Similarity-Based Content Scoring - A more Classroom-Suitable Alternative to Instance-Based Scoring?

Marie Bexte[1] and **Andrea Horbach**[1,2] and **Torsten Zesch**[1]
[1]CATALPA, FernUniversität in Hagen, Germany
[2]Hildesheim University, Germany

Abstract

Automatically scoring student answers is an important task that is usually solved using instance-based supervised learning. Recently, similarity-based scoring has been proposed as an alternative approach yielding similar performance. It has hypothetical advantages such as a lower need for annotated training data and better zero-shot performance, both of which are properties that would be highly beneficial when applying content scoring in a realistic classroom setting. In this paper we take a closer look at these alleged advantages by comparing different instance-based and similarity-based methods on multiple data sets in a number of learning curve experiments. We find that both the demand on data and cross-prompt performance is similar, thus not confirming the former two suggested advantages. The by default more straightforward possibility to give feedback based on a similarity-based approach may thus tip the scales in favor of it, although future work is needed to explore this advantage in practice.

1 Introduction

Approaches in automatic content scoring can be classified into two paradigms: *instance-based scoring* and *similarity-based scoring* (Horbach and Zesch, 2019). Figure 1 gives a schematic overview of the two, with most work in the area of content scoring falling into the instance-based paradigm, where an algorithm is trained on learner answers as the only information source and learns about properties of correct and incorrect answers directly from these answers. In similarity-based scoring, in contrast, learner answers are compared to one or more target answers and correctness judgments are based on either the similarity to a correct answer (such as a sample solution) or on the label of the closest answer(s) to a given learner answer.

In comparison to the instance-based paradigm, similarity-based scoring is substantially less well researched (see e.g. Sakaguchi et al. (2015)). Recent work by Bexte et al. (2022) shows that similarity-based content scoring methods can yield comparable results to instance-based scoring if a similarity metric is substantially fine-tuned. However, it also showed that more research is needed to understand when it can be successful and how it compares to instance-based scoring. To do this, we first identify three possible advantages of similarity-based scoring: reduced **data hunger**, better **cross-prompt performance** and **explainability**. These aspects would be highly beneficial when it comes to the application of automatic scoring in a realistic classroom setting: A typical classroom (ideally) does not consist of hundreds of students, meaning that collecting large amounts of answers to a question from students is unrealistic. Since state-of-the-art content scoring builds on prompt-specific models, it would be highly desirable for a model to either be able to work well on this smaller amount directly or at least by making use of larger already existing cross-prompt data in training a prompt-specific model. Finally, feedback has been identified as one of the major influence factors for learning success (Hattie and Timperley, 2007), but one-on-one student-teacher time is limited, so a model that can justify why it awarded a certain number of points would be preferred over a performance-wise comparable one that simply returns a score.

We perform a comparison of the two paradigms on different data sets typically used in one but not the other, focusing on a setup with limited data and also assessing to what extent using cross-prompt data can help overcome these limitations. We find that while overall highly-dependent on the choice of cross-prompt data, instance-based scoring benefits more. For a more encompassing comparison of the two paradigms, we also compute learning curves extending over a wider range of training data sizes and while we find that there is no one best method for smaller amounts of data, there is a

Findings of the Association for Computational Linguistics: ACL 2023, pages 1892–1903

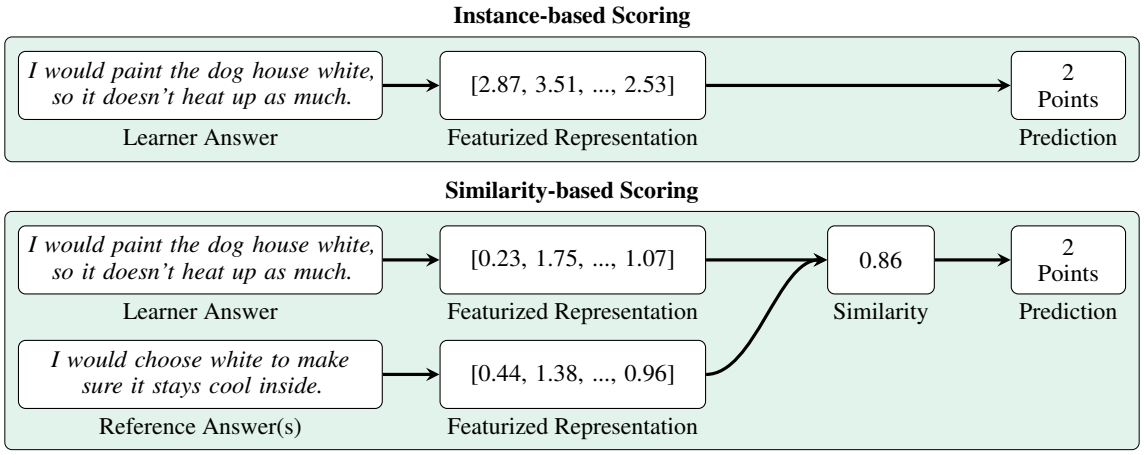


Figure 1: Comparison of instance-based and similarity-based scoring.

point where similarity-based deep learning starts to consistently outperform all other methods, closely followed by instance-based deep learning. In comparing how much predictions vary based on the choice of training data, we find an overall smaller standard deviation for similarity-based predictions. We make all our code publicly available.[1]

2 Instance-Based vs. Similarity-Based Scoring

Instance-based scoring has become the de facto state of the art in automated scoring. Recent experiments however showed that, with the emergence of deep learning, similarity-based models can keep up with instance-based ones:

For essay scoring, Xie et al. (2022) use a BERT model in a pairwise contrastive regression setup to score an essay in comparison to a reference, thereby outperforming the instance-based state of the art. For content scoring, Bexte et al. (2022) reach comparable performance to an instance-based BERT model by using fine-tuned SBERT embeddings in a knn-like search for the most similar answer(s). Tunstall et al. (2022) introduce Sentence Transformer Finetuning (SETFIT), which successfully uses SBERT in a few-shot setting by using the fine-tuned embeddings to train a classification head.

In line with this low-resource setup, similarity-based scoring is often applied to data sets containing only few answers per prompt. This includes work on computer science questions (Mohler and Mihalcea, 2009; Mohler et al., 2011), English and German reading comprehension data (Bailey and Meurers, 2008; Meurers et al., 2011) and several approaches on the Student Response Analysis data set (Dzikovska et al., 2013), such as Levy et al. (2013) or more recently Willms and Padó (2022). Even though in contrast, research on data with hundreds of answers per prompt or more is often associated with instance-based methods, such as most work on the ASAP data set (e.g., Higgins et al. (2014); Heilman and Madnani (2015); Kumar et al. (2019)), this does not necessarily mean that the data hunger is smaller for similarity-based models than for instance-based models as the former are often used to train a classifier across prompts. Still, also considering the recent success of SETFIT in a few-shot setting , we address the perceived dichotomy in data sets by contrasting the performance of both paradigms on both kinds of data sets. This gives insight into the difference regarding their **data hunger**. To investigate the supposed advantage of similarity-based scoring on limited data, we focus on learning curve experiments on smaller amounts of training data.

Previous work comparing instance-based to similarity-based scoring however showed similarity-based performance to be close to the respective best-performing instance-based model on both small (Logistic Regression) or larger amounts of training data (BERT). (Bexte et al., 2022), whereas Logistic Regression and BERT have their strengths towards the lower and higher end of the training size spectrum, respectively. To further investigate this, we extend our learning curves beyond the low-resource spectrum and include a wider range of training sizes.

Another aspect Tunstall et al. (2022) already touched on the influence of the reference answer choice on scoring performance, thus asking how

[1] https://github.com/mariebexte/sbert-learning-curves

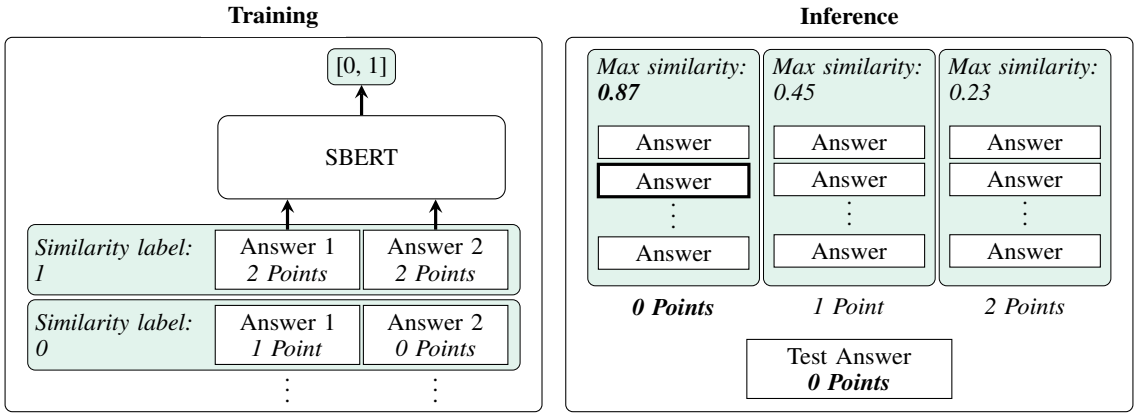


Figure 2: Overview of how SBERT is used for similarity-based scoring, adapted from Bexte et al. (2022). Left: Fine-tuning using pairs of answers with either the same (similarity label 1) or a different (similarity label 0) scoring label. Right: Using SBERT at inference to identify the most similar training answer, thus predicting its number of points for the test answer.

(un)lucky one can be when selecting these and whether it is worth investing time to carefully pick them. To investigate this, we compare the standard deviation of different training data samples for instance-based and similarity-based scoring.

As mentioned above, the dichotomy of similarity-based and instance-based data sets is accompanied by instance-based scoring typically training one model per prompt, while similarity-based approaches often make use of data across different prompts, suggesting a possible superiority of similarity-based methods regarding **cross-prompt transfer**. Further supporting this notion is the fact that a similiarity-based model won the cross-prompt track of the 2021 NAEP Automatic Scoring challenge[2], although the overall performance level of submissions lagged behind state-of-the-art instance-based models in within-prompt settings. It is however unclear how well a state-of-the-art instance-based model would fare on the same cross-prompt data, as such comparisons are lacking. Condor et al. (2021) use different ways of encoding answers to train a cross-prompt model in an instance-based fashion. They find SBERT embeddings to be superior over Word2Vec embeddings or a bag of words approach, leaving open the question of whether using the SBERT embeddings in a similarity-based fashion would have yielded even better performance. Since the similarity-based zero-shot cross-prompt experiments by Bexte et al. (2022) showed mixed results, we undertake a comparison of the non-zero-shot cross-prompt performance of instance-based and similarity-based methods.

A third possible advantage of similarity-based scoring that requires user studies to investigate and is thus beyond the scope of this paper is that one can show which reference answer(s) led to a certain classification decision, by default lending it a certain degree of **explainability** that could serve as pedagogical **feedback** to students. This feedback is mainly aimed at students or teachers as opposed to AI experts, since we do not directly disclose the inner workings of the algorithm, but rather provide some rationale about why a score has been assigned. A similar direction is addressed by clustering approaches for automatic scoring (such as Basu et al. (2013); Wolska et al. (2014); Zehner et al. (2016)) with clustering essential also being a similarity-based method bearing the advantage of structured output that can be used to provide human feedback to learners efficiently.

To summarize, we identified three potential benefits of similarity-based models: a reduced training data hunger, the ability to abstract across prompts and the possibility of giving feedback based on reference answers, the latter of which we leave for future work.

3 Experimental Setup

3.1 Scoring Approaches

Similarity-based approach We use the similarity-based approach described in Bexte et al. (2022), where a pre-trained Sentence-BERT (SBERT) model (All-miniLM-L6-v2) is fine-tuned on sentence pairs formed from the training data.

[2] https://github.com/NAEP-AS-Challenge/reading-prediction

These sentence pairs are labeled with a similarity score of 1 (0), if both answers in the pair have the same (a different) label. In this manner, we create as many pairs as possible. Figure 2 gives an overview of this fine-tuning setup, and also shows how the fine-tuned model is then used to obtain predictions on the test data: With the training data serving as a set of reference answers, each answer from the test set is compared against every answer from the training set, and the label of the most similar training answer is then used as prediction.

We train for 5 epochs with batch size 8 and without warmup, using an OnlineContrastiveLoss and an EmbeddingSimilarityEvaluator, otherwise keeping all values at their defaults. Validation is done after each epoch and we use the model with minimal validation loss for evaluation on the test data.

Similarity-based baselines Since similarity-based scoring also works without any finetuning, we include similarity-based baselines that essentially perform only the inference step described in the above SBERT setup. An answer from the test set is thus compared to all answers from the reference (i.e. training) set, predicting the scoring label of the most similar reference answer.

While we also ran experiments using overlap and cosine similarity of word count vectors[3], we for the sake of brevity only report results for **edit** distance, as an example of surface similarity, and the **pre-trained** SBERT model without any adaptation to the respective prompt, as an example of working on vectorized representations.

Instance-based approaches Experimenting with a number of shallow algorithms[4] showed **Logistic Regression (LR)** to perform best, which is why we only report results for this method. We used the scikit-learn implementation in standard configuration (apart from setting max_iter to 1000) with token uni- to trigram features. As a representation of instance-based deep learning, we also fine-tune a **BERT** model (bert_base_uncased) from huggingface[5]. We train this model for 20 epochs with a batch size of 8, running evaluation after each epoch and keeping the model with the lowest validation loss for evaluation on testing data. Other than that, parameters are kept at their default values.

[3]Results using these methods were in the same ballpark as edit distance and pre-trained SBERT model.

[4]We used SVMs, random forests and logistic regression.

[5]https://huggingface.co/bert-base-uncased

	ASAP	**SRA-Beetle**	**SRA-SEB**
Domains	Science, Bio, ELA*	Electricity, Electronics	Science
# Prompts	10	47	135
# Answers/prompt			
- Train	1704	84	37
- Test	522	9	4
Label set			
- # Labels	2-3	2 or 5	2 or 5
- Scale	numerical	categorical	categorical

Table 1: data sets used in our experiments. *English Language Arts

We trained on NVIDIA Quadro RTX 6000 and A100 GPUs for a total of close to 4000 GPU hours.

3.2 Data

We perform experiments on two widely used English content scoring data sets that are freely available for research purposes: **ASAP**[6], which is typically used for instance-based scoring, and the **Student Response Analysis (SRA)** corpus (Dzikovska et al., 2013), which has often been used for similarity-based experiments and consists of the two subsets **Beetle** and SciEntsBank (**SEB**). Since these data sets consist of answers to factual questions, they do not contain identifying information of students or offensive content.

While labels in ASAP are numerical (0 to either 2 or 3 points), answers in SRA are labeled nominally following a textual entailment view on automatic scoring with 5 possible outcomes: *correct*, *contradictory*, *partially_correct_incomplete*, *irrelevant* or *non_domain*. We refer to this data set as **5-way**. In addition, we also use the **2-way** version, where labels other than *correct* are merged into an *incorrect* class.

We use the default split into training and test data as provided in the respective data set. In all deep learning setups (i.e. fine-tuning BERT & SBERT), we use parts of the training data for each prompt as a separate validation data set, whereas in shallow learning all training instances are used in the actual learning process. The rationale behind this is that we want to compare the overall amount of human annotation effort required to train a model, regardless how exactly this annotated data is used.

We randomly chose 4 answers per prompt for validation. Picking just 4 answers might seem a low number, but is reasonable since our experiments

[6]https://www.kaggle.com/c/asap-sas

specifically target the use of limited training data.[7]

3.3 Evaluation

We compare the instance-based and similarity-based methods in a learning curve setup to examine the influence of different training set sizes. For ASAP with numeric labels, we use quadratically weighted kappa (QWK) (Cohen, 1968) as evaluation metric, whereas we use weighted F1 measure for the categorical labels in SRA.

Depending on the number of labels present in a data set, we consider different training sizes for the learning curve. For ASAP and SRA with 5-way labels, we start with five instances and go up to 50 in steps of five. For SRA with 2-way labels, we start with two instances, and also go up to 50, but first in steps of 2 (until 14 instances) and then in steps of four. For each training size, we train with 20 different randomly taken training data samples to mitigate sampling effects.

Due to the low number of on average 37 answers per prompt in SEB, we for this data set cut off results at a maximum training size of 30, as results for larger training sizes would only rely on the few prompts with enough answers to compute these results. Also note that the limited number of training answers to sample from allows for little variance between the 20 randomly sampled subsets.

4 Data Hunger

In comparing instance-based and similarity-based scoring methods, we focus on the amount of training data needed (i.e. how data hungry the approaches are). We focus on the low-resource setting, as (i) it is more realistic in a classroom setting, and (ii) the fact that similarity-based and instance-based perform on par has already been established when training data is abundant (Bexte et al., 2022).

Results in Figure 3 show that SBERT has the upper hand on SEB and ASAP, while it is outperformed by LR on Beetle. Other than on ASAP, baseline similarity-based methods are often surprisingly strong on both Beetle and SEB. We speculate that this might be due to shorter and simpler answers, which is also indicated by a higher overall performance. As expected, performance is overall higher on the 2-way-labeled data, but apart from this, relative results of the different methods are similar on the five-way-labeled data. Note that results are averaged across all prompts of the respective data set and that individual performances per prompt again vary tremendously.

One application that would benefit from models that are doing well on small amounts of data is the use of automated scoring in a realistic classroom setting, since the average number of students in a class does not allow collecting larger amounts of answers to any given question. If a teacher were however to make up exemplary answers for the different possible outcomes, they might produce a more balanced sample of reference answers than what we use in our random sampling of training data. In Figure 4(a), we therefore also show learning curves using balanced sampling of ASAP data, which means that samples will contain the same amount of answers for each label.[8]

Averaged for LR, BERT and SBERT over all training sizes, this yields a .09 increase in QWK compared to random sampling. The order of performance for individual methods does however vary substantially between the two settings and across different training sizes, with a tendency in most cases of SBERT outperforming other methods and the baseline methods (pre-trained and edit) being inferior. A curious exception to this observation is the curve for BERT on randomly drawn data.

Previous work on ASAP had found that both BERT and SBERT outperform LR on larger amounts of training data, while LR was superior on smaller data sizes (Bexte et al., 2022). Although our results do not find a general superiority of LR, we take a closer look at how the different methods compare for larger training sizes. We therefore extend the ASAP learning curve (with random sampling) to include up to 1000 training instances (Figure 4(b)).[9] We observe that soon after 100 training instances, there is a clear advantage of neural over shallow methods, with SBERT outperforming LR much earlier. Overall, SBERT consistently outperforms or is at least on par with all other methods.

4.1 Potential for Combining Approaches

As the different methods sometimes show widely differing performance, one idea towards improving overall performance is to combine their predictions.

[7] We also validated on a few random prompts that this split is a good trade-off to save as many instances as possible for the actual training process.

[8] Since, apart from the expected slight performance increase, there were no notable effects of the different sampling strategy on SRA, Figure 4(a) only shows results on ASAP.

[9] Due to data set sizes, this experiment can only be performed on ASAP.

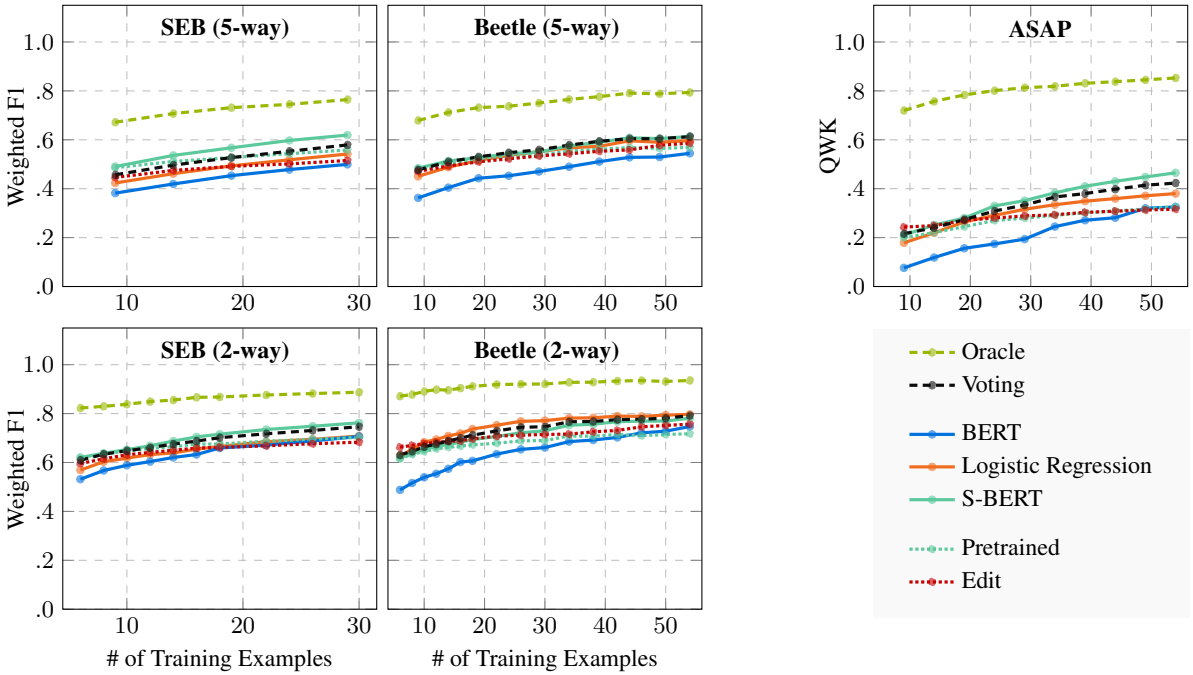


Figure 3: Learning curves on Beetle and SEB with 5-way (top) and 2-way (bottom) labeling, as well as on ASAP. Dotted lines represent baselines, dashed ones combinations of the individidual methods.

We do this in two different ways:

In the **voting** condition, we employ a majority voting strategy over the predictions of all methods, i.e. take the most frequently predicted label. In case of ties, we randomly decide on one of them.

In the **oracle** condition, we predict the correct label whenever at least one of the methods is able to do so. If none of them is, we use the prediction that is closest to the ground truth. This is of course a hypothetical, idealized setting, as we in practice do not know beforehand which method gives the correct prediction, and can therefore be seen as the ceiling performance on combining all methods.

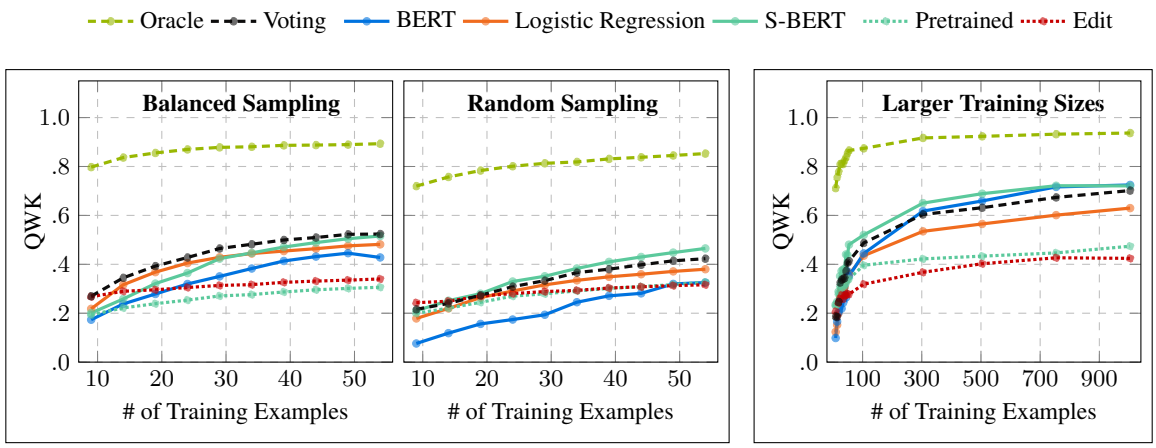
Results for both settings are included in Figures 3 and 4. The only setting where the voting condition tops all individual methods is ASAP with balanced sampling. In all other cases there is enough disagreement between the individual predictions that there is always one method that is on par with and in many cases even outperforming the voting condition. Combining predictions of all methods into an oracle condition, however, yields a pronounced performance increase of around .2 in weighted F1 for SRA and an even more pronounced one of around .4 in QWK for ASAP, suggesting that future experiments might build a stacked classifier to test how much of this potential can be realized.

To dissect the cause for these performance increases, we perform two further analyses: In the **unique** condition, we for each of the methods evaluate which proportion of the answers in a data set was scored correctly by the respective method alone, i.e. misclassified by all other methods. In the **all** condition, we evaluate which proportion of answers was scored correctly by all methods, i.e. misclassified by none of them.

			Unique to				
		All	LR	BERT	SBERT	Pretr.	Edit	Σ
2-way	Beetle	.44	.02	.02	.01	.01	.03	.09
	SEB	.44	.02	.02	.01	.01	.02	.08
5-way	Beetle	.32	.02	.02	.01	.01	.03	.09
	SEB	.30	.02	.02	.02	.01	.03	.10
	ASAP	.23	.02	.03	.02	.03	.04	.14

Table 2: Overview of which percentage of the test answers only one of the methods classifies correctly (unique to), and for which proportion all of them are able to predict the correct score (all).

Table 2 shows the results, with the percentage of answers falling into the all condition indicating how many are easy to predict correctly, which is of course varying in line with the overall performance level on the different data sets. We observe the highest proportion of 'easy' answers .44 for SRA with 2-way labeling and the lowest of .23 for ASAP.



(a) Influence of training data sampling (b) Longer Curves

Figure 4: Additional learning curves on the ASAP data. (a) Comparing different training data sampling strategies. (b) Computing curves for larger training sizes (with random sampling of training data).

While this proportion tells us how many answers are reasonably easy to score correctly, it also tells us that the remainder of the answers it mislabeled by at least one of the methods. Taking this to the extreme and looking at the fraction of answers that is scored correctly by only one of them, i.e. looking towards unique condition, the per-method percentages are highest for ASAP and lowest for the SBERT methods (both pre-trained and fine-tuned). Even though the individual numbers may overall not seem that high, note that in the oracle condition it is actually the sum of all these proportions that contributes to the observed high performance.

4.2 Influence of Reference Answer Selection

The choice of the specific training answers (which are the reference answers in similarity-based scoring) influences performance beyond the balanced/random dichotomy. To highlight this variability, Figure 5 plots the distribution of performances across the 20 runs for ASAP for both balanced and randomly sampled data.[10]

In general, we see that standard deviation is lower and varies less for SBERT than for BERT. Notably, for SBERT it shows a further decline for larger training sizes when using balanced sampling, which we do not see for BERT. A similarly pronounced decline in standard deviation was observed for the similarity-based baselines. Overall, this indicates that the choice of reference answers for the similarity-based approach introduces less variance than for instance-based BERT training .

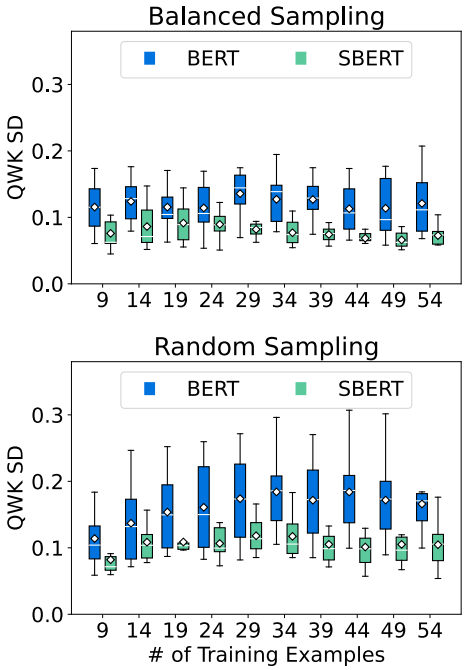


Figure 5: Distribution of average SD over the 10 ASAP prompts for BERT and SBERT, with balanced (top) and random (bottom) sampling of training examples.

5 Cross-Prompt Scoring

Another claim often implicitly attached to similarity-based methods is that they might have **greater capabilities of learning a cross-prompt model**. This intuitively makes sense as instance-based approaches rely on the presence or absence of certain lexical material while similarity-based approaches can exploit the closeness to a model

[10] We limit this analysis to ASAP, as its larger pool of training instances allows for more sampling variance. For the sake of brevity we only report results for BERT and SBERT.

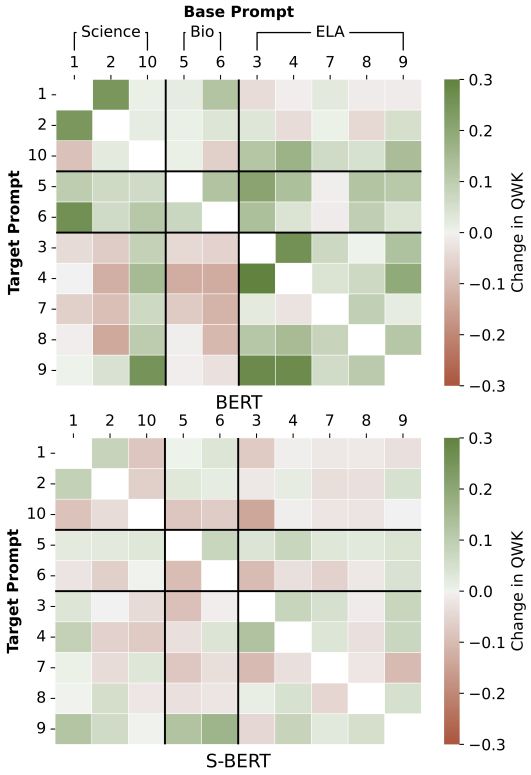


Figure 6: Cross-prompt performance change over within-prompt performance for BERT (top) and SBERT (bottom). Black borders indicate topic groups.

answer. Bexte et al. (2022) did however find that in some cases fine-tuning an SBERT model to one prompt before adapting it to another was actually detrimental to performance, with an off-the-shelf pre-trained SBERT model sometimes even outperforming the fine-tuned one. Since they did a zero-shot application to the new prompt, no data from the target prompt was used adapt the model to it.

We therefore first fine-tune a model on 1000 answers from a base prompt, and then use a smaller amount (again building the learning curves from Figure 3) to adapt this model to the target prompt.[11]

Figure 6 shows the change in performance for each combination of prompts in ASAP[12] compared to a prompt-specific setup without pre-training (i.e. the results from Experimental Study 1). To gain a better overview, results are not only averaged over all prompts but also all training sizes.

Like Bexte et al. (2022), we group prompts according to the underlying topics Science, Biology ELA, as a transfer within the same topic group might be more successful than one across topic groups. We see that - contrary to the implied superiority of similarity-based scoring - , the largest performance increases of up to .3 in QWK happen for the instance-based BERT model. These relatively pronounced increases mostly occur for transfers within topic groups, but there are also instances of (albeit less) successful cross-prompt transfer, thus partially confirming the hypothesis - at least for BERT. There seems to be a systematic detrimental effect of using a Biology base prompt for a target ELA prompt, which does however not occur when prompts are used the other way round. Apart from this, there is quite some symmetry to the results, meaning that if using prompt A as base for target prompt B helps (harms), the same is true for using B as a base for A.

6 Summary & Future Work

We compared instance-based and similarity-based methods for content scoring, examining whether properties that are often implicitly attributed to the latter are in fact empirically observable. In a set of learning curve experiments directed at the claim of similarity-based methods being less data hungry, we find that a fine-tuned SBERT model does often yield the best results, but not for Beetle, where this method was outperformed by the instance-based logistic regression. The suggested superiority of similarity-based scoring when it comes to smaller training sizes could thus not be confirmed.

When running experiments with larger training sizes on ASAP, SBERT remains the best-performing method up until using 750 training instances, from when on it is joined by Bert. In a comparison of how much performance varies depending on the choice of training data, SBERT had the upper hand, especially when a relatively large amount of balanced training data that is sampled.

Another proposed property of similarity-based scoring is the ability to transfer across prompts. This could however not be confirmed by our experiments, where the largest performance increases were observed for the instance-based BERT model.

Examining performance of a hypothetical oracle condition showed that it might be worthwhile to learn a stacked classifier, thus combining the strengths of the different (both similarity- and instance-based) methods. Other possible avenues of future work are topics that have been researched in the context of instance-based scoring but not, or at least not to the same extent, for similarity-based

[11] We again only report results for the SOTA models BERT and SBERT for the sake of brevity.

[12] As only this data set provides a large enough amount of answers, we only perform this experiment on ASAP.

scoring. These include the importance of spelling errors or the vulnerability to adversarials.

7 Limitations

Since our results regarding a fine-tuned similarity method are limited to the SBERT fine-tuning introduced by Bexte et al. (2022), our findings are limited to this specific similarity-based setup and cannot exclude that other similarity-based methods might behave differently. We also did not consider training sizes larger than 1000 instances of ASAP, and can therefore not speak for how the relative performance of the different methods would be affected by using even more training data. Regarding the experiment on larger training data sizes, we also limited our analysis to ASAP, so it is necessary to compare the observed effects to those that occur on other data sets. The same goes for our cross-prompt experiments, which were also limited to ASAP . Other data sets cover other content domains and can thus produce different effects. Finally, while we do discuss the advantage of a more straightforward explainability of similarity-based models regarding feedback, this is an entirely theoretical argument that goes beyond the scope of this paper and would therefore have to be investigated further in future work.

8 Ethical Considerations

Automatic scoring can foster great efficiency over manual scoring, and can thus, especially considering limitations regarding human scoring resources, be a highly useful addition to the educational world. It enables instantaneous teacher-indepedent feedback and frees up teacher resources.

Nonetheless, automatically scoring student answers brings about a number of concerns regarding when it may be more or less appropriate.

While automated scoring in general can, depending on model implementation and quality, both contribute to and reduce fairness, similarity-based scoring at least provides model introspection at the level of being able to return the answers that lead to a certain classification outcome as feedback. In general, automatic scoring puts a certain pressure of conformity on answers: An answer that differs in style from what was observed during training, irrespective of whether it is in fact correct, is at risk of being misclassified.

Regarding such biases, it should be noted that humans are not perfect either - but an English teacher is biased against a particular student, they still have the option of switching classes. The same may not be possible if a widely used scoring model is negatively biased against the kinds of answers they give.

Finally, whether to use automatic or manual scoring does not have to be a question of one or the other - it may be worthwhile to have a model only perform a first grouping, in hopes that this would speed up the human grading process (Pado and Kiefer, 2015) or return answers it is unsure about for manual reassessment. Another option that is already employed in practice (for example by the Educational Testing Service) is to have the same set of answers graded by both a human and a scoring model, only requiring a second humand annotator when there is too much disagreement between the two. This ensures that the high-stakes TOEFL test can benefit from more efficient, machine-supported scoring while also putting a layer of quality control on its predictions. In a lower-stakes scoring setup, for example in an optional training exercise for students, one may want to be more lenient towards the model predictions, employing a scoring approach without human involvement at the risk of getting a certain percentage of erroneous predictions.

Acknowledgements

This work was partially conducted at "CATALPA - Center of Advanced Technology for Assisted Learning and Predictive Analytics" of the FernUniversität in Hagen, Germany, and partially within the KI-Starter project "Explaining AI Predictions of Semantic Relationships" funded by the Ministry of Culture and Science, Nordrhein-Westfalen, Germany.

References

Stacey Bailey and Detmar Meurers. 2008. Diagnosing meaning errors in short answers to reading comprehension questions. In *Proceedings of the third workshop on innovative use of NLP for building educational applications*, pages 107–115.

Sumit Basu, Chuck Jacobs, and Lucy Vanderwende. 2013. Powergrading: A clustering approach to amplify human effort for short answer grading. *Transactions of the Association for Computational Linguistics*, 1:391–402.

Marie Bexte, Andrea Horbach, and Torsten Zesch. 2022. Similarity-based content scoring - How to make S-BERT keep up with BERT. In *Proceedings of the*

17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022), pages 118–123.

J. Cohen. 1968. Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. *Psychol Bull.*, 70(4):213–220.

Aubrey Condor, Max Litster, and Zachary Pardos. 2021. Automatic short answer grading with SBERT on out-of-sample questions. *International Educational Data Mining Society*.

Myroslava O Dzikovska, Rodney D Nielsen, Chris Brew, Claudia Leacock, Danilo Giampiccolo, Luisa Bentivogli, Peter Clark, Ido Dagan, and Hoa T Dang. 2013. Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. Technical report, North Texas State Univ Denton.

John Hattie and Helen Timperley. 2007. The power of feedback. *Review of educational research*, 77(1):81–112.

Michael Heilman and Nitin Madnani. 2015. The impact of training data on automated short answer scoring performance. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 81–85.

Derrick Higgins, Chris Brew, Michael Heilman, Ramon Ziai, Lei Chen, Aoife Cahill, Michael Flor, Nitin Madnani, Joel Tetreault, Daniel Blanchard, et al. 2014. Is getting the right answer just about choosing the right words? The role of syntactically-informed features in short answer scoring. *arXiv preprint arXiv:1403.0801*.

Andrea Horbach and Torsten Zesch. 2019. The influence of variance in learner answers on automatic content scoring. In *Frontiers in Education*, volume 4, page 28. Frontiers.

Yaman Kumar, Swati Aggarwal, Debanjan Mahata, Rajiv Ratn Shah, Ponnurangam Kumaraguru, and Roger Zimmermann. 2019. Get it scored using autosas — An automated system for scoring short answers. In *Proceedings of the AAAI conference on artificial intelligence*, pages 9662–9669.

Omer Levy, Torsten Zesch, Ido Dagan, and Iryna Gurevych. 2013. Recognizing partial textual entailment. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 451–455.

Detmar Meurers, Ramon Ziai, Niels Ott, and Janina Kopp. 2011. Evaluating answers to reading comprehension questions in context: Results for German and the role of information structure. In *Proceedings of the TextInfer 2011 Workshop on Textual Entailment*, pages 1–9.

Michael Mohler, Razvan Bunescu, and Rada Mihalcea. 2011. Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 752–762.

Michael Mohler and Rada Mihalcea. 2009. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 567–575.

Ulrike Pado and Cornelia Kiefer. 2015. Short answer grading: When sorting helps and when it doesn't. In *Proceedings of the fourth workshop on NLP for computer-assisted language learning*, pages 42–50.

Keisuke Sakaguchi, Michael Heilman, and Nitin Madnani. 2015. Effective feature integration for automated short answer scoring. In *Proceedings of the 2015 conference of the North American Chapter of the association for computational linguistics: Human language technologies*, pages 1049–1054.

Lewis Tunstall, Nils Reimers, Unso Eun Seo Jo, Luke Bates, Daniel Korat, Moshe Wasserblat, and Oren Pereg. 2022. Efficient few-shot learning without prompts. In *Advances in Neural Information Processing Systems*.

Nico Willms and Ulrike Padó. 2022. A transformer for sag: What does it grade? In *Proceedings of the 11th Workshop on NLP for Computer Assisted Language Learning*, pages 114–122.

Magdalena Wolska, Andrea Horbach, and Alexis Palmer. 2014. Computer-assisted scoring of short responses: The efficiency of a clustering-based approach in a real-life task. In *International Conference on Natural Language Processing*, pages 298–310. Springer.

Jiayi Xie, Kaiwei Cai, Li Kong, Junsheng Zhou, and Weiguang Qu. 2022. Automated essay scoring via pairwise contrastive regression. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2724–2733.

Fabian Zehner, Christine Sälzer, and Frank Goldhammer. 2016. Automatic coding of short text responses via clustering in educational assessment. *Educational and psychological measurement*, 76(2):280–303.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
7

☑ A2. Did you discuss any potential risks of your work?
8

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

3.2

☑ B1. Did you cite the creators of artifacts you used?
3.2

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
3.2

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
3.2

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
3.2

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
3.2

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
3.2

C ☑ Did you run computational experiments?

4, 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
3.1

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
3.1

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
4, 5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
3.1

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.