Should You Fine-Tune BERT for Automated Essay Scoring?

Elijah Mayfield and **Alan W Black**
Language Technologies Institute
Carnegie Mellon University
elijah@cmu.edu, awb@cs.cmu.edu

Abstract

Most natural language processing research now recommends large Transformer-based models with fine-tuning for supervised classification tasks; older strategies like bag-of-words features and linear models have fallen out of favor. Here we investigate whether, in automated essay scoring (AES) research, deep neural models are an appropriate technological choice. We find that fine-tuning BERT produces similar performance to classical models at significant additional cost. We argue that while state-of-the-art strategies do match existing best results, they come with opportunity costs in computational resources. We conclude with a review of promising areas for research on student essays where the unique characteristics of Transformers may provide benefits over classical methods to justify the costs.

1 Introduction

Automated essay scoring (AES) mimics the judgment of educators evaluating the quality of student writing. Originally used for summative purposes in standardized testing and the GRE (Chen et al., 2016), these systems are now frequently found in classrooms (Wilson and Roscoe, 2019), typically enabled by training data scored on reliable rubrics to give consistent and clear goals for writers (Reddy and Andrade, 2010).

More broadly, the natural language processing (NLP) research community in recent years has been dominated by deep neural network research, in particular, the Transformer architecture popularized by BERT (Devlin et al., 2019). These models use large volumes of existing text data to pre-train multilayer neural networks with context-sensitive meaning of, and relations between, words. The models, which often consist of over 100 million parameters, are then fine-tuned to a specific new labeled dataset and used for classification, generation, or structured prediction.

Research in AES, though, has tended to prioritize simpler models, usually multivariate regression using a small set of justifiable variables chosen by psychometricians (Attali and Burstein, 2004). This produces models that retain direct mappings between variables and recognizable characteristics of writing, like coherence or lexical sophistication (Yannakoudakis and Briscoe, 2012; Vajjala, 2018). In psychometrics more generally, this focus on features as valid "constructs" leans on a rigorous and well-defined set of principles (Attali, 2013). This approach is at odds with Transformer-based research, and so our core question for this work is: for AES specifically, is a move to deep neural models worth the cost?

The chief technical contribution of this work is to measure results for BERT when fine-tuned for AES. In section 3 we describe an experimental setup with multiple levels of technical difficulty from bag-of-words models to fine-tuned Transformers, and in section 5 we show that the approaches perform similarly. In AES, human inter-rater reliability creates a ceiling for scoring model accuracy. While Transformers match state-of-the-art accuracy, they do so with significant tradeoffs; we show that this includes a slowdown in training time of up to 100x. Our data shows that these Transformer models improve on N-gram baselines by no more than 5%. Given this result, in section 6 we describe areas of contemporary research on Transformers that show both promising early results and a potential alignment to educational pedagogy beyond reliable scoring.

2 Background

In AES, student essays are scored either on a single holistic scale, or analytically following a rubric that breaks out subscores based on "traits." These scores are almost always integer-valued, and almost universally have fewer than 10 possible score points, though some research has used scales with

Proceedings of the 15th Workshop on Innovative Use of NLP for Building Educational Applications, pages 151–162

as many as 60 points (Shermis, 2014). In most contexts, students respond to "prompts," a specific writing activity with predefined content. Work in natural language processing and speech evaluation has used advanced features like discourse coherence (Wang et al., 2013) and argument extraction (Nguyen and Litman, 2018); for proficient writers in professional settings, automated scaffolds like grammatical error detection and correction also exist (Ng et al., 2014).

Natural language processing has historically used n-gram bag-of-words features to predict labels for documents. These were the standard representation of text data for decades and are still in widespread use (Jurafsky and Martin, 2014). In the last decade, the field moved to word *embeddings*, where words are represented not as a single feature but as dense vectors learned from large unsupervised corpora. While early approaches to dense representations using latent semantic analysis have been a major part of the literature on AES (Foltz et al., 2000; Miller, 2003), these were corpus-specific representations. In contrast, recent work is general-purpose, resulting in off-the-shelf representations like GloVe (Pennington et al., 2014). This allows similar words to have approximately similar representations, effectively managing lexical sparsity.

But the greatest recent innovation has been *contextual* word embeddings, based on deep neural networks and in particular, Transformers. Rather than encoding a word's semantics as a static vector, these models adjust the representation of words based on their context in new documents. With multiple layers and sophisticated *attention mechanisms* (Bahdanau et al., 2015), these newer models have outperformed the state-of-the-art on numerous tasks, and are currently the most accurate models on a very wide range of tasks (Vaswani et al., 2017; Dai et al., 2019). The most popular architecture, BERT, produces a 768-dimensional final embedding based on a network with over 100 million total parameters in 12 layers; pre-trained models are available for open source use (Devlin et al., 2019).

For document classification, BERT is "fine-tuned" by adding a final layer at the end of the Transformer architecture, with one output neuron per class label. When learning from a new set of labeled training data, BERT evaluates the training set multiple times (each pass is called an *epoch*). A loss function, propagating backward to the network, allows the model to learn relationships between the class labels in the new data and the contextual meaning of the words in the text. A learning rate determines the amount of change to a model's parameters. Extensive results have shown that careful control of the learning rate in a *curriculum* can produce an effective fine-tuning process (Smith, 2018). While remarkably effective, our community is only just beginning to identify exactly what is *learned* in this process; research in "BERT-ology" is ongoing (Kovaleva et al., 2019; Jawahar et al., 2019; Tenney et al., 2019).

These neural models are just starting to be used in machine learning for AES, especially as an intermediate representation for automated essay feedback (Fiacco et al., 2019; Nadeem et al., 2019). End-to-end neural AES models are in their infancy and have only seen exploratory studies like Rodriguez et al. (2019); to our knowledge, no commercial vendor yet uses Transformers as the representation for high-stakes automated scoring.

3 NLP for Automated Essay Scoring

To date, there are no best practices on fine-tuning Transformers for AES; in this section we present options. We begin with a classical baseline of traditional bag-of-words approaches and non-contextual word embeddings, used with Naïve Bayes and logistic regression classifiers, respectively. We then describe three curriculum learning options for fine-tuning BERT using AES data based on broader best practices. We end with two approaches based on BERT but without fine-tuning, with reduced hardware requirements.

3.1 Bag-of-Words Representations

The simplest features for document classification tasks, "bag-of-words," extracts surface N-grams of length 1-2 with "one-hot" binary values indicating presence or absence in a document. In prior AES results, this representation is surprisingly effective, and can be improved with simple extensions: N-grams based on part-of-speech tags (of length 2-3) to capture syntax independent of content, and character-level N-grams of length 3-4, to provide robustness to misspellings (Woods et al., 2017; Riordan et al., 2019). This high-dimensional representation typically has a cutoff threshold where rare tokens are excluded: in our implementation, we exclude N-grams without at

least 5 occurrences in training data. Even after this reduction, this is a sparse feature space with thousands of dimensions. For learning with bag-of-words, we use a Naïve Bayes classifier with Laplace smoothing from Scikit-learn (Pedregosa et al., 2011), with part-of-speech tagging from SpaCy (Honnibal and Montani, 2017).

3.2 Word Embeddings

A more modern representation of text uses word-level embeddings. This produces a vector, typically of up to 300 dimensions, representing each word in a document. In our implementation, we represent each document as the term-frequency-weighted mean of word-level embedding vectors from GloVe (Pennington et al., 2014). Unlike one-hot bag-of-words features, embeddings have dense real-valued features and Naïve Bayes models are inappropriate; we instead train a logistic regression classifier with the LibLinear solver (Fan et al., 2008) and L2 regularization from Scikit-learn.

3.3 Fine-Tuning BERT

Moving to neural models, we fine-tune an uncased BERT model using the Fast.ai library. This library's visibility to first-time users of deep learning and accessible online learning materials[1] mean their default choices are the most accessible route for practitioners.

Fast.ai recomends use of *cyclical* learning rate curricula for fine-tuning. In this policy, an upper and lower bound on learning rates are established. lr_{max} is a hyperparameter defining the maximum learning rate in one epoch of learning. In cyclical learning, the learning rate for fine-tuning begins at the lower bound, rises to the upper bound, then descends back to the lower bound. A high learning rate midway through training acts as regularization, allowing the model to avoid overfitting and avoiding local optima. Lower learning rates at the beginning and end of cycles allow for optimization within a local optimum, giving the model an opportunity to discover fine-grained new information again. In our work, we set $lr_{max} = 0.00001$. A lower bound is then derived from the upper bound, $lr_{min} = 0.04 * lr_{max}$; this again is default behavior in the Fast.ai library.

We assess three different curricula for cyclical learning rates, visualized in Figure 1. In the default approach, a maximum learning rate is set and

[1] https://course.fast.ai/

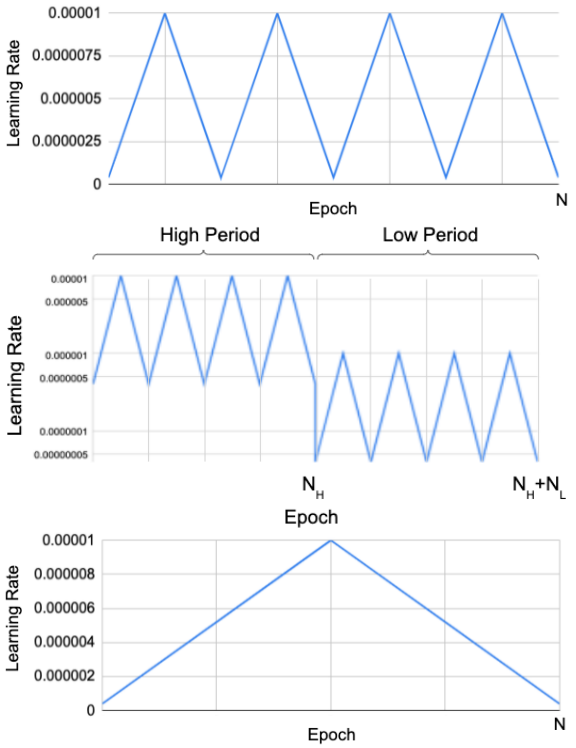

Figure 1: Illustration of cyclical (top), two-period cyclical (middle, log y-scale), and 1-cycle (bottom) learning rate curricula over *N* epochs.

cycles are repeated until reaching a threshold; for a halting criterion, we measure validation set accuracy. Because of noise in deep learning training, halting at *any* decrease can lead to premature stops; it is preferable to allow some occasional, small drop in performance. In our implementation we halt when accuracy on a validation set, measured in quadratic weighted kappa, decreases by over 0.01. In the second, "two-rate" approach (Smith, 2018), we follow this algorithm, but when we would halt, we instead backtrack by one epoch to a saved version of the network and restart training with a learning rate of 1×10^{-6} (one order of magnitude smaller). Finally, in the "1-cycle" policy, training is condensed into a single rise-and-fall pattern, spread over N epochs. Defining the exact training time N is a hyperparameter tuned on validation data. Finally, while BERT is optimized for sentence encoding, it is able to process documents up to 512 words long. In our data, we truncate a small number of essays longer than this maximum, mostly in ASAP dataset #2.

3.4 Feature Extraction from BERT

Fine-tuning is computationally expensive and can only run on GPU-enabled devices. Many prac-

titioners in low-resource settings may not have access to appropriate cloud computing environments for these techniques. Previous work has described a compromise approach for using Transformer models without fine-tuning. In Peters et al. (2019), the authors describe a new pipeline. Document texts are processed with an untuned BERT model; the final activations from network on the `[CLS]` token are then used directly as contextual word embeddings. This 768-dimensional feature vector represents the full document, and is used as inputs for a linear classifier. In the education context, a similar approach was described in Nadeem et al. (2019) as a baseline for evaluation of language-learner essays. This process allows us to use the world knowledge embedded in BERT without requiring fine-tuning of the model itself, and without need for GPUs at training or prediction time. For our work, we train a logistic regression classifier as described in Section 3.2.

3.5 DistilBERT

Recent work has highlighted the extreme carbon costs of full Transformer fine-tuning (Strubell et al., 2019) and the desire for Transformer-based prediction on-device without access to cloud compute. In response to these concerns, Sanh et al. (2019) introduce DistilBERT, which they argue is equivalent to BERT in most practical aspects while reducing parameter size by 40% to 66 million, and decreasing model inference time by 60%. This is accomplished using a distillation method (Hinton et al., 2015) in which a new, smaller "student" network is trained to reproduce the behavior of a pretrained "teacher" network. Once the smaller model is pretrained, interacting with it for the purposes of fine-tuning is identical to interacting with BERT directly. DistilBERT is intended for use cases where compute resources are a constraint, sacrificing a small amount of accuracy for a drastic shrinking of network size. Because of this intended use case, we only present results for DistilBERT with the "1-cycle" learning rate policy, which is drastically faster to fine-tune.

4 Experiments

To test the overall impact of fine-tuning in the AES domain, we use five datasets from the ASAP competition, jointly hosted by the Hewlett Foundation and Kaggle.com (Shermis, 2014). This set of essay prompts was the subject of intense public attention and scrutiny in 2012 and its public release has shaped the discourse on AES ever since. For our experiments, we use the original, deanonymized data from Shermis and Hamner (2012); an anonymized version of these datasets is available online[2]. In all cases, human inter-rater reliability (IRR) is an approximate upper bound on performance, but reliability above human IRR is possible, as all models are trained on *resolved* scores that represent two scores plus a resolution process for disagreements between annotators.

We focus our analysis on the five datasets that most closely resemble standard AES rubrics, discarding three datasets - prompts #1, 7, and 8 - with a scale of 10 or more possible points. Results on these datasets are not representative of overall performance and can skew reported results due to rubric idiosyncracies, making comparison to other published work impossible (see for example (Alikaniotis et al., 2016), which groups 60-point and 4-point rubrics into a single dataset and therefore produced correlations that cannot be aligned to results from any other published work). Prompts 2-6 are scored on smaller rubric scales with 4-6 points, and are thus generalizable to more AES contexts. Note that nevertheless, each dataset has its own idiosyncracies; for instance, essays in dataset #5 were written by younger students in 7th and 8th grade, while dataset #4 contains writing from high school seniors; datasets #3 and 4 were responses to specific texts while others were open-ended; and scores in dataset #2 was actually scored on two separate traits, the second of which is often discarded in followup work (as it is here). Our work here does not specifically isolate effects of these differences that would lead to discrepancies in performance or in modeling behavior.

4.1 Metrics and Baselines

For measuring reliability of automated assessments, we use a variant of Cohen's κ, with quadratic weights for "near-miss" predictions on an ordinal scale (QWK). This metric is standard in the AES community (Williamson et al., 2012). High-stakes testing organizations differ on exact cutoffs for acceptable performance, but threshold values between 0.6 and 0.8 QWK are typically used as a floor for testing purposes; human reliability below this threshold is generally not fit for summative student assessment.

[2]https://www.kaggle.com/c/asap-aes

In addition to measuring reliability, we also measure training and prediction time, in seconds. As this work seeks to evaluate the practical tradeoffs of the move to deep neural methods, this is an important secondary metric. For all experiments, training was performed on Google Colab Pro cloud servers with 32 GB of RAM and an NVideo Tesla P100 GPGPU.

We compare the results of BERT against several previously published benchmarks and results.

- Human IRR as initially reported in the Hewlett Foundation study (Shermis, 2014).
- Industry best performance, as reported by eight commercial vendors and one open-source research team in the initial release of the ASAP study. (Shermis, 2014).
- An early deep learning approach using a combination CNN+LSTM architecture that outperformed most reported results at that time (Taghipour and Ng, 2016).
- Two recent results using traditional non-neural models: Woods et al. (2017), which uses n-gram features in an ordinal logistic regression, and Cozma et al. (2018), which uses a mix of string kernels and word2vec embeddings in a support vector regression.
- Rodriguez et al. (2019), the one previously-published work that attempts AES with a variety of pretrained neural models, including BERT and the similar XLNet (Yang et al., 2019), with numerous alternate configurations and training methods. We report their result with a baseline BERT fine-tuning process, as well as their best-tuned model after extensive optimization.

4.2 Experimental Setup

Following past publications, we train a separate model on each dataset, and evaluate all dataset-specific models using 5-fold cross-validation. Each of the five datasets contains approximately 1,800 essays, resulting in folds of 360 essays each. Additionally, for measuring loss when fine-tuning BERT, we hold out an additional 20% of each training fold as a validation set, meaning that each fold has approximately 1,150 essays used for training and 300 essays used for validation. We report mean QWK across the five folds. For measurement of training and prediction time, we report the sum of training time across all five folds and all datasets. For slow-running feature extraction, like N-gram part-of-speech features and word embedding-based features, we tag each sentence in the dataset only once and cache the results, rather than re-tagging each sentence on each fold. Finally, for models where distinguishing extraction from training time is meaningful, we present those times separately.

5 Results

5.1 Accuracy Evaluation

Our primary results are presented in Table 1. We find, broadly, that all approaches to machine learning replicate human-level IRR as measured by QWK. Nearly eight years after the publication of the original study, no published results have exceeded vendor performance on three of the five prompt datasets; in all cases, a naive N-gram approach underperforms the state-of-the-art in industry and academia by 0.03-0.06 QWK.

Of particular note is the low performance of GloVe embeddings relative to either neural or N-gram representations. This is surprising: while word embeddings are less popular now than deep neural methods, they still perform well on a wide range of tasks (Baroni et al., 2014). Few publications have noted this negative result for GloVe in the AES domain; only Dong et al. (2017) uses GloVe as the primary representation of ASAP texts in an LSTM model, reporting lower QWK results than any baseline we presented here. One simple explanation for this may be that individual keywords matter a great deal for model performance. It is well established that vocabulary-based approaches are effective in AES tasks (Higgins et al., 2014) and the lack of access to specific word-based features may hinder semantic vector representation. Indeed, only one competitive recent paper on AES uses non-contextual word vectors: Cozma et al. (2018). In this implementation, they do use word2vec, but rather than use word embeddings directly they first cluster words into a set of 500 "embedding clusters." Words that appear in texts are then counted in the feature vector as the centroid of that cluster - in effect, creating a 500-dimensional bag-of-words model.

Our results would suggest that fine-tuning with BERT also reaches approximately the same level

Table 1: Performance on each of ASAP datasets 2-6, in QWK. The final row shows the gap in QWK between the best neural model and the N-gram baseline.

Model	**2**	**3**	**4**	**5**	**6**
Human IRR	.80	.77	.85	.74	.74
Hewlett	**.74**	**.75**	.82	**.83**	.78
Taghipour	.69	.69	.81	.81	.82
Woods	.71	.71	.81	.82	**.83**
Cozma	.73	.68	**.83**	**.83**	**.83**
Rodriguez (BERT)	.68	.72	.80	.81	.81
Rodriguez (best)	.70	.72	.82	.82	.82
N-Grams	.71	.71	.78	.80	.79
Embeddings	.42	.41	.60	.49	.36
BERT-CLR	.66	.70	.80	.80	.79
BERT-1CYC	.64	.71	.82	.81	.79
BERT Features	.61	.59	.75	.75	.74
DistilBERT	.65	.70	.82	.81	.79
N-Gram Gap	-.05	.00	.04	.01	.00

of performance as other methods, slightly underperforming previous published results. This is likely an underestimate, due to the complexity of hyperparameter optimization and curriculum learning for Transformers. Rodriguez et al. (2019) demonstrate that it is, in fact, possible to improve the performance of neural models to more closely approach (but not exceed) the state-of-the-art using neural models. Sophisticated approaches like gradual unfreezing, discriminative fine-tuning, or greater parameterization through newer deep learning models in their work consistently produces improvements of 0.01-0.02 QWK compared to the default BERT implementation. But this result emphasizes our concern: we do not claim our results are the best that could be achieved with BERT fine-tuning. We are, in fact, confident that they can be improved through optimization. What the results demonstrate instead is that the ceiling of results for AES tasks lessens the value of that intensive optimization effort.

5.2 Runtime Evaluation

Our secondary evaluation of models is based on training time and resource usage; those results are reported in Table 2. Here, we see that deep learning approaches on GPU-enabled cloud compute produce an approximately 30-100 fold increase in end-to-end training time compared to a naive approach. In fact, this understates the gap, as approximately 75% of feature extraction and model training time in the naive approach is due to part-of-speech tagging rather than learning. Using BERT features as inputs to a linear classifier is an interesting compromise option, producing slightly lower performance on these datasets but with only a 2x slowdown at training time, all in feature extraction, and potentially retaining some of the semantic knowledge of the full BERT model. Further investigation should test whether additional features for intermediate layers, as explored in Peters et al. (2019), is merited for AES.

Table 2: Cumulative experiment runtime, in seconds, of feature extraction (F), model training (T), and predicting on test sets (P), for ASAP datasets 2-6 with 5-fold cross-validation. Models with 1-cycle fine-tuning are measured at 5 epochs.

Model	**F**	**T**	**P**	**Total**
Embeddings	93	6	1	100
N-Grams	82	27	2	111
BERT Features	213	10	1	224
DistilBERT	1,972		108	2,080
BERT-1CYC	2,956		192	3,148
BERT-CLR	11,309		210	11,519

We can look at this gap in training runtime more closely in Figure 2. Essays in the prompt 2 dataset are longer persuasive essays and are on average 378 words long, while datasets 3-6 correspond to shorter, source-based content knowledge prompts and are on average 98-152 words long. The need for truncation in dataset #2 for BERT, but not for other approaches, may explain the underperformance of the model in that dataset. Additionally, differences across datasets highlight two key differences for fine-tuning a BERT model:

- Training time increases linearly with number of epochs and with average document length. As seen in Figure 2, this leads to a longer training for the longer essays of dataset 2, nearly as long as the other datasets combined.

- Performance converges on human inter-rater reliability more quickly for short content-based prompts, and performance begins to decrease due to overfitting in as few as 4 epochs. By comparison, in the longer, persuasive arguments of dataset 2, very small performance gains on held-out data continued even at the end of our experiments.

Figure 2 also presents results for DistilBERT. Our work verifies prior published claims of speed

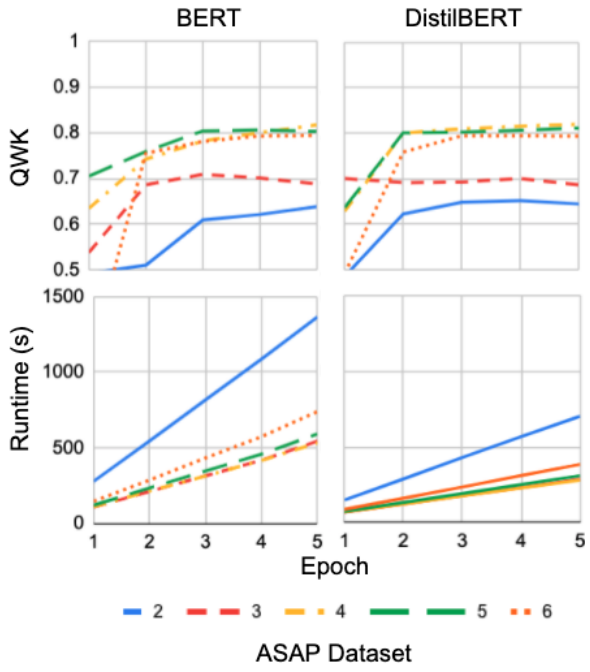

Figure 2: QWK (top) and training time (bottom, in seconds) and for 5-fold cross-validation of 1-cycle neural fine-tuning on ASAP datasets 2-6, for BERT (left) and DistilBERT (right).

improvements both in fine-tuning and at prediction time, relative to the baseline BERT model: training time was reduced by 33% and prediction time was reduced by 44%. This still represents at least a 20x increase in runtime relative to N-gram baselines both for training and prediction.

6 Discussion

For scoring essays with reliably scored, prompt-specific training sets, both classical and neural approaches produce similar reliability, at approximately identical levels to human inter-rater reliability. There is a substantial increase in technical overhead required to implement Transformers and fine-tune them to reach this performance, with minimal gain compared to baselines. The policy lesson for NLP researchers is that using deep learning for scoring alone is unlikely to be justifiable, given the slowdowns at both training and inference time, and the additional hardware requirements. For scoring, at least, Transformer architectures are a hammer in search of a nail.

But it's hardly the case that automated writing evaluation is limited to scoring. In this section we cover major open topics for technical researchers in AES to explore, focusing on areas where neural models have proven strengths above baselines in other domains. We prioritize three major areas: *domain transfer*, *style*, and *fairness*. In each we cite specific past work that indicates a plausible path forward for research.

6.1 Domain Transfer

A major challenge in AES is the inability of prompt-specific models to generalize to new essay topics (Attali et al., 2010; Lee, 2016). Collection of new prompt-specific training sets, with reliable scores, continues to be one of the major stumbling blocks to expansion of AES systems in curricula (Woods et al., 2017). Relatively few researchers have made progress on generic essay scoring: Phandi et al. (2015) introduces a Bayesian regression approach that extracts N-gram features then capitalizes on correlated features across prompts. Jin et al. (2018) shows promising prompt-independent results using an LSTM architecture with surface and part-of-speech N-gram inputs, underperforming prompt-specific models by relatively small margins across all ASAP datasets. But in implementations, much of the work of practitioners is based on workarounds for prompt-specific models; Wilson et al. (2019), for instance, describes psychometric techniques for measuring generic writing ability across a small sample of known prompts.

While Transformers *are* sensitive to the data they were pretrained on, they are well-suited to transfer tasks in mostly unseen domains, as evidenced by part-of-speech tagging for historical texts (Han and Eisenstein, 2019), sentiment classification on out-of-domain reviews (Myagmar et al., 2019), and question answering in new contexts (Houlsby et al., 2019). This last result is promising for content-based short essay prompts, in particular. Our field's open challenge in scoring is to train AES models that can meaningfully evaluate short response texts for correctness based on world knowledge and domain transfer, rather than memorizing the vocabulary of correct, in-domain answers. Promising early results show that relevant world knowledge is *already* embedded in BERT's pretrained model (Petroni et al., 2019). This means that BERT opens up a potentially tractable path to success that was simply not possible with N-gram models.

6.2 Style and Voice

Skepticism toward AES in the classroom comes from rhetoric and composition scholars, who ex-

press concerns about its role in writing pedagogy (NCTE, 2013; Warner, 2018). Indeed, the relatively "solved" nature of summative scoring that we highlight here is of particular concern to these experts, noting the high correlation between scores and features like word count (Perelman, 2014).

Modern classroom use of AES beyond high-stakes scoring, like *Project Essay Grade* (Wilson and Roscoe, 2019) or Turnitin *Revision Assistant* (Mayfield and Butler, 2018), makes claims of supporting student agency and growth; here, adapting to writer individuality is a major current gap. Dixon-Román et al. (2019) raises a host of questions about these topics specifically in the context of AES, asking how algorithmic intervention can produce strong writers rather than merely good essays: *"revision, as adjudicated by the platform, is [...] a re-direction toward the predetermined shape of the ideal written form [...] a puzzle-doer recursively consulting the image on the puzzle-box, not that of author returning to their words to make them more lucid, descriptive, or forceful."*

This critique is valid: research on machine translation, for instance, has shown that writer style is not preserved across languages (Rabinovich et al., 2017). There is uncharted territory for AES to adapt to individual writer styles and give feedback based on *individual* writing rather than prompt-specific exemplars. Natural language understanding researchers now argue that *"...style is formed by a complex combination of different stylistic factors"* (Kang and Hovy, 2019); Style-specific natural language generation has shown promise in other domains (Hu et al., 2017; Prabhumoye et al., 2018) and has been extended not just to individual preferences but also to overlapping identities based on attitudes like sentiment and personal attributes like gender (Subramanian et al.). Early work suggests that style-specific models *do* see major improvements when shifting to high-dimensionality Transformer architectures (Keskar et al., 2019). This topic bridges an important gap: for assessment, research has shown that "authorial voice" has measurable outcomes on writing impact (Matsuda and Tardy, 2007), while individual expression is central to decades of pedagogy (Elbow, 1987). Moving the field toward individual expression and away from prompt-specific datasets may be a path to lending legitimacy to AES, and Transformers may be the technical leap necessary to make these tasks work.

6.3 Fairness

Years ago, researchers suggested that demographic bias is worth checking in AES systems (Williamson et al., 2012). But years later, the field has primarily reported fairness experiments on simulated data, and shared toolkits for measuring bias, rather than results on real-world AES implementations or high-stakes data (Madnani et al., 2017; Loukina et al., 2019).

Prompted by social scientists (Noble, 2018), NLP researchers have seen a renaissance of fairness research based on the flaws in default implementations of Transformers (Bolukbasi et al., 2016; Zhao et al., 2017, 2018). These works typicallly seek to reduce the amplification of bias in pretrained models, starting with easy-to-measure proof that demographic bias can be "removed" from word embedding spaces. But iterating on inputs to algorithmic classifiers – precisely the intended use case of formative eeedback for writers! – can reduce the efficacy of "de-biasing" (Liu et al., 2018; Dwork and Ilvento, 2019). More recent research has shown that bias may simply be masked by these approaches, rather than resolved (Gonen and Goldberg, 2019).

What these questions offer, though, is a wellspring of new and innovative technical research. Developers of learning analytics software, including AES, are currently encouraged to focus on scalable experimental evidence of efficacy for learning outcomes (Saxberg, 2017), rather than focus on specific racial or gender bias, or other equity outcomes that are more difficult to achieve through engineering. But Transformer architectures are nuanced enough to capture immense world knowledge, producing a rapid increase in explainability in NLP (Rogers et al., 2020).

Meanwhile, in the field of learning analytics, a burgeoning new field of fairness studies are learning how to investigate these issues in algorithmic educational systems (Mayfield et al., 2019; Holstein and Doroudi, 2019). Outside of technology applications but in writing assessment more broadly, fairness is also a rich topic with a history of literature to learn from (Poe and Elliot, 2019). Researchers at the intersection of *both* these fields have an enormous open opportunity to better understand AES in the context fairness, using the latest tools not just to build reliable scoring but to advance social change.

References

Dimitrios Alikaniotis, Helen Yannakoudakis, and Marek Rei. 2016. Automatic text scoring using neural networks. In *Proceedings of the Association for Computational Linguistics*, pages 715–725.

Yigal Attali. 2013. Validity and reliability of automated essay scoring. *Handbook of automated essay evaluation: Current applications and new directions*, page 181.

Yigal Attali, Brent Bridgeman, and Catherine Trapani. 2010. Performance of a generic approach in automated essay scoring. *The Journal of Technology, Learning and Assessment*, 10(3).

Yigal Attali and Jill Burstein. 2004. Automated essay scoring with e-rater® v. 2.0. *ETS Research Report Series*, (2).

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations*.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the Association for Computational Linguistics*, pages 238–247.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4349–4357. Curran Associates, Inc.

Jing Chen, James H Fife, Isaac I Bejar, and André A Rupp. 2016. Building e-rater® scoring models using machine learning methods. *ETS Research Report Series*, 2016(1):1–12.

Mădălina Cozma, Andrei Butnaru, and Radu Tudor Ionescu. 2018. Automated essay scoring with string kernels and word embeddings. In *Proceedings of the Association for Computational Linguistics*.

Zihang Dai, Zhilin Yang, Yiming Yang, William W Cohen, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the Association for Computational Linguistics*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the NAACL HLT Conference*.

Ezekiel Dixon-Román, T. Philip Nichols, and Ama Nyame-Mensah. 2019. The racializing forces of/in ai educational technologies. *Learning, Media & Technology Special Issue on AI and Education: Critical Perspectives and Alternative Futures*.

Fei Dong, Yue Zhang, and Jie Yang. 2017. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the Conference on Computational Natural Language Learning*, pages 153–162.

Cynthia Dwork and Christina Ilvento. 2019. Fairness under composition. In *Innovations in Theoretical Computer Science Conference*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Peter Elbow. 1987. Closing my eyes as i speak: An argument for ignoring audience. *College English*, 49(1):50–69.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9(Aug):1871–1874.

James Fiacco, Elena Cotos, and Carolyn Rosé. 2019. Towards enabling feedback on rhetorical structure with neural sequence models. In *Proceedings of the International Conference on Learning Analytics & Knowledge*, pages 310–319. ACM.

Peter W Foltz, Sara Gilliam, and Scott Kendall. 2000. Supporting content-based feedback in on-line writing evaluation with lsa. *Interactive Learning Environments*, 8(2):111–127.

Hila Gonen and Yoav Goldberg. 2019. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. *Proceedings of the Association for Computational Linguistics*.

Xiaochuang Han and Jacob Eisenstein. 2019. Unsupervised domain adaptation of contextualized embeddings: A case study in early modern english. *arXiv preprint arXiv:1904.02817*.

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*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *stat*, 1050:9.

Kenneth Holstein and Shayan Doroudi. 2019. Fairness and equity in learning analytics systems (fairlak). in. In *Companion Proceedings of the International Learning Analytics & Knowledge Conference (LAK 2019)*.

Matthew Honnibal and Ines Montani. 2017. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *Proceedings of the International Conference on Machine Learning*, pages 2790–2799.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *Proceedings of the International Conference on Machine Learning*, pages 1587–1596.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does bert learn about the structure of language? In *Proceedings of the Association for Computational Linguistics*, pages 3651–3657.

Cancan Jin, Ben He, Kai Hui, and Le Sun. 2018. Tdnn: a two-stage deep neural network for prompt-independent automated essay scoring. In *Proceedings of the Association for Computational Linguistics*, pages 1088–1097.

Dan Jurafsky and James H Martin. 2014. *Speech and language processing*, volume 3. Pearson London.

Dongyeop Kang and Eduard Hovy. 2019. xslue: A benchmark and analysis platform for cross-style language understanding and evaluation. *arXiv preprint arXiv:1911.03663*.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of bert. In *Proceedings of Empirical Methods in Natural Language Processing*, volume 1, pages 2465–2475.

Yong-Won Lee. 2016. Investigating the feasibility of generic scoring models of e-rater for toefl ibt independent writing tasks. *English Language Teaching*, 71.

Lydia T Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. 2018. Delayed impact of fair machine learning. In *Proceedings of the International Conference on Machine Learning*.

Anastassia Loukina, Nitin Madnani, and Klaus Zechner. 2019. The many dimensions of algorithmic fairness in educational applications. In *Proceedings of the ACL Workshop on Innovative Use of NLP for Building Educational Applications*.

Nitin Madnani, Anastassia Loukina, Alina von Davier, Jill Burstein, and Aoife Cahill. 2017. Building better open-source tools to support fairness in automated scoring. In *Proceedings of the ACL Workshop on Ethics in Natural Language Processing*, pages 41–52.

Paul Kei Matsuda and Christine M Tardy. 2007. Voice in academic writing: The rhetorical construction of author identity in blind manuscript review. *English for Specific Purposes*, 26(2):235–249.

Elijah Mayfield and Stephanie Butler. 2018. Districtwide implementations outperform isolated use of automated feedback in high school writing. In *Proceedings of the International Conference of the Learning Sciences (Industry and Commercial Track)*.

Elijah Mayfield, Michael Madaio, Shrimai Prabhumoye, David Gerritsen, Brittany McLaughlin, Ezekiel Dixon-Román, and Alan W Black. 2019. Equity beyond bias in language technologies for education. In *Proceedings of ACL Workshop on Innovative Use of NLP for Building Educational Applications*.

Tristan Miller. 2003. Essay assessment with latent semantic analysis. *Journal of Educational Computing Research*, 29(4):495–512.

Batsergelen Myagmar, Jie Li, and Shigetomo Kimura. 2019. Transferable high-level representations of bert for cross-domain sentiment classification. In *Proceedings on the International Conference on Artificial Intelligence (ICAI)*, pages 135–141.

Farah Nadeem, Huy Nguyen, Yang Liu, and Mari Ostendorf. 2019. Automated essay scoring with discourse-aware neural models. In *Proceedings of the ACL Workshop on Innovative Use of NLP for Building Educational Applications*, pages 484–493.

NCTE. 2013. Position statement on machine scoring. *National Council of Teachers of English*. Accessed 2019-09-24.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The conll-2014 shared task on grammatical error correction. In *Proceedings of the Conference on Computational Natural Language Learning: Shared Task*, pages 1–14.

Huy V Nguyen and Diane J Litman. 2018. Argument mining for improving the automated scoring of persuasive essays. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Safiya Umoja Noble. 2018. *Algorithms of oppression: How search engines reinforce racism*. NYU Press.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn:

Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of Empirical Methods in Natural Language Processing*, pages 1532–1543.

Les Perelman. 2014. When "the state of the art" is counting words. *Assessing Writing*, 21:104–111.

Matthew E Peters, Sebastian Ruder, and Noah A Smith. 2019. To tune or not to tune? adapting pretrained representations to diverse tasks. In *Proceedings of the Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 7–14.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of Empirical Methods in Natural Language Processing*, pages 2463–2473.

Peter Phandi, Kian Ming A Chai, and Hwee Tou Ng. 2015. Flexible domain adaptation for automated essay scoring using correlated linear regression. In *Proceedings of Empirical Methods in Natural Language Processing*, pages 431–439.

Mya Poe and Norbert Elliot. 2019. Evidence of fairness: Twenty-five years of research in assessing writing. *Assessing Writing*, 42:100418.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *Proceedings of the Association for Computational Linguistics*, pages 866–876.

Ella Rabinovich, Raj Nath Patel, Shachar Mirkin, Lucia Specia, and Shuly Wintner. 2017. Personalized machine translation: Preserving original author traits. In *Proceedings of the European Chapter of the Association for Computational Linguistics*, pages 1074–1084.

Y Malini Reddy and Heidi Andrade. 2010. A review of rubric use in higher education. *Assessment & evaluation in higher education*, 35(4):435–448.

Brian Riordan, Michael Flor, and Robert Pugh. 2019. How to account for mispellings: Quantifying the benefit of character representations in neural content scoring models. In *Proceedings of the ACL Workshop on Innovative Use of NLP for Building Educational Applications*, pages 116–126.

Pedro Uria Rodriguez, Amir Jafari, and Christopher M Ormerod. 2019. Language models and automated essay scoring. *arXiv preprint arXiv:1909.09482*.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in bertology: What we know about how bert works. *arXiv preprint arXiv:2002.12327*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Bror Saxberg. 2017. Learning engineering: the art of applying learning science at scale. In *Proceedings of the ACM Conference on Learning@ Scale*. ACM.

Mark D Shermis. 2014. State-of-the-art automated essay scoring: Competition, results, and future directions from a united states demonstration. *Assessing Writing*, 20:53–76.

Mark D Shermis and Ben Hamner. 2012. Contrasting state-of-the-art automated scoring of essays: Analysis. In *Annual national council on measurement in education meeting*, pages 14–16.

Leslie N Smith. 2018. A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay. *arXiv preprint arXiv:1803.09820*.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in nlp. *Proceedings of the Association for Computational Linguistics*.

Sandeep Subramanian, Guillaume Lample, Eric Michael Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. Multiple-attribute text style transfer. *Age*, 18(24):65.

Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of Empirical Methods in Natural Language Processing*, pages 1882–1891.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline. In *Proceedings of the Association for Computational Linguistics*.

Sowmya Vajjala. 2018. Automated assessment of non-native learner essays: Investigating the role of linguistic features. *International Journal of Artificial Intelligence in Education*, 28(1):79–105.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Xinhao Wang, Keelan Evanini, and Klaus Zechner. 2013. Coherence modeling for the automated assessment of spontaneous spoken responses. In *Proceedings of the North American chapter of the Association for Computational Linguistics: Human language technologies*, pages 814–819.

John Warner. 2018. *Why They Can't Write: Killing the Five-Paragraph Essay and Other Necessities*. JHU Press.

David M Williamson, Xiaoming Xi, and F Jay Breyer. 2012. A framework for evaluation and use of automated scoring. *Educational measurement: issues and practice*, 31(1):2–13.

Joshua Wilson, Dandan Chen, Micheal P Sandbank, and Michael Hebert. 2019. Generalizability of automated scores of writing quality in grades 3–5. *Journal of Educational Psychology*, 111(4):619.

Joshua Wilson and Rod D Roscoe. 2019. Automated writing evaluation and feedback: Multiple metrics of efficacy. *Journal of Educational Computing Research*.

Bronwyn Woods, David Adamson, Shayne Miel, and Elijah Mayfield. 2017. Formative essay feedback using predictive scoring models. In *Proceedings of ACM Conference on Knowledge Discovery and Data Mining*, pages 2071–2080. ACM.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.

Helen Yannakoudakis and Ted Briscoe. 2012. Modeling coherence in esol learner texts. In *Proceedings of the ACL Workshop on Building Educational Applications Using NLP*, pages 33–43. Association for Computational Linguistics.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2017. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of Empirical Methods in Natural Language Processing*, pages 2979–2989.

Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018. Learning gender-neutral word embeddings. In *Proceedings of Empirical Methods in Natural Language Processing*, pages 4847–4853.