depicts F1 scores

	Electric Energy	Electric Indicator	Thermal Energy	Thermal Indicator	Transformation Process	Radiant Energy	Radiant Indicator	Macro. F1
Present								
Multitask-GBERT-large	**93.29**	**89.29**	90.53	**90.36**	91.89	**91.07**	**90.46**	**90.98**
Multitask-GBERT-base	90.76	86.69	**91.07**	89.50	**92.22**	89.99	89.79	90.00
Random Forests	92.31	80.81	85.37	87.55	86.58	87.19	88.80	86.94
Explainable Boosting Machine	90.68	78.78	83.60	87.84	86.67	86.33	89.14	86.15
Logistic Regression	87.44	78.91	83.58	85.92	81.14	84.30	88.60	84.27
Ridge Regression	89.91	78.25	82.10	83.13	82.15	87.11	86.00	84.09
Decision Tree	87.72	79.18	76.80	82.14	78.89	78.72	83.26	80.96
Keywords (Evidence Spans)	60.17	62.59	73.19	85.50	46.99	61.19	86.86	68.07
Keywords (Odds Ratio)	61.90	18.77	76.89	81.21	61.78	49.49	80.53	61.51
Random	*19.72*	*32.87*	*26.17*	*71.50*	*20.87*	*22.90*	*72.73*	*38.11*
Not Present								
Multitask-GBERT-large	**98.3**	**94.68**	96.86	**75.69**	97.80	**97.20**	**70.79**	**90.19**
Multitask-GBERT-base	97.63	93.44	**96.88**	72.23	**97.86**	96.90	66.17	88.73
Explainable Boosting Machine	97.70	90.93	94.49	68.22	96.61	96.05	64.70	86.96
Random Forests	98.09	92.02	95.00	64.57	96.61	96.29	62.14	86.38
Logistic Regression	97.06	90.65	94.45	66.24	95.32	95.56	66.91	86.59
Ridge Regression	97.56	89.76	93.74	63.73	95.31	96.18	60.90	85.31
Decision Tree	97.00	89.79	91.74	62.44	94.43	93.72	54.06	83.31
Keywords (Evidence Spans)	80.73	68.70	86.70	48.98	58.45	76.40	35.50	65.07
Keywords (Odds Ratio)	83.25	71.42	90.12	43.03	81.22	58.97	36.92	66.42
Random	*80.28*	*67.13*	*73.38*	*28.50*	*79.13*	*77.10*	*27.27*	*61.82*

Note: Bold scores are the best achieved in the respective category.

models achieving the highest F1 scores also achieve the highest percentage of complete matches between gold-standard and predicted networks. However, high F1 scores calculated for single responses did not translate well into a high rate of full matches. While the macro F1 score for the best model based on GBERT-large is 90.98, only 54.14% of all networks could be fully reconstructed by it. The mean Euclidean distance of 0.821 indicates that most network differences seem to be minor, but it is still a limitation that needs to be considered. Figure 6 shows the distributions of Euclidean distances between predicted and gold standard networks for the different models.

It can reveal that most predicted networks that are different from gold standard ones differ from these by Euclidean distances between mostly 1 and 3. For the decision tree- and regression-based models, higher distances up to 4 are also common. The figure can also reveal that the networks predicted by the BERT-based models come closest to the gold standard ones.

5.1.1 | Secondary evaluation using SemEval-2013 Task 7 data

The dataset we focused on in this article is an entirely novel one. This makes it hard to relate our results to past work in the field. To enable respective comparisons, we also evaluated the best-performing model architecture using the *SciEntsBank 3-way* dataset from *SemEval-2013 Task 7* (Dzikovska et al., 2013) as a reference point. This dataset consists of 10,000 student responses from 197 different science-related open-ended tasks at the university level. The evaluation set is divided into three subsets: unseen responses (responses to open-ended tasks which were also used for training), unseen questions (responses to open-ended tasks that were not seen during training but are from similar domains as the ones encountered during training), and unseen domains (responses to open-ended tasks which were not seen during training and stem from different domains). Moreover, for each open-

ended task, a sample solution is provided. Each response was coded with one out of these three labels:

- Correct: a response is correct and matches a given sample solution.
- Incorrect: a response is incorrect and does not match a given sample solution.
- Contradictory: a response is contradictory with respect to the provided sample solution.

This dataset is in English, so we used the English transformer language model *RoBERTa-large-MNLI* (Liu et al., 2019) as the basis for our model. This model was also fine-tuned by Camus and Filighera (2020) to achieve (to our best knowledge) state-of-the-art results for this data set and is equal in size and architecture to GBERT-large, which means that all adjustments we made to the GBERT models should translate. However, as multitask learning did not apply to this data set, we used a single classification layer and not multiple ones. Nevertheless, scalar mixing, mean pooling, and dropout were used similarly for the model. Table 7 shows the corresponding evaluation results:

Especially for the unseen questions subset, our model could achieve significant improvements over past approaches while performing only worse in the unseen responses category. This can demonstrate that our adjustments to the transformer-based models were reasonable. Incorporating scalar mixing and mean pooling into the models could help them achieve (to our best knowledge) state-of-the-art results for the unseen questions- and unseen domains subsets of *SciEntsBank 3-way* and thus seem to be an improvement over the standard transformer architectures used by Camus and Filighera (2020) and Sung et al. (2019). In particular, these results suggest that our adjustments could improve the cross-domain transfer capabilities of transformer language models. However, this claim would need to be further evaluated using more specific datasets, which is out of the scope of this work.

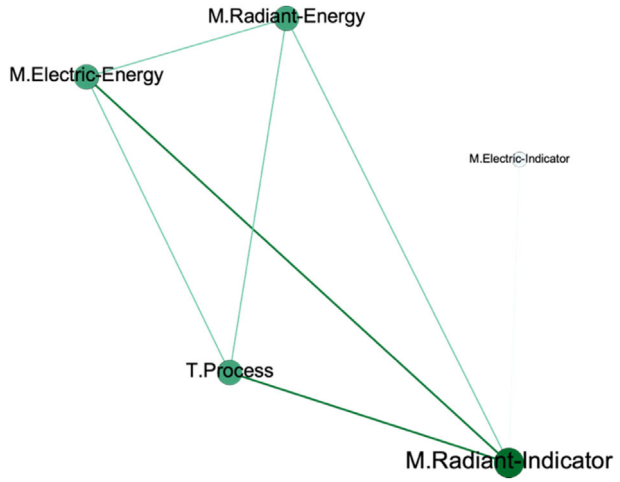

FIGURE 5 An example network predicted by one of the decision tree-based models. The nodes represent different ideas detected in the constructed responses of a particular student, the edges how often these co-occurred. The size and color of the nodes indicate their degree (the darker and smaller the node/edge, the lower the degree/the number of co-occurrences).

5.2 | Descriptive accuracy

5.2.1 | Feature importance and learned shortcuts

In the next steps, we addressed the descriptive accuracy of our models. To analyze the contribution of the various features for the different models, we computed their importance scores for each code. This was conducted in the same 5×5 cross-validation setup as used during the previous evaluation steps. Importance scores were then averaged over all folds. For the regression-based models, the scores were simply given through the respective coefficients, while for the explainable boosting machines, these were determined through the different scoring functions and their possible contributions. For random forests, the mean decrease in impurity and, for the decision trees, Gini importance were used to acquire feature importance scores. The transformer-based models are omitted as they do not operate on the same features as the other models. The scores were normalized. Figure 7 shows the distribution of feature importance scores for the different feature categories. Individual feature importance scores were collected for all folds and codes to get a general overview of the features used by the models.

The figure can reveal that the models make use of the features differently. For example, for the decision tree-based model, only a limited set of character and word *n*-gram features plays a role, while the importance of all other features is relatively minor, while for the regression and ensemble-based models more features seem to be of importance. Character and word n-grams from the responses were of comparably high importance for all models, while the ones from prompts and sample solutions are the least important features across all models. A pattern that is observable across nearly all feature categories across all models is that successful prediction outcomes seem to depend mainly on a smaller number of essential features, given that the importance of many features is close to 0 across all models. After taking a close look at the data, we can state that,

TABLE 6 This table shows the mean Euclidean distance, max Euclidean distance, and the percentage of complete matches between predicted and gold standard networks

Model	Mean Euclidean	Max Euclidean	Percentage of complete Matches
Multitask-GBERT-large	**0.821**	**7.874**	**54.14**
Multitask-GBERT-base	0.836	**7.874**	53.95
Random Forests	0.960	**7.874**	50.86
Explainable Boosting Machine	0.992	8.246	49.67
Logistic Regression	1.086	**7.874**	45.65
Ridge Regression	1.134	9.434	42.89
Decision Tree	1.326	8.660	36.77

Note: Bold refers to the best result within a category.

for the most part, the important features stem from expectable words, for example, the word *Spannung* (voltage) as evidence for *Electric Indicator*, or *umwandeln* (transform) for *Transformation Process*, although words beyond these also play a role. Part-of-speech and dependency tags are important for the regression models and explainable boosting machines but less so for decision trees and random forests. The similarity score-based features are of higher importance for the random forests and explainable boosting machine models than for the others. Out of these scores, especially the *fastText*-based ones seemed to be informative for the models. In general, it can be stated that features from different categories contain signals which turned out useful for positive classification outcomes.

In the next step, we manually inspected the most important features by hand to identify if the models had learned undesired shortcuts (Geirhos et al., 2020). For feature-based models, we could simply inspect the feature importance lists and then evaluate what words certain important features stemmed from and if they were coherent with the coding guidelines, while, for the transformer-based models, feature importance needed to be assessed through methods for generating post-hoc explanations. We used the technique of *Transformer Explainability* introduced by Chefer et al. (2021), which builds upon a specialized variant of layer-wise relevancy propagation (Binder et al., 2016) and gradient-based methods adapted to transformers to predict importance scores for individual words in test sentences. By assembling these importance scores and their distribution for the different words within the whole data set, it is possible to get an overview of important terms for positive classification outcomes in most contexts. We ran this method in 5×5 cross-validation to inspect what words the models considered important over multiple different folds. From these runs, we calculated the overall distribution of importance for the different words encountered in the data set. We then ranked these words to acquire lists of the most important words for each idea. Figure 8 illustrates exemplarily how word importance can be distributed over a single input response for transformer models.

Inspecting the most important features for all different models, we could then reveal a small set of impactful shortcuts all of them had picked up. Table 8 shows words that we identified as these undesired shortcuts within respective categories (Geirhos et al., 2020).

All the words listed were among the top 20 most important features for the respective codes for both feature-based and transformer-based models. Especially for *Radiant Indicator* and *Energy*, *Electric Energy* and *Transformation Process*, these were similar words, namely the respective energy forms and the word *umwandeln* (transform). As the items from the dataset deal with energy transformation, it is expectable from a statistical point of view that words referring to different forms of energy co-occur in the responses, but the fact that these co-occurrences result in shortcuts is still undesirable.

TABLE 7 Weighted F1 scores for the SciEntsBank 3-way dataset. Unseen responses refer to responses to open-ended tasks contained in the training data, unseen questions to responses whose open-ended tasks were not encountered, and unseen domains to responses where the open-ended tasks were not encountered during training and stem from different domains

Model	Unseen Responses	Unseen Questions	Unseen Domains
RoBERTa-large-MNLI + Scalar Mixing + Mean Pooling + Dropout (ours)	77.2	73.8	73.2
Camus and Filighera (2020)	78.8	66.4	71.8
Sung et al. (2019)	75.8	64.8	63.4
Saha et al. (2018)	71.4	62.8	61.2

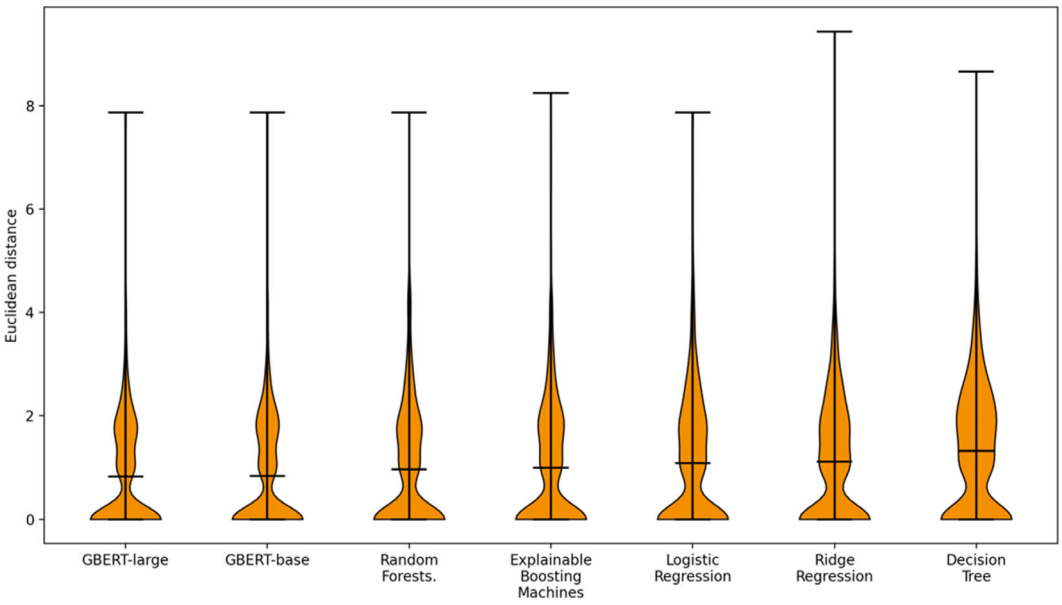

FIGURE 6 This figure illustrates the distributions of Euclidean distances between predicted and gold standard networks. The upper bar marks the maximum Euclidean distance, the middle bar is the mean, and the lower bar is the minimum.

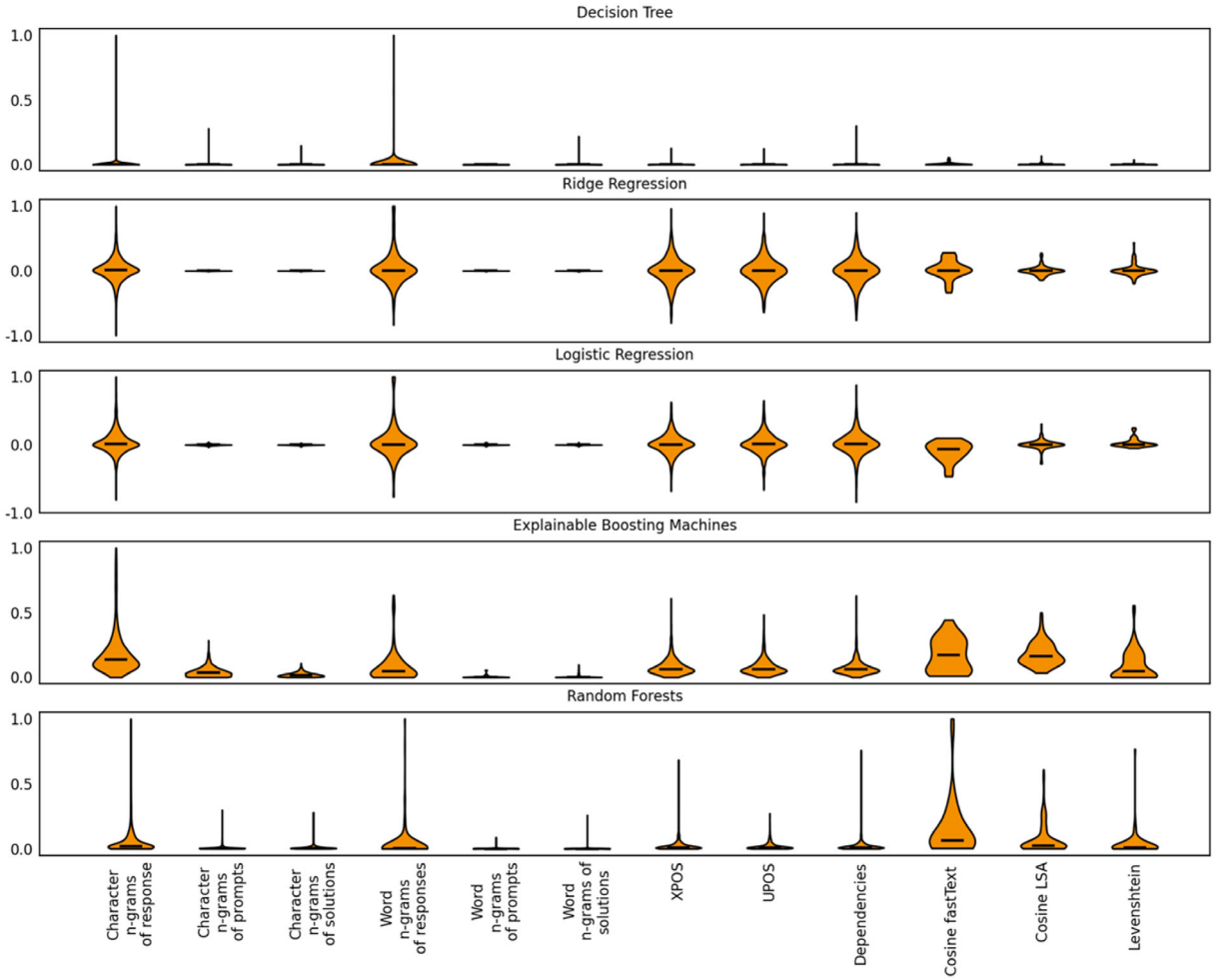

FIGURE 7 This figure shows violin plots visualizing the distribution of feature importance for the different models and feature categories. Ranges are set according to the ranges of feature importance the different models produce, ergo to a range of 0.0 to 1.0 for the tree- and ensemble models, −1.0 to 1.0 for the regression-based models.

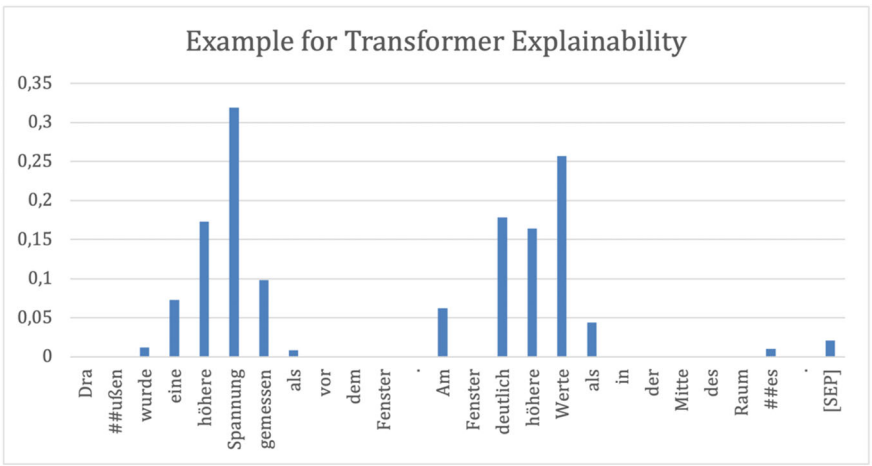

FIGURE 8 Importance scores of the different tokens of an example sentence taken from the data set for the idea Electric Indicator. For better visibility of the model's important words, we subtracted the mean importance from all individual scores and set all resulting scores below zero to zero. (Translation: Outside, a higher voltage was measured than in front of the window. At the window, much higher values than in the middle of the room).

5.2.2 | Occlusion analysis

We used the human-coded evidence spans to analyze if models carried out their predictions for similar reasons as humans. For this purpose, we occluded the data in two ways. On the one hand, we generated versions of the dataset with the human-coded evidence spans masked within the responses for each label. On the other hand, we generated versions where all parts of the answers except these

spans were masked. If models trained on the regular data set match human coders and are forwarded the prior, their predictive accuracy should drastically decrease, while, for the latter, it should stay relatively equal and only get decreased to minor degrees. Therefore, we measured the decrease in true positives compared to the regular inputs for both cases to assess how much positive classification outcomes depended on the human-coded evidence. We carried out respective analyzes for all the models in a 5×5 cross-validation setup. Table 9 shows the individual results:

This revealed that masking the evidence spans affected models more than masking everything except them, demonstrating that they seem to be overall important for the models. The decision tree- and transformer-based models were the most affected by masking the evidence spans, while the explainable boosting machine models were affected the least, which implies that, for these models, the evidence spans were the most important. The models that could deal best with masking everything except the evidence were the regression- and transformer-based ones. It seems as if the transformer-based models seem to be the overall truest to human coding, although this differs from idea to idea. On the other hand, the Explainable Boosting Machines differed the most from human coders. However, the percentage of true positives turned into false negatives varies tremendously

TABLE 8 Words that we identified as shortcuts the models had learned for the different codes. These had high feature importance for all models, and usually, the presence of one of these words led to a positive classification outcome for the respective class, even though these words should not be responsible for the latter

	Electric energy	Electric indicator	Thermal energy	Thermal indicator	Transformation process	Radiant energy	Radiant indicator
Shortcut words	Strahlungsenergie (radiant energy) umgewandelt (transformed)	—	Strahlungsenergie (radiant energy)	—	Elektrische energie (electric energy)	Elektrische energie (electric energy)	Elektrische energie (electric energy) umgewandelt (transformed)

TABLE 9 The percentages of true positives turned into false negatives by masking human-coded evidence and everything except for it per model and code. The upper section of the table shows the prior, while the lower section shows the latter

Percentage of true positives which turned into false negatives through masking evidence spans								
Model	**Electric energy**	**Electric indicator**	**Thermal energy**	**Thermal indicator**	**Transformation process**	**Radiant energy**	**Radiant indicator**	**Mean**
Decision Tree	80.09	**61.92**	77.47	44.21	81.85	**70.19**	39.78	**65.07**
Multitask-GBERT-large	68.79	52.17	67.37	**47.82**	84.71	66.41	**67.73**	65.00
Multitask-GBERT-base	62.85	47.27	66.73	40.02	85.37	68.48	66.94	62.52
Logistic Regression	80.83	46.57	71.71	27.16	86.07	65.04	46.97	60.62
Ridge Regression	72.70	49.44	66.43	33.48	81.35	64.94	41.69	58.58
Random Forests	**88.34**	56.53	**80.49**	14.91	**88.03**	45.60	34.18	58.29
Explainable Boosting Machines	80.49	47.30	78.85	25.83	86.37	44.33	32.23	56.49
Percentage of true positives which turned into false negatives through masking everything except the evidence spans								
Model	**Electric energy**	**Electric indicator**	**Thermal energy**	**Thermal indicator**	**Transformation process**	**Radiant energy**	**Radiant indicator**	**Mean**
Ridge Regression	13.56	25.67	**15.07**	28.01	**6.85**	**10.53**	13.23	**16.13**
Logistic Regression	**11.61**	26.29	20.65	**27.08**	11.13	10.67	9.22	16.67
Multitask-GBERT-large	16.47	22.97	30.41	27.27	8,48	14.61	**3.19**	17.63
Multitask-GBERT-base	17.63	24.63	26.36	32.47	10.04	15.90	3.77	18.69
Decision Tree	15.87	**18.44**	39.84	29.31	19.80	22.52	15.95	23.10
Random Forests	13.38	26.95	32.46	55.20	28.12	20.29	4.81	25.88
Explainable Boosting Machines	17.24	39.51	28.67	44.15	23.50	22.27	6.64	26.00

for the different codes and models, and, in general, only up to roughly two-thirds of the true positives were rendered false negatives by masking the evidence spans.

For this reason, responses seem to contain signals ranging beyond the human-coded evidence, which can lead to successful prediction outcomes. This could be caused by features that are likely to co-occur with respective evidence. Following the distributional-semantic notion that similar words appear in similar contexts (Firth, 1957), it is expectable that contexts of evidence spans are similar to each other and, therefore, contain similar signals across responses, which the models can then use. On the other hand, this could also result from less apparent shortcuts within the models that we did not manage to detect. The regression- and transformer-based models seem to be the overall least affected by these signals, as indicated by the reduction in true positives for masking everything except the evidence spans.

6 | DISCUSSION

We assessed to which degree techniques used for automatic constructed response assessment can be applied to automate the coding of core ideas from the domain of energy physics in constructed responses. We implemented multiple systems based on transformer language models and five different feature-based classification algorithms. For the latter, we built a shared feature set inspired by past work on automatic constructed response assessment exclusively from interpretable features. All models achieved high macro F1 scores (90.98 resp. 90.19 for the best transformer-based variant; 86.15 resp. 86.96 for the best feature-based approaches) and overall high F1 scores for all codes, indicating that they can indeed be applied for automating the coding process. Moreover, we implemented two keyword-based baselines, which performed better than random but worse than the machine learning models. We also evaluated to which degree co-occurrence-based knowledge networks generated for students from the co-occurrence of codes in their responses matched networks derived from a human-coded gold standard. As expectable from the results of the previous evaluation step, the transformer-based models also turned out as the most appropriate models for generating respective networks as they came closest to the gold standard networks. However, only 54.14% of the networks generated were a complete match. Therefore, in response to RQ1 (the question of whether it is feasible to code core ideas of energy physics in constructed responses) and RQ2 (the question of what trade-off between transformer-based models and explainable feature-based models is to be made concerning predictive accuracy in this context), it can be stated that the methods we applied were overall successful in detecting the ideas within individual responses. However, there is room for further improvement in how well this translates into knowledge networks, and it is necessary to explore further methodology for constructing the latter. Concerning RQ2, it can be stated that the transformer-based models achieved an overall superior predictive performance. The approach used by us also seems to be applicable to other data, as we could use a similar model to achieve state-of-the-art results in two out of three evaluation categories from the *SciEntsBank 3-Way* dataset. However, as transformers are, by far, more heavyweight in terms of pure resource consumption than feature-based models, it is debatable if they are needed in all cases (Bender et al., 2021). Of course, also the achieved F1 scores leave room for some future improvements.

We then analyzed the models for their descriptive accuracies to address RQ3 (the question of the trade-offs between transformer-based and explainable feature-based models) and RQ4 (whether the models consider similar signals as important as human coders). First, we inspected the distribution of feature importance for the different feature-based models. This could reveal that a limited set of essential features were mainly responsible for positive classification outcomes for most models. Most of the features were of comparably minor importance. However, the important features were still distributed differently for different feature categories and models. While the character and word *n*-gram features were important for all models, the distance metrics were only of higher relevance to the ensemble models.

We also inspected the feature importance scores to reveal undesired shortcuts (Geirhos et al., 2020) among the models' learned features. For the black box transformer-based approaches, we could successfully apply the method introduced by Chefer et al. (2021) to glass box these models and acquire importance scores for individual words within responses. Our analysis could reveal that all models had picked up one major variant of shortcuts where words referring to certain energy forms were undesirably important for predicting codes corresponding to other energy forms. This issue was present for both feature- and transformer-based models and stemmed from the issue that respective words co-occurred with important evidence words in individual responses. Additional data that balances out these co-occurrences would be needed to fix this issue. Concerning RQ3, it can be stated that shortcut learning affected both transformers and feature-based models.

Following this, we inspected to what extent the models considered similar evidence as important as human coders to address RQ4. For this purpose, we created two perturbated versions of the data set. In the first of these versions, we masked human-coded evidence in the responses and everything except it in the second version. We then measured the percentage of true positives turned into false negatives through these perturbations for both datasets. Overall, masking human-coded evidence impacted results stronger than masking everything except it, which shows that the models picked up on overall plausible signals. The decision tree- and transformer-based models were affected by masking human evidence the most, indicating that they rely on similar signals as humans for their predictions. On the other hand, the regression-based models were the least affected by masking everything around human-coded evidence. For different ideas, the human-coded evidence spans seem to be of varying importance for the models, and there seem to be features beyond these spans, which also affect prediction outcomes, but nonetheless, the models consider evidence also considered important by humans as more important than other parts of the responses.

It is important to remark that education is highly contextual. For this reason, the performance of our models might not translate well to use cases beyond the ones intended by us. It is unlikely that the models trained by us can be used for data from other assessments without retraining them with appropriate data. Nonetheless, our results confirm the findings from Sung et al. (2019) and Camus and Filighera (2020) that transformers seem to be a promising choice for implementing response coding systems. These results are also in line with general developments in NLP, where transformers could be used to achieve significant progress for a wide range of problems (e.g., Rogers et al., 2020). Moreover, as the changes we made to the architecture of our transformers compared to the default setup led to superior evaluation results on the dataset they used, it can be stated that these were reasonable.

7 | CONCLUSION

In this article, we approached the automated coding of seven ideas related to energy manifestation and transformation in the constructed responses of 305 K-12 students to a total of 38 constructed response items. We solved this task by applying machine learning methodology for automated constructed response assessment. Our approaches yielded fruitful outcomes, although there are some limitations, especially regarding the successful translation of single codes into accurate knowledge networks. Furthermore, to comply with Slade and Tait (2019) and the demands of Trusted Learning Analytics (Drachsler & Greller, 2016), we also addressed the topic of model explainability. We found out all approaches picked up on mainly plausible features, but unfortunately, the models also picked up on some undesired shortcuts. Furthermore, the features deemed important by our models generally matched with human-coded evidence, although the responses seem to contain features beyond these which also impact classification results.

8 | FUTURE WORK

Our next steps will involve applying the demonstrated methods to new, unseen learner data to study the development of the corresponding knowledge networks. Comparing such networks to network representations of the knowledge structure contained in instructional materials (Christianson et al., 2020) could provide a basis for efficiently identifying gaps in students' knowledge, giving feedback, and planning future instruction. Besides this, future work on related coding systems could target implementing systems that are not solely trained for a pre-defined set of fixed codes but approach the subject more broadly. For example, more generalizable architectures that learn to predict if an input text entails unseen input ideas are imaginable. This would allow studying students' knowledge networks and how they develop without coding datasets for specific didactic areas. However, for creating such models, one likely needs to train them on larger, more diverse datasets that address ideas from many different areas to guarantee a successful regularization.

ACKNOWLEDGMENT

Open Access funding enabled and organized by Projekt DEAL. [Correction added on 10 February 2023, after first online publication: <Open Access funding enabled and organized by Projekt DEAL.> Funding statement has been added.]

PEER REVIEW

The peer review history for this article is available at https://publons.com/publon/10.1111/jcal.12767.

DATA AVAILABILITY STATEMENT

The authors of the paper will hand out the data on request for non-commercial, scientific purposes such as reproduction studies. The usage of the data for any commercial purposes is strictly forbidden. Moreover, any attempts at de-anonymizing the data are strictly forbidden, as well.

ORCID

Sebastian Gombert https://orcid.org/0000-0001-5598-9547
Daniele Di Mitri https://orcid.org/0000-0002-9331-6893
Onur Karademir https://orcid.org/0000-0002-6985-4202
Marcus Kubsch https://orcid.org/0000-0001-5497-8336
Adrian Grimm https://orcid.org/0000-0003-2701-3349
Isabell Bohm https://orcid.org/0000-0003-1646-5163
Knut Neumann https://orcid.org/0000-0002-4391-7308
Hendrik Drachsler https://orcid.org/0000-0001-8407-5314

REFERENCES

Anders, C. J., Weber, L., Neumann, D., Samek, W., Müller, K.-R., & Lapuschkin, S. (2022). Finding and removing Clever Hans: Using explanation methods to debug and improve deep models. *Information Fusion*, *77*, 261–295. https://doi.org/10.1016/j.inffus.2021.07.015

Andersen, N., & Zehner, F. (2021). shinyReCoR: A Shiny Application for Automatically Coding Text Responses Using R. *Psych*, *3*(3), 422–446. https://doi.org/10.3390/psych3030030

Anderson, J. R. (2013). *The Architecture of Cognition*. Psychology Press. https://doi.org/10.4324/9781315799438

Bachman, L. F., Carr, N., Kamei, G., Kim, M., Pan, M. J., Salvador, C., & Sawaki, Y. (2002). A Reliable Approach to Automatic Assessment of Short Answer Free Responses. Proceedings of the 19th International Conference on Computational Linguistics, COLING 2002, Howard International House and Academia Sinica, Taipei, Taiwan, August 24 - September 1, 2002. https://aclanthology.org/C02-2023

Bastings, J., & Filippova, K. (2020). The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, 149–155. https://doi.org/10.18653/v1/2020.blackboxnlp-1.14

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 610–623. https://doi.org/10.1145/3442188.3445922

Binder, A., Bach, S., Montavon, G., Müller, K.-R., & Samek, W. (2016). Layer-Wise Relevance Propagation for Deep Neural Network Architectures. In K. J. Kim & N. Joukov (Eds.), *Information Science and Applications (ICISA) 2016* (pp. 913–922). Springer Singapore.

Birhane, A. (2021). Algorithmic injustice: A relational ethics approach. *Patterns*, *2*(2), 100205. https://doi.org/10.1016/j.patter.2021.100205

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, *5*, 135–146. https://doi.org/10.1162/tacl_a_00051

Bransford, J. D. (2000). *How People Learn: Brain, Mind, Experience, and School* (Expanded ed., 9853). National Academies Press. https://doi.org/10.17226/9853

Burrows, S., Gurevych, I., & Stein, B. (2015). The Eras and Trends of Automatic Short Answer Grading. *International Journal of Artificial Intelligence in Education*, *25*(1), 60–117. https://doi.org/10.1007/s40593-014-0026-8

Burstein, J., Wolff, S., & Lu, C. (1999). Using lexical semantic techniques to classify free-responses. In *Breadth and depth of semantic lexicons* (pp. 227–244). Springer.

Callear, D. H., Jerrams-Smith, J., & Soh, V. (2001). *CAA of short non-MCQ answers*. Loughborough University.

Camus, L., & Filighera, A. (2020). Investigating Transformers for Automatic Short Answer Grading. In I. I. Bittencourt, M. Cukurova, K. Muldner, R. Luckin, & E. Millán (Eds.), *Artificial Intelligence in Education* (Vol. 12164, pp. 43–48). Springer International Publishing. https://doi.org/10.1007/978-3-030-52240-7_8

Chan, B., Schweter, S., & Möller, T. (2020). German's Next Language Model. Proceedings of the 28th International Conference on Computational Linguistics, 6788–6796. https://doi.org/10.18653/v1/2020.coling-main.598

Chefer, H., Gur, S., & Wolf, L. (2021). Transformer interpretability beyond attention visualization. IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, Virtual, June 19-25, 2021, 782–791. https://doi.org/10.1109/CVPR46437.2021.00084

Christianson, N. H., Sizemore Blevins, A., & Bassett, D. S. (2020). Architecture and evolution of semantic networks in mathematics texts. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *476*(2239), 20190741. https://doi.org/10.1098/rspa.2019.0741

Crossley, S. A., Kyle, K., Davenport, J. L., & McNamara, D. S. (2016). Automatic Assessment of Constructed Response Data in a Chemistry Tutor, Proceedings of the 9th International Conference on Educational Data Mining, EDM 2016, Raleigh, North Carolina, USA. International Educational Data Mining Society (IEDMS) (pp. 336–340). http://www.educationaldatamining.org/EDM2016/proceedings/paper_43.pdf

Cutrone, L. A., & Chang, M. (2010). Automarking: Automatic Assessment of Open Questions. 2010 10th IEEE International Conference on Advanced Learning Technologies, 143–147. https://doi.org/10.1109/ICALT.2010.47

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 North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 4171–4186. https://doi.org/10.18653/v1/N19-1423

Disessa, A. A. (1988). Knowledge in pieces. In *Constructivism in the computer age* (pp. 49–70). Lawrence Erlbaum Associates, Inc.

Dougiamas, M., & Taylor, P. (2003). Moodle: Using Learning Communities to Create an Open Source Course Management System. In D. Lassner & C. McNaught (Eds.), *Proceedings of EdMedia + Innovate Learning 2003* (pp. 171–178). Association for the Advancement of Computing in Education (AACE) https://www.learntechlib.org/p/13739

Drachsler, H., & Greller, W. (2016). Privacy and analytics: It's a DELICATE issue a checklist for trusted learning analytics. Proceedings of the Sixth International Conference on Learning Analytics & Knowledge - LAK '16, 89–98. https://doi.org/10.1145/2883851.2883893

Duit, R. (2014). Teaching and Learning the Physics Energy Concept. In R. F. Chen, A. Eisenkraft, D. Fortus, J. Krajcik, K. Neumann, J. Nordine, & A. Scheff (Eds.), *Teaching and Learning of Energy in K – 12 Education* (pp. 67–85). Springer International Publishing. https://doi.org/10.1007/978-3-319-05017-1_5

Dzikovska, M., Nielsen, R., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., Clark, P., Dagan, I., & Dang, H. T. (2013). SemEval-2013 Task 7: The Joint Student Response Analysis and 8th Recognizing Textual Entailment Challenge. Second Joint Conference on Lexical and Computational Semantics (*SEM), Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), 263–274. https://aclanthology.org/S13-2045

Eckart de Castilho, R., & Gurevych, I. (2014). A broad-coverage collection of portable NLP components for building shareable analysis pipelines. Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT, *1–11*. https://doi.org/10.3115/v1/W14-5201

Firth, J. R. (1957). A synopsis of linguistic theory 1930-55. *Studies in Linguistic Analysis (Special Volume of the Philological Society)*, *1952–59*, 1–32.

Gautam, D., & Rus, V. (2020). Using Neural Tensor Networks for Open Ended Short Answer Assessment. In I. I. Bittencourt, M. Cukurova, K. Muldner, R. Luckin, & E. Millán (Eds.), *Artificial Intelligence in Education* (pp. 191–203). Springer International Publishing.

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., & Wichmann, F. A. (2020). Shortcut learning in deep neural networks. *Nature Machine Intelligence*, *2*(11), 665–673. https://doi.org/10.1038/s42256-020-00257-z

Greller, W., & Drachsler, H. (2012). Translating Learning into Numbers: A Generic Framework for Learning Analytics. *Journal of Educational Technology & Society*, *15*(3), 42–57.

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*, 326–336 https://aclanthology.org/W12-2039

Hastie, T., & Tibshirani, R. (1987). Generalized Additive Models: Some Applications. *Journal of the American Statistical Association*, *82*(398), 371–386. https://doi.org/10.1080/01621459.1987.10478440

Herrmann-Abell, C. F., & DeBoer, G. E. (2018). Investigating a learning progression for energy ideas from upper elementary through high school: Learning progression for energy ideas. *Journal of Research in Science Teaching*, *55*(1), 68–93. https://doi.org/10.1002/tea.21411

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Horbach, A., Palmer, A., & Pinkal, M. (2013). Using the text to evaluate short answers for reading comprehension exercises. Second Joint Conference on Lexical and Computational Semantics (*SEM), Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity, 286–295. https://aclanthology.org/S13-1041

Jimenez, S., Becerra, C., & Gelbukh, A. (2013). SOftcardinality: Hierarchical Text Overlap for Student Response Analysis. Second Joint Conference on Lexical and Computational Semantics (*SEM), Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), 280–284. https://aclanthology.org/S13-2047

Klein, R., Kyrilov, A., & Tokman, M. (2011). Automated assessment of short free-text responses in computer science using latent semantic analysis. Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education - ITiCSE '11, *158*. https://doi.org/10.1145/1999747.1999793

Krajcik, J. S., & Shin, N. (2014). Project-Based Learning. In R. K. Sawyer (Ed.), *The Cambridge Handbook of the Learning Sciences* (2nd ed., pp. 275–297). Cambridge University Press. https://doi.org/10.1017/CBO9781139519526.018

Kubsch, M., Nordine, J., Neumann, K., Fortus, D., & Krajcik, J. (2019). Probing the Relation between Students' Integrated Knowledge and Knowledge-in-Use about Energy using Network Analysis. *EURASIA*

Journal of Mathematics, Science and Technology Education, *15*(8), 1–20. https://doi.org/10.29333/ejmste/104404

LAK. (2011). 1st International Conference Learning Analytics and Knowledge. http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=11606

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

Landis, J. R., & Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, *33*(1), 159. https://doi.org/10.2307/2529310

Leacock, C., & Chodorow, M. (2003). C-rater: Automated Scoring of Short-Answer Questions. *Computers and the Humanities*, *37*(4), 389–405. https://doi.org/10.1023/A:1025779619903

Lee, H.-S., & Liu, O. L. (2010). Assessing learning progression of energy concepts across middle school grades: The knowledge integration perspective: Knowledge Integration Assessment. *Science Education*, *94*(4), 665–688. https://doi.org/10.1002/sce.20382

Levy, O., Zesch, T., Dagan, I., & Gurevych, I. (2013). UKP-BIU: Similarity and Entailment Metrics for Student Response Analysis. Second Joint Conference on Lexical and Computational Semantics (*SEM), Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), 285–289. https://aclanthology.org/S13-2048

Linn, M. C. (2006). The Knowledge Integration Perspective on Learning and Instruction. In *The Cambridge handbook of: The learning sciences* (pp. 243–264). Cambridge University Press.

Liu, N. F., Gardner, M., Belinkov, Y., Peters, M. E., & Smith, N. A. (2019). Linguistic knowledge and transferability of contextual representations. Proceedings of the 2019 Conference of the North AMerican Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 1073–1094. https://doi.org/10.18653/v1/N19-1112

Liu, O. L., Rios, J. A., Heilman, M., Gerard, L., & Linn, M. C. (2016). Validation of automated scoring of science assessments. *Journal of Research in Science Teaching*, *53*(2), 215–233. https://doi.org/10.1002/tea.21299

Liu, O. L., Ryoo, K., Linn, M. C., Sato, E., & Svihla, V. (2015). Measuring Knowledge Integration Learning of Energy Topics: A two-year longitudinal study. *International Journal of Science Education*, *37*(7), 1044–1066. https://doi.org/10.1080/09500693.2015.1016470

Liu, X., & McKeough, A. (2005). Developmental growth in students' concept of energy: Analysis of selected items from the TIMSS database. *Journal of Research in Science Teaching*, *42*(5), 493–517. https://doi.org/10.1002/tea.20060

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*. http://arxiv.org/abs/1907.11692

Livingston, S. A. (2009). Constructed-Response Test Questions: Why We Use Them; How We Score Them. R&D Connections. Number 11. In *Educational Testing Service*. Educational Testing Service https://eric.ed.gov/?id=ED507802

Loshchilov, I., & Hutter, F. (2019). *Decoupled Weight Decay Regularization*. 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. https://openreview.net/forum?id=Bkg6RiCqY7

Lou, Y., Caruana, R., Gehrke, J., & Hooker, G. (2013). Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 623–631). ACM. https://doi.org/10.1145/2487575.2487579

Lukhele, R., Thissen, D., & Wainer, H. (1994). On the Relative Value of Multiple-Choice, Constructed Response, and Examinee-Selected Items on Two Achievement Tests. *Journal of Educational Measurement*, *31*(3), 234–250.

Maharjan, N., Gautam, D., & Rus, V. (2018). Assessing Free Student Answers in Tutorial Dialogues Using LSTM Models. In C. Penstein Rosé, R. Martínez-Maldonado, H. U. Hoppe, R. Luckin, M. Mavrikis, K. Porayska-Pomsta, B. McLaren, & B. du Boulay (Eds.), *Artificial Intelligence in Education* (Vol. 10948, pp. 193–198). Springer International Publishing. https://doi.org/10.1007/978-3-319-93846-2_35

Marvaniya, S., Saha, S., Dhamecha, T. I., Foltz, P., Sindhgatta, R., & Sengupta, B. (2018). Creating Scoring Rubric from Representative Student Answers for Improved Short Answer Grading. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (pp. 993–1002). ACM. https://doi.org/10.1145/3269206.3271755

Mason, L., Baxter, J., Bartlett, P., & Frean, M. (2000). Boosting Algorithms as Gradient Descent. In S. Solla, T. Leen, & K. Müller (Eds.), *Advances in Neural Information Processing Systems* (Vol. 12). MIT Press https://proceedings.neurips.cc/paper/1999/file/96a93ba89a5b5c6c226e49b88973f46e-Paper.pdf

McClelland, J. L., & Cleeremans, A. (2009). Connectionist Models. In B. Tim, C. Axel, & W. Patrick (Eds.), *The Oxford Companion to Consciousness*. Oxford University Press.

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 2011 Workshop on Textual Entailment, 1–9. https://aclanthology.org/W11-2401

Mislevy, R. J., Almond, R. G., & Lukas, J. F. (2003). A brief introduction to evidence-centered design. *ETS Research Report Series*, *2003*(1), i–29. https://doi.org/10.1002/j.2333-8504.2003.tb01908.x

Mislevy, R. J., & Haertel, G. D. (2007). Implications of Evidence-Centered Design for Educational Testing. *Educational Measurement: Issues and Practice*, *25*(4), 6–20. https://doi.org/10.1111/j.1745-3992.2006.00075.x

Mitchell, T., Russell, T., Broomhead, P., & Aldridge, N. (2002). Towards robust computerised marking of free-text responses. Proceedings of the 6th CAA Conference, Loughborough University.

Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., & Yu, B. (2019). Interpretable machine learning: Definitions, methods, and applications. *Proceedings of the National Academy of Sciences*, *116*(44), 22071–22080. https://doi.org/10.1073/pnas.1900654116

National Research Council (U.S.)., C. on a C. F. for N. K.-12 S. E. S, (2012). *A framework for K-12 science education: Practices, crosscutting concepts, and core ideas*. National Academies. http://site.ebrary.com/id/10565370

Neumann, K., Viering, T., Boone, W. J., & Fischer, H. E. (2013). Towards a learning progression of energy. *Journal of Research in Science Teaching*, *50*(2), 162–188. https://doi.org/10.1002/tea.21061

Nivre, J., de Marneffe, M.-C., Ginter, F., Hajič, J., Manning, C. D., Pyysalo, S., Schuster, S., Tyers, F., & Zeman, D. (2020). Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection. Proceedings of the 12th Language Resources and Evaluation Conference, 4034–4043. https://aclanthology.org/2020.lrec-1.497

Nori, H., Jenkins, S., Koch, P., & Caruana, R. (2019). InterpretML: A Unified Framework for Machine Learning Interpretability. *ArXiv preprint*, 1909.09223.

Ott, N., Ziai, R., Hahn, M., & Meurers, D. (2013). CoMeT: Integrating different levels of linguistic modeling for meaning assessment. Second Joint Conference on Lexical and Computational Semantics (*SEM), Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), 608–616. https://aclanthology.org/S13-2102

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-Learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Pellegrino, J. W. (2013). Proficiency in science: Assessment challenges and opportunities. *Science*, *340*(6130), 320–323. https://doi.org/10.1126/science.1232065

Pellegrino, J. W., DiBello, L. V., & Goldman, S. R. (2016). A framework for conceptualizing and evaluating the validity of instructionally relevant

assessments. *Educational Psychologist*, *51*(1), 59–81. https://doi.org/10.1080/00461520.2016.1145550

Poulton, A., & Eliens, S. (2021). Explaining transformer-based models for automatic short answer grading. 2021 5th International Conference on Digital Technology in Education, 110–116. https://doi.org/10.1145/3488466.3488479

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations. https://nlp.stanford.edu/pubs/qi2020stanza.pdf

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

Rogers, A., Kovaleva, O., & Rumshisky, A. (2020). A Primer in BERTology: What We Know About How BERT Works. *Transactions of the Association for Computational Linguistics*, *8*, 842–866. https://doi.org/10.1162/tacl_a_00349

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. In C. Penstein Rosé, R. Martínez-Maldonado, H. U. Hoppe, R. Luckin, M. Mavrikis, K. Porayska-Pomsta, B. McLaren, & B. du Boulay (Eds.), *Artificial Intelligence in Education* (Vol. 10947, pp. 503–517). Springer International Publishing. https://doi.org/10.1007/978-3-319-93843-1_37

Sekretariat der ständigen Konferenz der Kultusminister der Länder in der Bundesrepublik Deutschland. (2020). *Bildungsstandards im Fach Physik für die Allgemeine Hochschulreife*. Carl Link Verlag.

Shaffer, D. W., Collier, W., & Ruis, A. R. (2016). A Tutorial on Epistemic Network Analysis: Analyzing the Structure of Connections in Cognitive, Social, and Interaction Data. *Journal of Learning Analytics*, *3*(3), 9–45. https://doi.org/10.18608/jla.2016.33.3

Siddiqi, R., & Harrison, C. (2008). A systematic approach to the automated marking of short-answer questions. *IEEE International Multitopic Conference*, *2008*, 329–332. https://doi.org/10.1109/INMIC.2008.4777758

Slade, S., & Tait, A. (2019). *Global guidelines: Ethics in Learning Analytics*. Association for the Advancement of Computing in Education.

Søgaard, A. (2021). Explainable Natural Language Processing. *Synthesis Lectures on Human Language Technologies*, *14*(3), 1–123. https://doi.org/10.2200/S01118ED1V01Y202107HLT051

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, 1070–1075. https://doi.org/10.18653/v1/N16-1123

Sun, X., Yang, D., Li, X., Zhang, T., Meng, Y., Qiu, H., Wang, G., Hovy, E., & Li, J. (2021). Interpreting Deep Learning Models in Natural Language Processing: A Review. arXiv preprint arXiv:2110.10470.

Sung, C., Dhamecha, T. I., & Mukhi, N. (2019). Improving Short Answer Grading Using Transformer-Based Pre-training. In S. Isotani, E. Millán, A. Ogan, P. Hastings, B. McLaren, & R. Luckin (Eds.), *Artificial Intelligence in Education* (Vol. 11625, pp. 469–481). Springer International Publishing. https://doi.org/10.1007/978-3-030-23204-7_39

Thomas, M. S. C., & McClelland, J. L. (2001). Connectionist Models of Cognition. In R. Sun (Ed.), *The Cambridge Handbook of Computational Psychology* (1st ed., pp. 23–58). Cambridge University Press. https://doi.org/10.1017/CBO9780511816772.005

Thomas, P. (2003). The evaluation of electronic marking of examinations. Proceedings of the 8th Annual Conference on Innovation and Technology in Computer Science Education - ITiCSE '03, *50*. https://doi.org/10.1145/961511.961528

Uto, M., & Uchida, Y. (2020). Automated Short-Answer Grading Using Deep Neural Networks and Item Response Theory. In I. I. Bittencourt, M. Cukurova, K. Muldner, R. Luckin, & E. Millán (Eds.), *Artificial Intelligence in Education* (pp. 334–339). Springer International Publishing.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems, 6000–6010.

Weinert, F. E. (Ed.). (2002). *Leistungsmessungen in Schulen* (2nd ed.). Beltz.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., … Rush, A. M. (2020). Transformers: State-of-the-Art Natural Language Processing. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 38–45. https://doi.org/10.18653/v1/2020.emnlp-demos.6

Yao, J.-X., Guo, Y.-Y., & Neumann, K. (2017). Refining a learning progression of energy. *International Journal of Science Education*, *39*(17), 2361–2381. https://doi.org/10.1080/09500693.2017.1381356

Zehner, F. (2016). *Automatic Processing of Text Responses in Large-Scale Assessments*, *Doctoral dissertation. Technische Universität München*. http://mediatum.ub.tum.de/node?id=1296326

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). https://aclanthology.org/L18-1365

How to cite this article: Gombert, S., Di Mitri, D., Karademir, O., Kubsch, M., Kolbe, H., Tautz, S., Grimm, A., Bohm, I., Neumann, K., & Drachsler, H. (2023). Coding energy knowledge in constructed responses with explainable NLP models. *Journal of Computer Assisted Learning*, *39*(3), 767–786. https://doi.org/10.1111/jcal.12767