

What Does BERT Know about Linguistic Features: Probing Finnish Model for Text Difficulty

Yiheng Wu¹, Jue Hou¹, Roman Yangarber¹

¹University of Helsinki
first.last@helsinki.fi

Abstract

We observe that fine-tuned BERT outperforms feature-based models at estimation of text difficulty, but it is unclear what drives this advantage. We explore a state-of-the-art Finnish BERT difficulty estimator and analyze it against a strong feature-based model trained on 66 hand-crafted linguistic features. First, we find that BERT linearly encodes the coarse length and complexity features, but not fine-grained morphology of Finnish. Second, BERT subsumes the information in the hand-crafted features and adds signal that they miss: the two models are evenly matched on most documents, and BERT’s advantage comes from rarely being very wrong, rather than from winning everywhere. Third, through input perturbation and layer-wise probing, we show that BERT’s predictions rely on sentence-internal structure rather than on lexical content alone. Our analysis explains why a neural difficulty estimator succeeds and what linguistic information it depends on.

1 Introduction

The task of estimation of text difficulty can be cast as predicting the proficiency level required to comprehend a text, e.g., on the CEFR scale (Council of Europe, 2001). Difficulty estimation supports applications ranging from graded reading and curriculum design to second-language learning and assessment. The task is particularly challenging for morphologically complex languages such as Finnish, where a single lemma can surface in a large number of inflected forms and where meaning is heavily affected by morphology.

Two paradigms dominate among the approaches to this task. The first builds interpretable models on hand-crafted linguistic features, such as measures of length, lexical richness, morphology, and syntactic complexity, paired with classical machine learning. The second paradigm fine-tunes end-to-end neural encoders, such as BERT (Devlin et al.,

2019), which typically achieve higher accuracy, but operate as a *black box*. For Finnish difficulty estimation, the current state-of-the-art (SOTA) is a fine-tuned Finnish BERT encoder (Wu et al., 2026a; Virtanen et al., 2019), which outperforms feature-based models. However, it is not understood why the neural model is more accurate, or what information it relies on.

This gap matters. It is generally observed that fine-tuned encoders surpass feature-based systems, yet it is scarcely established whether they rediscover the same linguistic properties that decades of feature engineering have identified, or whether they exploit additional signal beyond those features. The question is especially poignant for morphologically rich languages. It is unclear whether BERT implicitly recovers Finnish-specific morphological cues, or whether its advantage comes from coarser, surface-level information. To answer this, we must look inside the model, rather than only at its scores.

In this paper, we look inside the black box of the SOTA Finnish Document-level BERT difficulty estimator. Our contribution is analytical. We do not introduce a new difficulty model. Rather, we probe an existing one against a strong feature-based baseline built on 66 hand-crafted linguistic features. We organize the study around three research questions.

RQ1: Does BERT implicitly encode the hand-crafted features? We address this with linear probes and control tasks (Alain and Bengio, 2017; Conneau et al., 2018; Hewitt and Liang, 2019).

RQ2: Does BERT pick up difficulty signal beyond the hand-crafted features? We test this through error complementarity and bidirectional residual prediction, with paired-bootstrap significance testing (Berg-Kirkpatrick et al., 2012).

RQ3: What linguistic information drives BERT’s predictions? We use feature alignment, input perturbation, and layer-wise probing to separate surface-lexical from structural cues.

Our analysis yields a consistent picture. BERT

largely subsumes the hand-crafted features. It recovers most of them and explains a substantial share of the feature model’s residual errors, while the features carry essentially no signal that BERT misses. BERT also relies primarily on coarse length, complexity, and lexical-richness cues rather than on fine-grained morphological cues. Our contributions are:

- methodology for *interpretability*: comparing a fine-tuned encoder against hand-crafted linguistic features on a regression task;
- empirical evidence that BERT strictly subsumes features for Finnish text difficulty estimation;
- characterization of the linguistic level at which BERT’s difficulty judgments operate.

2 Related Work

Text difficulty estimation: Estimating the difficulty of written text, also referred to as readability, proficiency, or grade-level assessment, has a long history in educational research and NLP. Early formula-based approaches such as Flesch-Kincaid and the Lexile framework provide simple numeric scores (Kincaid et al., 1975; Stenner, 1996), but rely on surface features and miss deeper lexical or syntactic complexity. Supervised systems then incorporated richer hand-crafted features, including parse depth, grammatical constructions, and word-frequency lists (Collins-Thompson and Callan, 2004; Vajjala and Meurers, 2012; Lashina et al., 2018).

More recent neural approaches, from hierarchical attention networks (Azpiazu and Pera, 2019) to fine-tuned BERT models (Martinc et al., 2021), substantially outperform feature-based baselines, and Transformer-based models have been compared against feature-engineered systems for several languages (Sharoff, 2022). Large language models have also been applied to readability with competitive results. For Finnish, we build on existing annotated resources (Dmitrieva and Konovalova, 2023; Katinskaia et al., 2025). This work demonstrated that neural models win, but did not explain why, which is the question we take up.

Analysing what neural models encode: A large body of literature analyzes what fine-tuned encoders, such as BERT, represent internally (Rogers et al., 2020). A common tool is the probing classifier, which trains a simple model to recover a lin-

guistic property from a frozen representation (Alain and Bengio, 2017; Conneau et al., 2018). Probing has shown that contextual representations encode syntactic structure (Hewitt and Manning, 2019) and that linguistic information is distributed across layers in an interpretable way (Tenney et al., 2019). Because expressive probes can fit arbitrary targets, control tasks and selectivity were introduced to ensure that a probe reflects the representation rather than its own capacity (Hewitt and Liang, 2019; Belinkov, 2022). Complementary *interventional* approaches perturb the input, e.g., by shuffling word order, to test what models actually rely on (Sinha et al., 2021; Pham et al., 2021). In our analysis, we adopt probes with control tasks, input perturbation, and layer-wise probing.

Positioning: Most probing studies ask whether a representation encodes certain generic linguistic properties in the context of analyzing a single model. In contrast, we compare a fine-tuned BERT difficulty model against a set of hand-crafted features used by a competing model on the same task, and quantify the relationship between them: whether BERT recovers those features, whether it captures a signal that the features miss, and what linguistic information drives its predictions. To our knowledge this kind of feature-grounded comparison has not been carried out for text difficulty estimation, and we do so for a morphologically rich language, Finnish, where the contrast between coarse and fine-grained features is especially informative.

3 Method

3.1 Experimental Setup

We study why a fine-tuned BERT encoder outperforms a strong feature-based model on text difficulty estimation. We treat the task as continuous regression onto the CEFR scale, where the scores range from 1.0 (A1) to 6.0 (C2) in steps of 0.5. The mapping between scores and CEFR levels is given in Table 1. The details of the dataset (Wu et al., 2026b) is shown in Table 2,

The BERT model¹ is fine-tuned with a regression head. The feature-based model is a Random Forest regressor trained on 66 hand-crafted linguistic features. These features cover length, lexical richness, the rich case system, verbal morphology, derivation, compounding, and syntactic complexity,

¹bert-base-finnish-cased (Virtanen et al., 2019)

Score	CEFR	Score	CEFR
1	A1	4	B2
1.5	A1-A2	4.5	B2-C1
2	A2	5	C1
2.5	A2-B1	5.5	C1-C2
3	B1	6	C2
3.5	B1-B2		

Table 1: Mapping from continuous regression score to CEFR levels. CEFR levels map to integers; half-levels (e.g., 1.5) denote “intermediate” levels.

Level	Train	Dev	Test
A1	241	27	0
A1-A2	342	32	32
A2	595	90	98
A2-B1	75	13	0
B1	2,820	320	60
B1-B2	1,836	248	183
B2	772	95	48
B2-C1	0	0	0
C1	493	69	59
C1-C2	976	102	340
C2	143	15	45
Total	8,293	1,011	865

Table 2: Distribution of instances across CEFR levels by data split.

and follow standard descriptive accounts of Finnish grammar (Hakulinen et al., 2004; Karlsson, 2018). The details of features are listed in Appendix A. We extract them from the annotations produced by the Turku Neural Parser Pipeline (Kanerva et al., 2018), which provides tokenization, morphological tags, lemmas, and dependency trees.

For a document i , let $\mathbf{h}_i$ denote BERT’s [CLS] representation—the embedding vector that feeds the regression head; $\hat{y}_i^B$ is BERT’s prediction; $\hat{y}_i^F$ the feature-based model’s prediction; $\mathbf{x}_i$ the feature vector; and y_i the gold-standard CEFR score. All probes and auxiliary regressors are fitted on the training split and evaluated on a held-out test split.

Model performance: Both models perform well on the test set, but BERT outperforms the feature model. We report mean absolute error (MAE) and root mean squared error (RMSE); RMSE penalizes large errors more heavily than MAE, so a gap between the two indicates the presence of occasional large mistakes. BERT reaches MAE of 0.0824 and RMSE of 0.2055, compared with 0.2111 and 0.4211 for the feature-based model. Figures 1 and 2 show, for each true CEFR level, the distributions of the models’ predictions. Both models track the ground-truth levels well, but the feature model pro-

Model	MAE	RMSE
BERT	0.0824	0.2055
Feature model	0.2111	0.4211

Table 3: Test-set performance of the BERT vs. the feature-based model. Lower is better for both metrics.

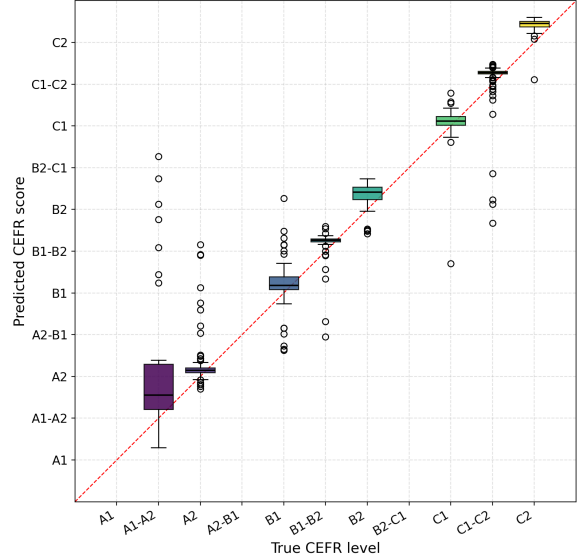


Figure 1: BERT: distribution of predicted scores for each true CEFR level. The red dashed mark indicates the ideal prediction (equal to the true level).

duces wider spreads and more outliers, which is consistent with its higher error.

3.2 RQ1: Does BERT implicitly encode the hand-crafted features?

We test whether information targeted by feature engineering is implicitly present in BERT’s embedding representation.

Linear probes. For each hand-crafted feature j , we fit a ridge linear probe that maps the representation $\mathbf{h}_i$ to the value $\hat{x}_i^{(j)}$ of the feature j :

$$\hat{x}_i^{(j)} = \mathbf{w}_j^\top \mathbf{h}_i + b_j, \text{ where}$$

$$\mathbf{w}_j = \arg \min_{\mathbf{w}} \sum_i (x_i^{(j)} - \mathbf{w}^\top \mathbf{h}_i)^2 + \lambda \|\mathbf{w}\|_2^2. \quad (1)$$

The probe is fit on the training split and evaluated on the test split, for each of the 66 features, following the linear-probing paradigm (Alain and Bengio, 2017; Conneau et al., 2018; Hewitt and Manning, 2019). Keeping the probe linear is deliberate. A high-capacity probe could recover a feature through its own computation rather than because the embedding representation encoded it, but if a simple

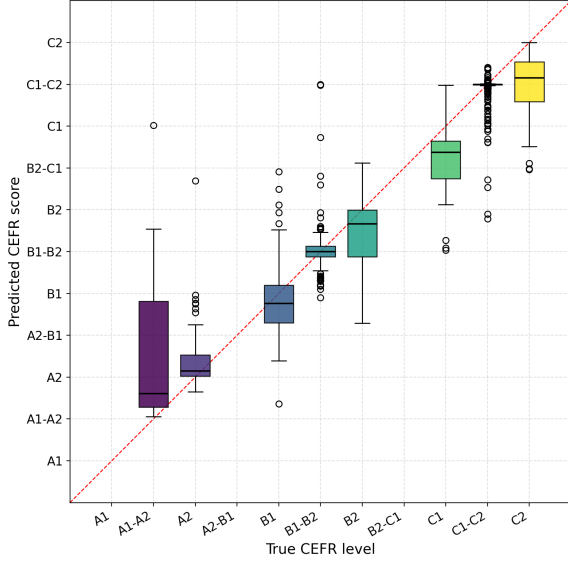


Figure 2: Feature-based model: distribution of predicted scores for each true CEFR level, shown on the same axes as Figure 1 for comparison.

linear model accurately predicts the value of a feature based on the embedding, then we can be more confident that the embedding itself in fact encodes the feature. We use ridge regression for its simplicity (Hewitt and Liang, 2019); the L_2 penalty λ , selected by generalized cross-validation, stabilises the estimate, and the same probe is used for every feature so that scores are directly comparable.

Recovery metrics. Two metrics measure how well a feature is recovered. R^2 measures absolute recovery,

$$R_j^2 = 1 - \frac{\sum_i (x_i^{(j)} - \hat{x}_i^{(j)})^2}{\sum_i (x_i^{(j)} - \bar{x}^{(j)})^2}, \quad (2)$$

where $\bar{x}^{(j)}$ is the mean value of feature j . The denominator is the error of predicting that mean, so $R^2 = 0$ means the probe is no better than this baseline and $R^2 = 1$ means exact recovery. We rely on R^2 as the main metric because it is scale-free: the 66 features have very different units (for example, sentence counts versus case ratios), and R^2 normalises each onto the same axis, so recovery scores are comparable across features. An absolute-error metric such as MAE would not be comparable, since an error of 0.5 means something different for each feature. Pearson r between $x^{(j)}$ and $\hat{x}^{(j)}$ measures whether the probe captures the correct trend, irrespective of magnitude. Reporting it alongside R^2 is informative, since a feature with low R^2 but

high r is one for which BERT recovers its direction, but not its absolute scale.

Control task. The third metric checks that a high R^2 reflects genuinely encoded information rather than the probe’s own capacity. Following the control-task method (Hewitt and Liang, 2019), we re-fit the same probe on a control target in which each feature’s values are randomly permuted across documents—we preserve the feature’s marginal distribution while destroying its relationship to the representation. This is averaged over several permutations.

3.3 RQ2: Does BERT pick up difficulty signal beyond the hand-crafted features?

We first establish that BERT’s advantage over the feature-based model is statistically significant, and then test whether each model picks up difficulty signals that the other misses.

Significance of the difference between models:

The performance of two models is similar, so we test whether BERT’s advantage is significant using paired bootstrap over the test set (Berg-Kirkpatrick et al., 2012). We draw $B = 10,000$ resamples, each obtained by sampling the test documents with replacement. On resample b we recompute the difference between the metrics for the two models,

$$\delta_b = M(\mathbf{y}_b, \hat{\mathbf{y}}_b^B) - M(\mathbf{y}_b, \hat{\mathbf{y}}_b^F), \quad (3)$$

where M is one of our metrics and both models are scored on the same resample b ; superscripts B and F denote BERT and the feature-based model. We report the observed difference, the 95% percentile confidence interval of $\{\delta_b\}$, and a two-sided bootstrap p -value, separately for the three metrics QWK, R^2 , and MAE. Because both models are evaluated on the same resampled documents in each iteration, the comparison controls for document-level difficulty and isolates each model’s per-document advantage. We report three metrics capturing different aspects of accuracy to avoid relying on any single one.

Complementarity via residual prediction. A model’s residual is the part of the ground-truth score it fails to predict. If a second information source can predict that residual, it captures the signal that the first model missed. We run this test in both directions with a shared ridge probe. Let the residuals of the two models be:

$$r_i^F = |y_i - \hat{y}_i^F|, \quad r_i^B = |y_i - \hat{y}_i^B|. \quad (4)$$

In one direction, we predict the feature-based model’s residual r^F using BERT’s representation, $\mathbf{h}_i$. In the other direction, we predict BERT’s residual r^B from the representation consisting of the vector of hand-crafted features $\mathbf{x}_i$. Each model’s residual is formed from the model’s own predictions on both the training and test splits, so the train and test targets are defined consistently, and both directions use the same ridge probe, so the two scores are comparable. We denote the two directions by $\mathbf{h} \rightarrow r^F$ for BERT to feature-based, and $\mathbf{x} \rightarrow r^B$ for feature-based to BERT. We report the test-set R^2 metric for each direction. A positive R^2 (in $\mathbf{h} \rightarrow r^F$) means that BERT captures a difficulty signal that the feature-based model missed; a near-zero R^2 (on $\mathbf{x} \rightarrow r^B$) would mean that the feature-based model adds nothing beyond BERT.

Error complementarity. As a simpler companion measure, we compare the per-document absolute errors, $|r_i^B|$ and $|r_i^F|$ for two models. We report their Pearson correlation, which indicates whether the two models err on the same documents or on different ones, and the share of test documents on which BERT is more accurate, i.e., where $|r_i^B| < |r_i^F|$.

3.4 RQ3: What linguistic information drives BERT’s predictions?

We characterise the linguistic basis of BERT’s predictions from three angles: which features its judgments track, what kind of input it relies on, and where in the network the difficulty signal arises.

Feature correlation. We first ask which linguistic properties BERT’s predictions of difficulty follow most closely. For each hand-crafted feature j we compute the correlation between the feature value $x^{(j)}$ and BERT’s prediction $\hat{y}^B$ across the test set, using both Pearson r (linear) and Spearman ρ (rank).

Input perturbation. To distinguish between the model’s reliance on surface-lexical cues from deep structural cues, we re-run BERT on *perturbed* inputs and measure how much its prediction changes. A perturbation maps text $\mathbf{x}_i$ to $\tilde{\mathbf{x}}_i$; we report the mean absolute shift:

$$\frac{1}{N} \sum_i |\hat{y}^B(\tilde{\mathbf{x}}_i) - \hat{y}^B(\mathbf{x}_i)|$$

and the correlation between the original and perturbed predictions. We use two perturbations. First,

word shuffling permutes the tokens within a document, which disturbs local syntax, while preserving the bag of words; a small shift then implies heavy reliance on lexical cues and a large shift implies reliance on word order. Second, sentence-order shuffling permutes entire sentences, which preserves local syntax but disturbs discourse order. Contrasting the two shifts distinguishes lexical, sentence-internal structural, and discourse-level cues.

Layer-wise probing. Lastly, we locate where in the neural network the difficulty signal arises. We take the [CLS] representation from each transformer layer ℓ and fit a ridge probe from that layer to the ground-truth difficulty score, reporting R^2 as a function of layer depth. We additionally report Pearson r between the probed and gold scores, which measures whether the probe captures the correct trend independently of absolute scale.

4 Experiments

4.1 RQ1: BERT encodes coarse features linearly, but not fine-grained morphology

Across all 66 features, the mean probe R^2 is 0.26 and the median is 0.22, so on average BERT recovers a moderate amount of each feature. This recovery is uneven: for 12 features R^2 surpasses 0.5, while for 29 features it drops below 0.2. No feature reaches $R^2 > 0.8$, so BERT does not encode any hand-crafted feature perfectly. The mean control R^2 is near zero (-0.31), and no feature has control R^2 above 0.1, which confirms that the probes are selective and that the recovery scores reflect information in the embedding representation rather than the probe’s own capacity.

How BERT recovers features is seen from Table 4. The best-recovered features are coarse measures of length, complexity, and lexical richness: the number of sentences and tokens, average and maximum sentence length, average word length, lexical variety, and dependency-tree depth. These describe how long and how structurally complex a text is. The least-recovered are fine-grained morphological and syntactic features that are specific to Finnish, such as individual case ratios, derivational suffixes, and negation. Their R^2 values are near zero or slightly negative, meaning BERT’s representation does not expose them *linearly*.

In summary, BERT linearly captures features encoding the coarse length and complexity properties, but not the fine-grained morphological features of Finnish. This sets up the question: if BERT does

Feature	probe R^2	Pearson	control R^2
<i>Top-10 most decodable (BERT “knows” these)</i>			
n_sentences	0.7581	0.8789	-0.2325
avg_max_dep_depth	0.6366	0.8007	-0.0207
n_unique_morph_feat_types	0.5993	0.7765	-0.8010
avg_word_length	0.5836	0.7652	-0.3706
n_unique_lemmas	0.5742	0.7692	-0.3596
long_word_ratio	0.5632	0.7541	-0.3036
max_sentence_length	0.5576	0.7484	-0.1021
n_compound_tokens	0.5537	0.7552	-0.4185
pos_ratio_pron	0.5410	0.7416	-0.6110
n_tokens	0.5348	0.7441	-0.3161
<i>Bottom-10 least decodable (BERT exposes these weakly)</i>			
negation_ratio	0.0578	0.3430	-0.2014
deriv_ja_ratio	0.0466	0.2803	-0.0680
case_ratio_ade	0.0044	0.1931	-0.0330
case_ratio_com	-0.0023	0.0135	-0.0030
case_ratio_ins	-0.0231	0.1308	-0.1028
case_ratio_tra	-0.0231	0.1209	-0.0202
case_ratio_abe	-0.0314	0.0065	-0.0846
deriv_ton_ratio	-0.0356	0.0749	-0.0168
pos_ratio_punct	-0.2251	0.2361	-2.9858
avg_compound_parts	-0.3411	0.2835	-5.3422

Table 4: Linear probing of hand-crafted features from BERT’s [CLS] representation. We show 10-most and 10-least decodable features. High probe R^2 with near-zero control R^2 indicates information genuinely encoded by BERT.

Metric	BERT	F-B	Δ	95% CI
QWK $\uparrow$	0.980	0.948	+0.032	[+0.019, +0.048]
R^2 $\uparrow$	0.960	0.906	+0.054	[+0.028, +0.081]
MAE $\downarrow$	0.094	0.211	-0.117	[-0.141, -0.092]

Table 5: BERT vs. Feature-Based model with paired-bootstrap significance (10,000 resamples). For MAE, lower is better, so negative difference favours BERT. All differences significant at $p < 0.001$.

not encode this detailed morphology, where does its advantage over the feature-based model come from?

4.2 RQ2: BERT subsumes the hand-crafted features, and adds signal that they miss

BERT is significantly more accurate. Table 5 reports the comparison between the two models, with significance assessed by a paired bootstrap over the test set (10,000 resamples). BERT is better on all three metrics: QWK is 0.980 vs. 0.948, R^2 is 0.960 vs. 0.906, and MAE is 0.094 vs. 0.211. Every difference is *significant* at $p < 0.001$, and none of the 95% confidence intervals cross zero, so the advantage is consistent, rather than an artefact of the particular test set. The gap is largest on MAE, where BERT’s average error is less than half that of the feature-based model.

Comparing performance. On about half of the documents, BERT is more accurate than the feature-based model (and on the other half it’s less accurate). However, when BERT makes mistakes, they are less serious than those of the feature-based model. Although BERT wins on every aggregate metric, the picture at the document level is more balanced. BERT is the more accurate model on only 49.4% of test documents, so the feature model is actually closer on slightly more than half of them (Figure 3). The correlation between the two models’ per-document absolute errors is moderate (0.334), meaning they tend to fail on different documents rather than the same ones. These two facts together explain the apparent contradiction: the models are evenly matched on most texts, and BERT’s clear advantage on aggregate metrics, especially the much lower MAE, comes from avoiding large errors on the documents where the feature model goes badly wrong. In other words, BERT does not win by a little everywhere; it wins by rarely being far off.

BERT subsumes the features and carries signal they miss. The residual-prediction analysis (Table 6) shows that BERT has more information than the feature-based model. BERT’s representation predicts 53% of the variance in the feature model’s residuals ($R^2 = 0.531$); therefore, much of what

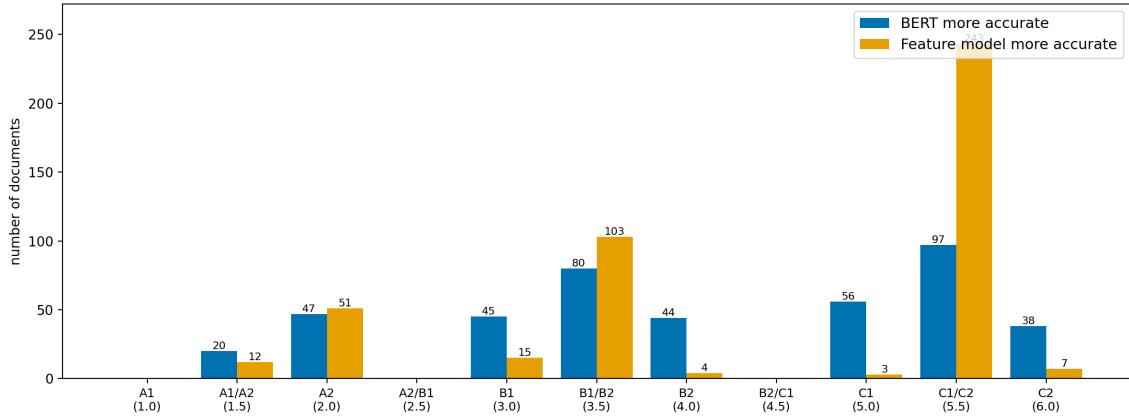


Figure 3: Per-document wins by CEFR level. For each 0.5-step, bars show how many test documents BERT predicts more accurately (blue) vs. how many feature-based model predicts more accurately (orange). Overall, the two models are nearly tied: BERT more accurate on 49.4% of documents, but their errors are only moderately correlated (0.334), indicating that they fail on different documents.

Predicting the residual of	R^2
Feature model, from BERT’s representation	0.531
BERT, from the hand-crafted features	−0.010

Table 6: Residual prediction. Each row asks whether one information source can predict the other model’s residual errors (test-set R^2). A positive value means the source carries signal the other model misses.

the feature-based model gets wrong is recoverable from BERT. In the opposite direction, the hand-crafted features explain essentially none of BERT’s residuals ($R^2 = -0.010$). The relationship is therefore one-directional: BERT contains the difficulty signal the features carry and adds more beyond that, while the features provide nothing that BERT is missing. This is consistent with the answer to RQ1, where BERT was shown to recover most of the hand-crafted features on its own.

4.3 RQ3: BERT’s predictions are driven by structural rather than lexical cues

Feature correlation. We first check which hand-crafted features BERT’s predictions track most closely. Table 7 lists the ten features most correlated with BERT’s predictions on the test set. The strongest correlates are again coarse measures of length and complexity, such as word length, sentence-length variability, maximum sentence length, and dependency length, together with a few morphological ratios. This mirrors the answer to RQ1: BERT’s judgments line up with the same coarse properties it recovers best. These correlations are descriptive; the perturbation and

Feature	Pearson r	Spearman ρ
long_word_ratio	0.534	0.465
avg_word_length	0.532	0.489
nonfinite_verb_ratio	0.516	0.507
std_sentence_length	0.482	0.664
max_sentence_length	0.443	0.585
n_compound_tokens	0.429	0.590
case_entropy	0.417	0.440
case_ratio_gen	0.412	0.388
derivation_ratio	0.407	0.424
max_dep_length	0.392	0.598

Table 7: The ten features most correlated with BERT’s difficulty predictions on the test set, ranked by Pearson r ; Spearman ρ is shown for comparison.

layer-wise analyses below provide interventional evidence.

Input perturbation. To separate lexical from structural cues, we re-run BERT on perturbed inputs and measure how much its prediction changes (Table 8). Shuffling the word order, which keeps the bag of words but destroys local syntax, shifts BERT’s prediction by 0.34 on average. Shuffling the sentence order, which keeps local syntax but destroys discourse order, shifts it by only 0.09 (median 0.01). Because word-order shuffling preserves the words and still changes the prediction substantially, BERT cannot be relying on lexical content alone; it draws on sentence-internal structure. The much smaller effect of sentence-order shuffling further shows that the relevant structure is found within sentences rather than across sentences.

Layer-wise probing: Finally, we probe the difficulty signal at each layer, as seen in Table 9.

Perturbation	Mean $ \Delta $	Median $ \Delta $
Word-order shuffle	0.344	0.196
Sentence-order shuffle	0.087	0.012

Table 8: Effect of input perturbations on BERT’s predictions. Larger mean absolute shift means BERT relies more on information destroyed by that perturbation.

Layer	R^2	r
0	-0.334	-
1	0.782	0.885
2	0.791	0.890
3	0.817	0.904
4	0.840	0.916
5	0.870	0.933
6	0.890	0.943
7	0.915	0.957
8	0.927	0.963
9	0.940	0.970
10	0.943	0.971
11	0.950	0.975
12	0.955	0.978

Table 9: Layer-wise probing. For each layer, test-set R^2 and Pearson r of probing the gold difficulty score from the [CLS] representation. The difficulty signal is absent at the embedding layer (0), appears at layer 1, and is refined as depth increases.

The signal starts out weak at the embedding layer ($R^2 = -0.33$), rises sharply in the first transformer layer ($R^2 = 0.78$), and then increases steadily with depth to $R^2 = 0.96$ in the final layer. Difficulty is therefore not a shallow surface statistic available at the input; it is built up and refined through the network, which confirms that difficulty is encoded as a relatively abstract property.

Summary. The three analyses agree. BERT’s predictions track coarse length and complexity features, they depend on word order rather than on the bag of words, and the difficulty signal is refined through successive layers. Together this indicates that BERT estimates difficulty from sentence-internal structure rather than from surface lexical content alone.

5 Conclusion

We explored the currently best-performing Finnish BERT text difficulty estimator to understand why it outperforms feature-based models, analysing it against a strong model built on 66 hand-crafted features. We found that BERT linearly encodes the coarse length and complexity features but not the fine-grained morphology of Finnish, that it subsumes the information in the hand-crafted fea-

tures and adds signal they miss while winning by rarely being far wrong rather than by winning everywhere, and that its predictions rely on sentence-internal structure rather than on lexical content alone. BERT’s advantage thus comes not from rediscovering the hand-picked “standard” features, but from exploiting additional structural information the features do not capture. In future work, we will explore a wider range of probing methods, in particular nonlinear probes, to more fully assess whether BERT encodes the linguistic features that our linear probes cannot recover.

6 Limitations

Our study has several limitations. It is based on a single Finnish dataset and a single fine-tuned model, so the findings may not transfer to other languages, domains, or architectures. We do not control for text length, which is a strong difficulty cue that both models may exploit, so part of the advantage we attribute to structure could be mediated by length. Our probes are linear, so they detect only linearly accessible information and may underestimate what BERT encodes nonlinearly; the residual and perturbation analyses are correlational or intervention-based and do not establish a full causal account. Finally, the hand-crafted features depend on automatic parser output, whose errors propagate into both the feature model and our analysis. We see these as natural directions for future work rather than threats to the main conclusions.

References

- Guillaume Alain and Yoshua Bengio. 2017. [Understanding intermediate layers using linear classifier probes](#).
- Ion Madrazo Azpiazu and Maria Soledad Pera. 2019. [Multiattentive recurrent neural network architecture for multilingual readability assessment](#). *Transactions of the Association for Computational Linguistics*, 7:421–436.
- Yonatan Belinkov. 2022. [Probing classifiers: Promises, shortcomings, and advances](#). *Computational Linguistics*, 48(1):207–219.
- Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. [An empirical investigation of statistical significance in NLP](#). In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 995–1005, Jeju Island, Korea. Association for Computational Linguistics.

- Kevyn Collins-Thompson and James P. Callan. 2004. [A language modeling approach to predicting reading difficulty](#). In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 193–200, Boston, Massachusetts, USA. Association for Computational Linguistics.
- Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. [What you can cram into a single \$\\$ \& ! \# *\$ vector: Probing sentence embeddings for linguistic properties](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.
- Council of Europe. 2001. *Common European Framework of Reference for Languages: Learning, Teaching, Assessment*. Cambridge University Press, Cambridge.
- 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 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Anna Dmitrieva and Aleksandra Kononova. 2023. [Creating a parallel Finnish-Easy Finnish dataset from news articles](#). In *Proceedings of the 1st Workshop on Open Community-Driven Machine Translation*, pages 21–26, Tampere, Finland. European Association for Machine Translation.
- Auli Hakulinen, Maria Vilkkumäki, Riitta Korhonen, Vesa Koivisto, Tarja Riitta Heinonen, and Irja Alho. 2004. *Iso suomen kielioppi*. Suomalaisen Kirjallisuuden Seura, Helsinki.
- John Hewitt and Percy Liang. 2019. [Designing and interpreting probes with control tasks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China. Association for Computational Linguistics.
- John Hewitt and Christopher D. Manning. 2019. [A structural probe for finding syntax in word representations](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.
- Jenna Kanerva, Filip Ginter, Niko Miekka, Akseli Leino, and Tapio Salakoski. 2018. [Turku neural parser pipeline: An end-to-end system for the CoNLL 2018 shared task](#). In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 133–142, Brussels, Belgium. Association for Computational Linguistics.
- Fred Karlsson. 2018. *Finnish: An Essential Grammar*, 3rd edition. Routledge, London.
- Anisia Katinskaia, Anh-Duc Vu, Jue Hou, Ulla Vahatalo, Yiheng Wu, and Roman Yangarber. 2025. [Estimation of text difficulty in the context of language learning](#). In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 594–611, Vienna, Austria. Association for Computational Linguistics.
- J. Peter Kincaid, Robert P. Fishburne, Richard L. Rogers, and Benjamin S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, Naval Air Station Memphis (Research Branch Report 8-75).
- Antonina Laposhina, Tatiana Veselovskaya, Maria Lebedeva, and Olga Kupreshchenko. 2018. [Automated text readability assessment for Russian second language learners](#). In *Proceedings of the International Conference "Dialogue 2018"*.
- Matej Martinc, Senja Pollak, and Marko Robnik-Šikonja. 2021. [Supervised and unsupervised neural approaches to text readability](#). *Computational Linguistics*, 47(1):141–179.
- Thang Pham, Trung Bui, Long Mai, and Anh Nguyen. 2021. [Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks?](#) In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1145–1160, Online. Association for Computational Linguistics.
- Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. [A primer in BERTology: What we know about how BERT works](#). *Transactions of the Association for Computational Linguistics*, 8:842–866.
- Serge Sharoff. 2022. What neural networks know about linguistic complexity. *Russian Journal of Linguistics*, 26(2):371–390.
- Koustuv Sinha, Robin Jia, Dieuwke Hupkes, Joelle Pineau, Adina Williams, and Douwe Kiela. 2021. [Masked language modeling and the distributional hypothesis: Order word matters pre-training for little](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2888–2913, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- A. Jackson Stenner. 1996. [Measuring reading comprehension with the Lexile framework](#). Technical report, MetaMetrics Inc., Durham, NC.

- Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. [BERT rediscovers the classical NLP pipeline](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.
- Sowmya Vajjala and Detmar Meurers. 2012. [On improving the accuracy of readability classification using insights from second language acquisition](#). In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 163–173, Montréal, Canada. Association for Computational Linguistics.
- Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. [Multilingual is not enough: BERT for Finnish](#). *arXiv preprint arXiv:1912.07076*.
- Yiheng Wu, Jue Hou, and Roman Yangarber. 2026a. Translation as augmentation: Effect of translated data on assessment of difficulty. In *Proceedings of the 2nd Workshop on Evaluating Text Difficulty in a Multilingual Context (DeTermIt! 2026)*, Palma de Mallorca, Spain. European Language Resources Association (ELRA).
- Yiheng Wu, Jue Hou, and Roman Yangarber. 2026b. Translation as augmentation: Effect of translated data on assessment of difficulty. In *Proceedings of the 2nd Workshop on Evaluating Text Difficulty in a Multilingual Context (DeTermIt!) LREC 2026*.

A Appendix: Feature list

Here we present in detail the hand-crafted features for Finnish that were used in the construction of the feature-based model.

Category	Feature	Description
Length & Surface	n_sentences	Number of sentences in document
Length & Surface	n_tokens	Total number of tokens
Length & Surface	avg_sentence_length	Mean sentence length (tokens per sentence)
Length & Surface	std_sentence_length	Standard deviation of sentence length (variability of sentence lengths)
Length & Surface	max_sentence_length	Length of the longest sentence
Length & Surface	min_sentence_length	Length of the shortest sentence
Length & Surface	avg_word_length	Mean word length (characters per word)
Length & Surface	long_word_ratio	Proportion of long words (above length threshold)
Lexical Richness	n_unique_lemmas	Number of distinct lemmas (unique words after removing inflection)
Lexical Richness	ttr_lemma_200	Type-token ratio over lemmas computed on the first 200 lemmas (lexical diversity)
Lexical Richness	n_unique_morph_feat_types	Number of distinct morphological feature types
Lexical Richness	avg_morph_feats_per_token	Average number of morphological features per token (morphological complexity)
Case System	case_ratio_nom	Proportion of nominative case
Case System	case_ratio_gen	Proportion of genitive case
Case System	case_ratio_par	Proportion of partitive case
Case System	case_ratio_acc	Proportion of accusative case
Case System	case_ratio_in	Proportion of inessive case ('inside')
Case System	case_ratio_ela	Proportion of elative case ('from inside')
Case System	case_ratio_ill	Proportion of illative case ('into')
Case System	case_ratio_ade	Proportion of adessive case ('at/on/by')
Case System	case_ratio_abl	Proportion of ablative case ('away from')
Case System	case_ratio_all	Proportion of allative case ('toward')
Case System	case_ratio_ess	Proportion of essive case ('as')
Case System	case_ratio_tra	Proportion of translative case ('becoming')
Case System	case_ratio_ins	Proportion of instructive case ('by means of')
Case System	case_ratio_abe	Proportion of abessive case ('without')
Case System	case_ratio_oblique	Proportion of oblique (non-nominative) cases
Case System	case_entropy	Entropy of case distribution (higher = more varied case usage)
Verbal Morphology	verb_ratio	Proportion of all verbs
Verbal Morphology	aux_ratio	Proportion of auxiliary verbs
Verbal Morphology	mood_ratio_ind	Proportion of indicative mood
Verbal Morphology	mood_ratio_cnd	Proportion of conditional mood ('would/will')
Verbal Morphology	mood_ratio_imp	Proportion of imperative mood (command)
Verbal Morphology	mood_ratio_pot	Proportion of potential mood ('maybe/possibly')
Verbal Morphology	tense_ratio_pres	Proportion of present tense
Verbal Morphology	tense_ratio_past	Proportion of past tense
Verbal Morphology	passive_ratio	Proportion of passive voice
Verbal Morphology	nonfinite_verb_ratio	Proportion of non-finite verbs (infinitives, participles, etc.)
Verbal Morphology	negation_ratio	Proportion of negation constructions

continued on next page

Table 10 – continued from previous page

Category	Feature	Description
Syntactic complexity	avg_dep_length	Mean dependency arc length (average distance between a word and its head)
Syntactic complexity	max_dep_length	Maximum dependency arc length
Syntactic complexity	avg_max_dep_depth	Average maximum dependency-tree depth per sentence (syntactic nesting)
Syntactic complexity	avg_clauses_per_sent	Average number of clauses per sentence
Syntactic complexity	avg_subord_per_sent	Average number of subordinate clauses per sentence
Compounding	compound_ratio	Proportion of compound words
Compounding	avg_compound_parts	Average number of parts of compound word
Compounding	n_compound_tokens	Number of compound tokens
Derivation	derivation_ratio	Total proportion of derived words
Derivation	deriv_minen_ratio	Proportion of -minen suffix (verb to noun, 'the act of doing')
Derivation	deriv_sti_ratio	Proportion of -sti suffix (adjective to adverb)
Derivation	deriv_llinen_ratio	Proportion of -llinen suffix (noun to adjective)
Derivation	deriv_ton_ratio	Proportion of -ton suffix ('without', like English -less)
Derivation	deriv_ja_ratio	Proportion of -ja suffix (verb to agent noun, 'one who does')
Derivation	deriv_vs_ratio	Proportion of -vs/-uus suffix (abstract noun-forming, '-ness')
Derivation	deriv_inen_ratio	Proportion of -inen suffix (adjective-forming)
POS Distribution	pos_ratio_noun	Proportion of nouns
POS Distribution	pos_ratio_verb	Proportion of verbs
POS Distribution	pos_ratio_adj	Proportion of adjectives
POS Distribution	pos_ratio_adv	Proportion of adverbs
POS Distribution	pos_ratio_pron	Proportion of pronouns
POS Distribution	pos_ratio_adp	Proportion of adpositions (pre/postpositions)
POS Distribution	pos_ratio_cconj	Proportion of coordinating conjunctions ('and/but')
POS Distribution	pos_ratio_sconj	Proportion of subordinating conjunctions ('because/if')
POS Distribution	pos_ratio_punct	Proportion of punctuation
POS Distribution	pos_ratio_num	Proportion of numerals

Table 10: The 66 hand-crafted linguistic features used by feature-based model.