Linguistic Analysis of Pretrained Sentence Encoders with Acceptability Judgments

Alex Warstadt[1]
warstadt@nyu.edu

Samuel R. Bowman[1,2,3]
bowman@nyu.edu

[1]Dept. of Linguistics
New York University
10 Washington Place
New York, NY 10003

[2]Dept. of Computer Science
New York University
60 Fifth Avenue
New York, NY 10011

[3]Center for Data Science
New York University
60 Fifth Avenue
New York, NY 10011

Abstract

Recent work on evaluating grammatical knowledge in pretrained sentence encoders gives a fine-grained view of a small number of phenomena. We introduce a new analysis dataset that also has broad coverage of linguistic phenomena. We annotate the development set of the Corpus of Linguistic Acceptability (CoLA; Warstadt et al., 2018) for the presence of 13 classes of syntactic phenomena including various forms of argument alternations, movement, and modification. We use this analysis set to investigate the grammatical knowledge of three pretrained encoders: BERT (Devlin et al., 2018), GPT (Radford et al., 2018), and the BiLSTM baseline from Warstadt et al. We find that these models have a strong command of complex or non-canonical argument structures like ditransitives (*Sue gave Dan a book*) and passives (*The book was read*). Sentences with long-distance dependencies like questions (*What do you think I ate?*) challenge all models, but for these, BERT and GPT have a distinct advantage over the baseline. We conclude that recent sentence encoders, despite showing near-human performance on acceptability classification overall, still fail to make fine-grained grammaticality distinctions for many complex syntactic structures.

1 Introduction

Models for sentence understanding such as BERT (Devlin et al., 2018) and GPT (Radford et al., 2018) are becoming more effective and ubiquitous, leading to a rise in new fine-grained datasets for evaluating their knowledge of grammar. Such evaluations are important for both guiding the development of more robust models and answering theoretical questions about what grammatical concepts are can be acquired through data-driven learning. To date, most evaluation datasets consist of constructed sentences illustrating a highly specific kind of grammatical contrast (Ettinger et al., 2016, 2018; Marvin and Linzen, 2018; Wilcox et al., 2018, 2019; Futrell and Levy, 2019), or naturalistic sentences labeled for a particular grammatical feature (Linzen et al., 2016; Shi et al., 2016; Conneau et al., 2017, 2018). These evaluation datasets are useful for studying a single phenomenon such as subject-verb agreement in depth (Linzen et al., 2016), but fail to test domain general grammatical knowledge.

By contrast, the Corpus of Linguistic Acceptability[1] (CoLA; Warstadt et al., 2018) tests these models' ability to judge the grammatical acceptability of sentences in a wide domain. CoLA is a dataset of over 10k example sentences labeled for acceptability and sampled from linguistics publications discussing numerous linguistic phenomena. However, in its original formulation, CoLA does not distinguish between phenomena, making it difficult to analyze a model's knowledge a particular phenomenon.

In this work, we augment CoLA with a new syntactically annotated evaluation set[2] that is both fine-grained and domain general. All 1043 examples in the CoLA development set are labeled with expert annotations that indicate the presence or absence of 13 classes of syntactic phenomena, each defined as a union of several more specific phenomena. This resource makes it uniquely easy to conduct analyses whose conclusions can be directly interpreted in terms of modern linguistic theory, since CoLA data is sampled from example sentences in mainstream linguistics publications and our annotations adapt concepts from that

[1]The original CoLA can be downloaded here: https://nyu-mll.github.io/CoLA/#

[2]The grammatically annotated CoLA can be downloaded here: https://nyu-mll.github.io/CoLA/#grammatical_annotations

literature.

We use our analysis set to assess GPT and BERT, which achieve near-human performance (Nangia and Bowman, 2019) on many tasks in the GLUE benchmark (Wang et al., 2018), including CoLA. We treat acceptability classification as a probing task (Adi et al., 2017), in which a small classifier is trained on CoLA on top of pretrained encoders and tested on the analysis set. We compare these models to a baseline pretrained BiLSTM model released by Warstadt et al. (2018) with CoLA.

Our results identify specific syntactic features that make sentences harder to classify, such as long distance dependencies (*What do you think I ate?*), and others that have little effect on difficulty, such as non-canonical argument structures like passives (*The book was read*). Furthermore, some constructions highlight or minimize the differences between models. For example, GPT and BERT far out perform the BiLSTM on movement phenomena such as clefts (*It is Bo that left*), yet have no advantage on sentences with adjuncts (*Sue exercises in the morning*). We wish to exercise caution in interpreting these results since it is not clear to what extent an encoder's failure on a particular phenomenon is due to a weakness of the encoder rather than the training data or probing classifier. However, this caveat applies to varying degrees to all probing resources. In this context of other similar linguistically informed datasets, our analysis set addresses the critical need for a evaluation task with wide coverage of linguistic phenomena.

2 Related Work

Sentence Encoders Recent research tries to reproduce the success of robust pretrained word embeddings (Mikolov et al., 2013; Pennington et al., 2014) at the sentence level, in the form of reusable sentence encoders with pretrained weights. Current state-of-the-art sentence encoders are pretrained on language modeling or related self-supervised tasks. Among these, ELMo (Peters et al., 2018) uses a BiLSTM architecture, while GPT (Radford et al., 2018) and BERT (Devlin et al., 2018) use the newer attention-based Transformer architecture (Vaswani et al., 2017). Unlike most earlier approaches where the weights of the encoder are frozen after pretraining, the last two fine-tune the encoder on the downstream task. They are among the top performing[3] models on the GLUE benchmark, an evaluation suite for general-purpose sentence understanding models like these, which is built around a set of nine sentence-level classification tasks (Wang et al., 2018).

Probing Sentence Representations The evaluation and analysis of sentence representations is an active area of research. Most of this work to date focuses on in-depth investigation of a particular phenomenon. A popular evaluation technique uses probing tasks in which a small probing classifier is trained to identify particular syntactic and surface features of a sentence based on a sentence representation.

Some datasets for probing tasks label naturally occurring data with the relevant features. For instance, Shi et al. (2016) label sentences with features such as past/present tense and active/passive voice. Linzen et al. (2016) label present tense verbs for whether they have singular or plural agreement marking. Adi et al. (2017) label sentences for length and word content. Conneau et al. (2018) label sentences for syntactic depth and morphological number.

Another common method is to semi-automatically generate data manipulating a small set of grammatical features. For instance, Ettinger et al. (2016, 2018) build datasets of this kind to test whether sentence encoders represent the scope of negation and semantic roles, and Kann et al. (2019) do so to test whether word and sentence encoders representations information about verbal argument structure.

CoLA & Acceptability Classification The Corpus of Linguistic Acceptability (Warstadt et al., 2018) is a dataset of 10k example sentences including expert annotations for grammatical acceptability. The sentences are examples taken from 23 theoretical linguistics publications and represent a wide array of phenomena discussed in that literature. Such example sentences are usually labeled for acceptability by their authors or a small group of native English speakers. A small random sample of the CoLA development set (with our added annotations) can be seen in Table 1.

Within computational linguistics, the acceptability classification task has been explored pre-

[3]The current top performing model (Liu et al., 2019) is based on BERT, but includes ensembling and extra fine-tuning.

Acceptability	Sentence	Simple	Locative	PP Arg-VP	High Arity	Passive	Binding:Other	Emb Q	Complex QP	Modal	Raising	Trans-Adj	Coord	Ellipsis/Anaphor	Comparative
✓	The magazines were sent by Mary to herself.			✗	✗	✗									
✓	John can kick the ball.									✗					
*	I know that Meg's attracted to Harry, but they don't know who.			✗				✗					✗	✗	
✓	They kicked them	✗					✗								
✓	Which topic did you choose without getting his approval?								✗						
*	It was believed to be illegal by them to do that.				✗	✗					✗	✗			
*	Us love they.	✗													
*	The more does Bill smoke, the more Susan hates him.						✗								✗
✓	I ate a salad that was filled with lima beans.			✗		✗									
✓	That surprised me.	✗													

Table 1: A random sample of sentences from the CoLA development set, shown with their original acceptability labels (✓= acceptable, *=unacceptable) and a subset of our new phenomenon-level annotations from the set of finer-grained features.

viously: Lawrence et al. (2000) train RNNs to do acceptability classification over sequences of POS tags corresponding to example sentences from a syntax textbook. Wagner et al. (2009) also train RNNs, but using naturally occurring sentences that have been automatically manipulated to be unacceptable. Lau et al. (2016) predict acceptability from language model probabilities, applying this technique to sentences from a syntax textbook, and sentences which were translated round-trip through various languages.

Lau et al. also attempt to model gradient crowd-sourced acceptability judgments, reflecting an ongoing debate about whether binary expert judgments like those in CoLA are reliable (Gibson and Fedorenko, 2010; Sprouse and Almeida, 2012). We remain agnostic as to the role of binary judgments in linguistic theory, but note that Warstadt et al. (2018) and Nangia and Bowman (2019) measure expert and non-expert human performance, respectively, on subsets of CoLA (see Table 4 for the former's results), both finding that new human annotators, while not in perfect agreement with the judgments in CoLA, still agree well and outperform the best neural network models.

3 Analysis Set

We introduce a grammatically annotated version of the entire CoLA development set to facilitate detailed error analysis of acceptability classifiers. These 1043 sentences are labeled with 13 major features, further divided into 59 minor features. Each feature marks the presence of a particular phenomenon or class of phenomena in the sentence. Each minor feature belongs to a single major feature. A sentence belongs to a major feature if it belongs to one or more of the relevant minor features. The supplementary materials include descriptions of each feature along with examples and the criteria used for annotation.

The major and minor features are listed in Table 2, and are fully documented in the Appendix. The average sentence is positively labeled with 3.22 major features (SD=1.66) on average, and the average a major feature is present in 224 sentences (SD=112). The average sentence is positively labeled with 4.31 minor features (SD=2.59). The average minor feature is present in 71.3 sentences (SD=54.7). Every sentence is labeled with at least one feature. Sentences without any obvious phenomena of interest are labeled SIMPLE.

3.1 Annotation

The sentences were annotated manually by one of the authors, who is trained in formal linguistics and linguistic annotation. The features were developed in a trial stage, in which the annotator performed a similar annotation with different annotation schema for several hundred sentences from CoLA not belonging to the development set.

3.2 Feature Descriptions

Here we briefly summarize the feature set in order of the major features. These constructions are well-studied in syntax, and further background can be found in textbooks such as Adger (2003) and Sportiche et al. (2013).

Major Feature (n)	**Minor Features** (n)
Simple (87)	Simple (87)
Pred (256)	Copula (187), Pred/SC (45), Result/Depictive (26)
Adjunct (226)	VP Adjunct (162), Misc Adjunct (75), Locative (69), NP Adjunct (52), Temporal (49), Particle (33)
Arg Types (428)	PP Arg VP (242), Oblique (141), PP Arg NP/AP (81), Expletive (78), by-Phrase (58)
Arg Altern (421)	High Arity (253), Passive (114), Drop Arg (112), Add Arg (91)
Bind (121)	Binding:Other (62), Binding:Refl (60)
Question (222)	Emb Q (99), Pied Piping (80), Rel Clause (76), Matrix Q (56), Island (22)
Comp Clause (190)	CP Arg VP (110), No C-izer (41), Deep Embed (30), CP Arg NP/AP (26), Non-finite CP (24), CP Subj (15)
Auxiliary (340)	Aux (201), Modal (134), Neg (111), Psuedo-Aux (26)
to-VP (170)	Control (80), Non-finite VP Misc (38), VP Arg NP/AP (33), VP+Extract (26), Raising (19)
N, Adj (278)	Compx NP (106), Rel NP (65), Deverbal (53), Trans Adj (39), NNCompd (35), Rel Adj (26), Trans NP (21)
S-Syntax (286)	Coord (158), Ellipsis/Anaphor (118), Dislocation (56), Subordinate/Cond (41), Info Struc (31), S-Adjunct (30), Frag/Paren (9)
Determiner (178)	Quantifier (139), NPI/FCI (29), Comparative (25), Partitive (18)

Table 2: Major features and their associated minor features (with number of occurrences n).

Simple This major feature contains only one minor feature, SIMPLE, including sentences where the subject and predicate are unmodified (*Bo ate an apple*.)

Pred(icate) These three features correspond to predicative phrases, including copular constructions (*Bo is awake.*), small clauses (*I saw Bo jump*), and resultatives/depictives (*Bo wiped the table clean*).

Adjunct These six features mark various kinds of optional modifiers, including modifiers of NPs (*The boy with blue eyes gasped*) or VPs (*Bo sang with Jo*), and temporal (*Bo awoke at dawn*) or locative (*Bo jumped on the bed*) adjuncts.

Argument types These five features identify syntactically selected arguments, differentiating, for example, obliques (*I gave a book to Bo*), PP arguments of NPs and VPs (*Bo voted for Jones*), and expletives (*It seems that Bo left*).

Argument Alternations These four features mark VPs with unusual argument structures, including added arguments (*I baked Bo a cake*) or dropped arguments (*Bo knows*), and the passive (*I was applauded*).

Bind These are two minor features, one for bound reflexives (*Bo loves himself*), and one for other bound pronouns (*Bo thinks he won*).

Question These five features apply to sentences with question-like properties. They mark whether the interrogative is an embedded clause (*I know who you are*), a matrix clause (*Who are you?*), or a relative clause (*Bo saw the guy who left*); whether it contains an island out of which extraction is unacceptable (**What was a picture of hanging on the wall?*)[4]; or whether there is pied-piping or a multi-word *wh*-expressions (*With whom did you eat?*).

Comp(lement) Clause These six features apply to various complement clauses (CPs), including subject CPs (*That Bo won is odd*); CP arguments of VPs or NPs/APs (*The fact that Bo won*); CPs missing a complementizer (*I think Bo's crazy*); or non-finite CPs (*This is ready for you to eat*).

Aux(iliary) These four minor features mark the presence of auxiliary or modal verbs (*I can win*), negation, or "pseudo-auxiliaries" (*I have to win*).

to-VP These five features mark various infinitival embedded VPs, including control VPs (*Bo wants to win*); raising VPs (*Bo seemed to fly*); VP arguments of NPs or APs (*Bo is eager to eat*); and VPs with extraction (e.g. *This is easy to read* _).

N(oun), Adj(ective) These seven features mark complex NPs and APs, including ones with PP arguments (*Bo is fond of Mo*), or CP/VP arguments; noun-noun compounds (*Bo ate mud pie*); modified NPs, and NPs derived from verbs (*Baking is fun*).

S-Syntax These seven features mark various unrelated syntactic constructions, including dislocated phrases (*The boy left who was here earlier*); movement related to focus or information structure (*This I've gotta see* ___); coordination, subor-

[4]Following standard notation in linguistics, the '*' precedes sentences that are not grammatically acceptable.

dinate clauses, and ellipsis (*I can't*); or sentence-level adjuncts (*Apparently, it's raining*).

Determiner These four features mark various determiners, including quantifiers, partitives (*two of the boys*), negative polarity items (*I *do/don't have any pie*), and comparative constructions.

3.3 Correlations

These features are overlapping and in many cases are correlated, so not all results from using this analysis set will be independent. We analyzed the between-feature pairwise Matthews Correlation Coefficient (MCC; Matthews, 1975) of the 63 minor features (giving 1953 pairs), and of the 15 major features (giving 105 pairs). MCC is a special case of Pearson's r for Boolean variables.[5] These results are summarized in Table 3. Regarding the minor features, 60 pairs had a correlation of 0.2 or greater and 15 had a correlation of 0.5 or greater. Turning to the major features, 6 pairs had a correlation of 0.2 or greater, and 2 had an anti-correlation of greater magnitude than -0.2.

We can see at least three reasons for these observed correlations. First, some features have overlapping definitions; for example EXPLETIVE is a strict subset of ADD ARG because expletive arguments (e.g. *There are birds singing*) are by definition non-canonical. Similarly, the strong anti-correlation between SIMPLE and the two features related to argument structure, ARGUMENT TYPES and ARG ALTERN, follows from the definition of SIMPLE, which explicitly excludes sentences with unusual argument structure. Second, grammatical facts of English drive the correlation between, for instance, QUESTION and AUX, because main-clause questions in English require subject-aux inversion. Third, the unusually high correlation of, for example, EMB-Q and ELLIPSIS/ANAPHOR, can be attributed largely to a bias in a particular source in CoLA, Chung et al. (1995), which is an article about the sluicing construction involving ellipsis of an embedded interrogative (e.g. *I saw someone, but I don't know who*). This third case highlights a limitation of this analysis set. The set of examples associated with a particular feature is not a controlled set designed to test knowledge of that particular construction, but rather a sample of sentences from the linguistics literature, and as such may not be full representative of the construction in question. However, this cost comes with the advantage that results on the analysis set can be directly connected to relevant linguistics literature.

Label 1	**Label 2**	**MCC**
	Major Features	
Argument Types	Arg Altern	0.406
Question	Auxiliary	0.273
Question	S-Syntax	0.232
Predicate	N, Adj	0.231
Auxiliary	S-Syntax	0.224
Question	N, Adj	0.211
Simple	Arg Altern	-0.227
Simple	Argument Types	-0.238
	Minor Features	
PP Arg NP/AP	Rel NP	0.755
by-Phrase	Passive	0.679
Coord	Ellipsis/Anaphor	0.634
VP Arg NP/AP	Trans Adj	0.628
NP Adjunct	Compx NP	0.623
Oblique	High Arity	0.620
RC	Compx NP	0.565
Expletive	Add Arg	0.558
CP Arg NP/AP	Trans NP	0.546
PP Arg NP/AP	Rel Adj	0.528

Table 3: Correlation (MCC) of features in the annotated analysis set. We display only the correlations with the greatest magnitude.

4 Models Evaluated

We train MLP acceptability classifiers for CoLA on top of three sentence encoders: (1) CoLA's pretrained BiLSTM baseline encoder, (2) OpenAI GPT, and (3) BERT. We use publicly available pretrained sentence encoders.[6]

LSTM Encoder: CoLA Baseline The CoLA baseline model is the sentence encoder with the highest performance on CoLA from Warstadt et al. The encoder uses a BiLSTM, which reads the sentence word-by-word in both directions, with max-pooling over the hidden states. Similar to ELMo (Peters et al., 2018), the inputs to the BiLSTM are the hidden states of a language model (only a forward language model is used in contrast with

[5] MCC measures correlation of two binary distributions, giving a value between -1 and 1. On average, any two unrelated distributions will have a score of 0, regardless of class imbalance. This is contrast to metrics like accuracy or F1, which favor classifiers with a majority-class bias.

[6] CoLA baseline: https://github.com/nyu-mll/CoLA-baselines
OpenAI GPT: https://github.com/openai/finetune-transformer-lm
BERT: https://github.com/google-research/bert

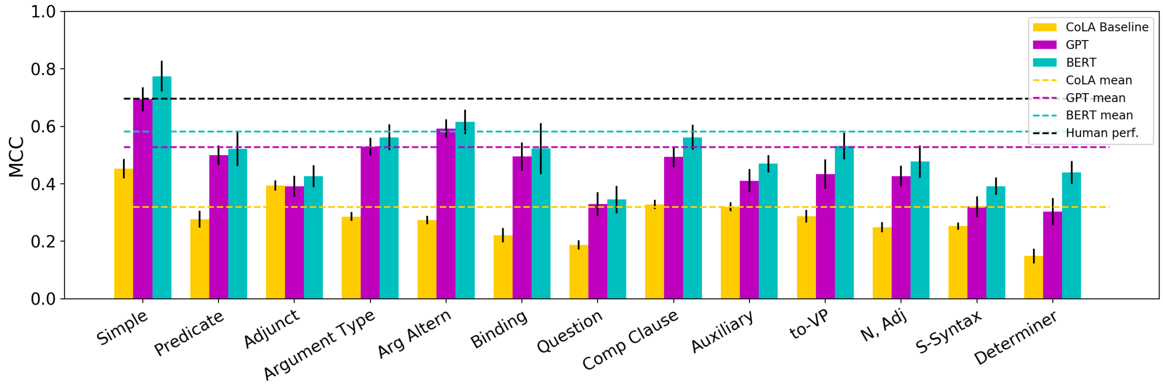

Figure 1: Performance (MCC) on our analysis set by major feature. Dashed lines show mean performance on the entire CoLA development set. Error bars mark the mean ±1 standard deviation. From left to right, performance for each feature is given for the CoLA Baseline, OpenAI GPT, and BERT.

ELMo). The encoder is trained on a real/fake discrimination task which requires it to identify whether a sentence is naturally occurring or automatically generated. We train acceptability classifiers on CoLA using the CoLA baselines codebase with 20 random restarts, following the original authors' transfer-learning approach: The sentence encoder's weights are frozen, and the sentence embedding serves as input to an MLP with a single hidden layer. All hyperparameters are held constant across restarts.

Transformer Encoders: GPT and BERT In contrast with recurrent models, GPT and BERT use a self attention mechanism which combines representations for each (possibly non-adjacent) pair of words to give a sentence embedding. GPT is trained using a standard language modeling task, while BERT is trained with masked language modeling and next sentence prediction tasks. We use $\text{BERT}_{\text{LARGE}}$. For each encoder, we train 20 random restarts on CoLA feeding the pretrained models published by these authors into a single output layer, using code which will be released upon publication. Following the methods of the original authors, we fine-tune the encoders during training on CoLA. All hyperparameters are held constant across restarts.

5 Results

5.1 Overall CoLA Results

The overall performance of the three sentence encoders is shown in Table 4. Following Warstadt et al., performance on CoLA is measured using MCC. We present the best single restart for each encoder, the mean over restarts for an encoder, and the result of ensembling the restarts for a given encoder, i.e. the majority classification for a given sentence, or *acceptable* if tied.[7] For BERT results, we exclude 5 out of the 20 restarts because they were degenerate (MCC=0).

	Mean (STD)	**Max**	**Ensemble**
CoLA	0.320 (0.007)	0.330	0.320
GPT	0.528 (0.023)	0.575	0.567
BERT	**0.582** (0.032)	**0.622**	**0.601**
Human	*0.697 (0.042)*	*0.726*	*0.761*

Table 4: Performance (MCC) on the CoLA test set, including mean over restarts of a given model with standard deviation, max over restarts, and majority prediction over restarts. Human agreement is measured by Warstadt et al..[8]

Across the board, BERT outperforms GPT, which outperforms the CoLA baseline. However, BERT and GPT are much closer in performance than they are to CoLA baseline. While ensemble performance exceeded the average for BERT and GPT, it did not outperform the best single model.

5.2 Analysis Set Results

The results for the major and minor features are shown in Figures 1 and 2, respectively. For each feature, we measure the MCC of the sentences including that feature. We plot the mean of these results across the different restarts for each model.

Comparison across features reveals that the presence of certain features has a large effect on performance, and we comment on some patterns

[7]Because we use the development set for analysis, we do not use it to weight models for weighted ensembling.

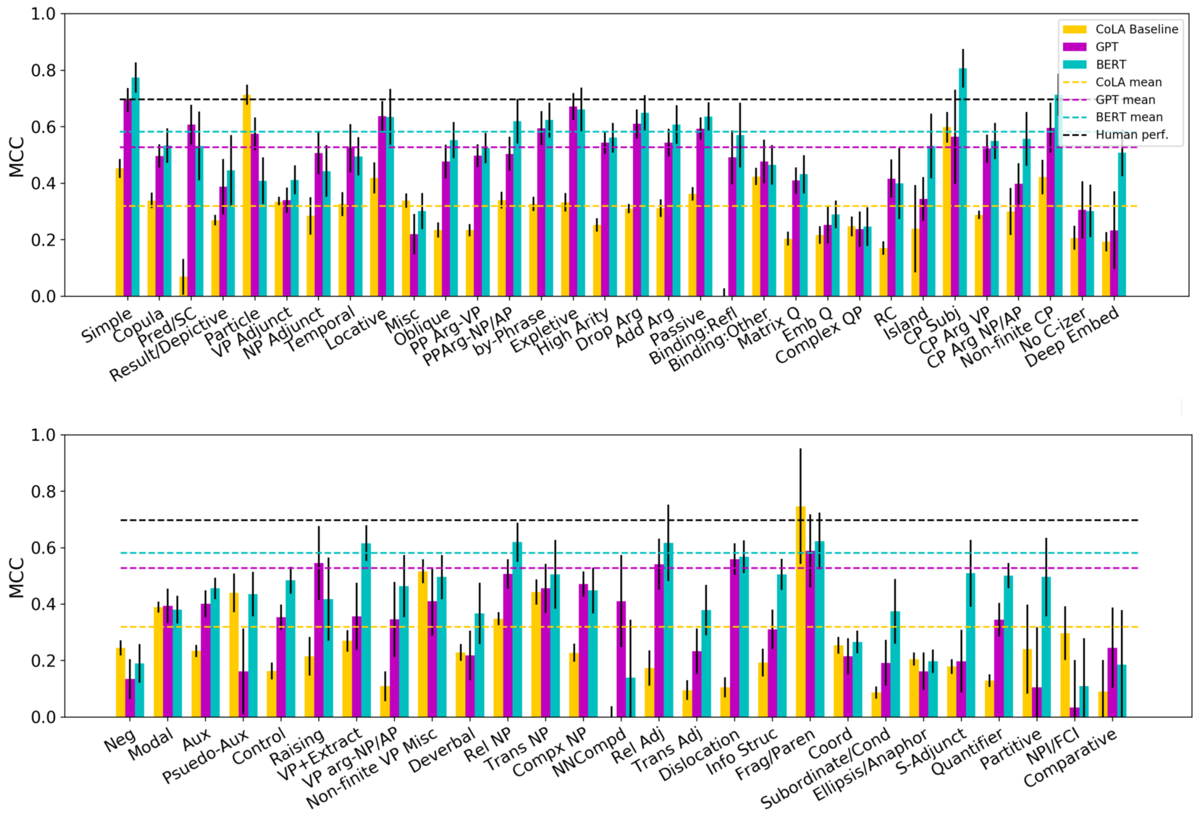

Figure 2: Performance (MCC) on our analysis set by minor feature. Dashed lines show mean performance on the entire CoLA development set. Error bars mark the mean ± 1 standard deviation. From left to right, performance for each feature is given for the CoLA Baseline, OpenAI GPT, and BERT.

below. Within a given feature, the effect of model type is overwhelmingly stable, and resembles the overall difference in performance. However, we observe several interactions, i.e. specific features where the relative performance of models does not track their overall relative performance. In interpreting these results, we caution against drawing strong conclusions from rare minor features. For this reason, we do not discuss any results for features appearing in fewer than 50 sentences. Furthermore, we cannot conclude with certainty that any particular mode of success or failure reflects what the information in the pretrained encoder, rather than what sorts of contrasts are easy or hard to learn from the CoLA training data. However, we consider results for major features more likely to be reliable due to the large number and variety of sentences with each label.

Comparing Features Among the major features (Figure 1), performance is universally highest on the SIMPLE sentences, and is higher than each model's overall performance. Otherwise we find that a model's performance on sentences of a given feature is on par with or lower than its overall performance, reflecting the fact that features mark the presence of unusual or complex syntactic structure. Performance is also high (and close to overall performance) on sentences with marked argument structures (ARGUMENT TYPES and ARG(UMENT) ALT(ERNATION)), indicating that argument structure is relatively easy to learn.

Comparing different kinds of embedded content, we observe higher performance on sentences with embedded clauses (major feature=COMP CLAUSE) embedded VPs (major feature=TO-VP) than on sentences with embedded interrogatives (minor features=EMB-Q, REL CLAUSE). Interrogatives are quite challenging in general (major feature=QUESTION). Sentences with question-like syntax may be difficult because they usually involve extraction of a *wh*-word, creating a long-distance dependency between the *wh*-word and its extraction site, which may be difficult for models to recognize.

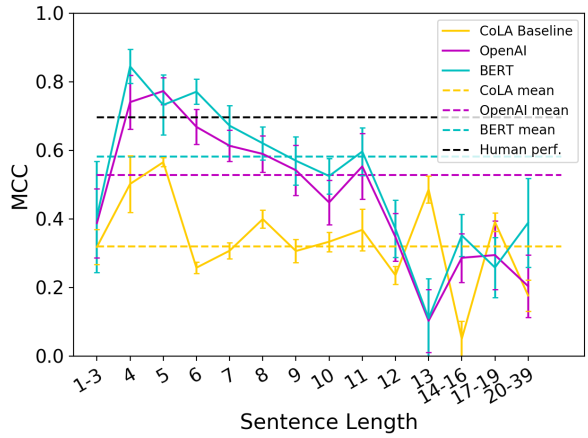

Figure 3: Performance (MCC) on the CoLA analysis set by sentence length.

Comparing Models Comparing within-feature performance of the three encoders to their overall performance, we find they have differing strengths and weaknesses. BERT and GPT generally far outperform the CoLA baseline, with BERT performing best in most cases. BERT and GPT have a particularly large advantage in sentences involving long-distance dependencies. They outperform the CoLA baseline by an especially wide margin on BIND:REFL, which involves establishing a dependency between a reflexive and its antecedent (*Bo tries to love himself*), as well as DISLOCATION, in which expressions are separated from their dependents (*Bo practiced on the train an important presentation*). The advantage of BERT and GPT may be due in part to their use of the Transformer architecture. Unlike the BiLSTM used by the CoLA baseline, the Transformer uses a self-attention mechanism that associates all pairs of words regardless of distance.

In some specific instances, we do not observe the usual pattern of BERT outperforming GPT and both far outperforming the CoLA baseline, revealing possible idiosyncrasies of the sentence representations they output. For instance, the CoLA baseline performs on par with the others on the major feature ADJUNCT, especially considering the minor feature PARTICLE (*Bo looked the word up*).

5.3 Length Analysis

For comparison, we analyze the effect of sentence length on acceptability classifier performance. The results are shown in Figure 3. The results for the CoLA baseline are inconsistent, but do drop off as sentence length increases. For BERT and GPT, performance decreases very steadily with length. Exceptions are extremely short sentences (length 1-3), which may be challenging due to insufficient information; and extremely long sentences, where we see a small (but somewhat unreliable) boost in BERT's performance. BERT and GPT are generally quite close in performance, except on the longest sentences, where BERT's performance is considerably better.

6 Conclusion

Using a new grammatically annotated analysis set, we identify several syntactic phenomena that are predictive of good or bad performance of current state-of-the-art sentence encoders on CoLA. We also use these results to develop hypotheses about why BERT is successful, and why Transformer models outperform sequence models.

Our findings can guide future work on sentence representation. Transformer models appear to have an advantage over sequence models with long-distance dependencies, but still struggle with these constructions relative to more local phenomena. It stands to reason that this performance gap might be widened by training larger or deeper Transformer models, or training on longer or more complex sentences. This analysis set can be used by engineers interested in evaluating the syntactic knowledge of their encoders.

Finally, these findings suggest possible controlled experiments that could confirm whether there is a causal relation between the presence of the syntactic features we single out as interesting and model performance. Our results are purely correlational, and do not mark whether a particular construction is crucial for the acceptability of the sentence. Future experiments following Ettinger et al. (2018) and Kann et al. (2019) can semi-automatically generate datasets by manipulating, for example, length of long-distance dependencies, inflectional violations, or the presence of interrogatives, while controlling for factors like sentence length and word choice, in order to determine the extent to which these features impact the quality of sentence representations.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 1850208. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This project has also benefited from financial support to SB by Samsung Research under the project *Improving Deep Learning using Latent Structure* and from the donation of a Titan V GPU by NVIDIA Corporation.

References

David Adger. 2003. *Core Syntax: A Minimalist Approach*. Oxford University Press Oxford.

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *Proceedings of ICLR Conference Track. Toulon, France.*

Sandra Chung, William A Ladusaw, and James McCloskey. 1995. Sluicing and logical form. *Natural Language Semantics*, 3(3):239–282.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680.

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. *arXiv preprint arXiv:1805.01070*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Allyson Ettinger, Ahmed Elgohary, Colin Phillips, and Philip Resnik. 2018. Assessing composition in sentence vector representations. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1790–1801. Association for Computational Linguistics.

Allyson Ettinger, Ahmed Elgohary, and Philip Resnik. 2016. Probing for semantic evidence of composition by means of simple classification tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 134–139.

Richard Futrell and Roger P Levy. 2019. Do rnns learn human-like abstract word order preferences? *Proceedings of the Society for Computation in Linguistics*, 2(1):50–59.

Edward Gibson and Evelina Fedorenko. 2010. Weak quantitative standards in linguistics research. *Trends in Cognitive Sciences*, 14(6):233–234.

Katharina Kann, Alex Warstadt, Adina Williams, and Samuel R Bowman. 2019. Verb argument structure alternations in word and sentence embeddings. *Proceedings of the Society for Computation in Linguistics*, 2(1):287–297.

Jey Han Lau, Alexander Clark, and Shalom Lappin. 2016. Grammaticality, acceptability, and probability: a probabilistic view of linguistic knowledge. *Cognitive Science*, 41(5):1202–1241.

Steve Lawrence, C Lee Giles, and Sandiway Fong. 2000. Natural language grammatical inference with recurrent neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 12(1):126–140.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of lstms to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Improving multi-task deep neural networks via knowledge distillation for natural language understanding. *CoRR*, abs/1904.09482.

Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202.

Brian W Matthews. 1975. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, 405(2):442–451.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Nikita Nangia and Samuel R Bowman. 2019. A conservative human baseline estimate for glue: People still (mostly) beat machines.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 2227–2237.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/language-unsupervised/language_ understanding_paper. pdf*.

Xing Shi, Inkit Padhi, and Kevin Knight. 2016. Does string-based neural MT learn source syntax? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1526–1534.

Dominique Sportiche, Hilda Koopman, and Edward Stabler. 2013. *An Introduction to Syntactic Analysis and Theory*. John Wiley & Sons.

Jon Sprouse and Diogo Almeida. 2012. Assessing the reliability of textbook data in syntax: Adger's core syntax. *Journal of Linguistics*, 48(3):609–652.

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.

Joachim Wagner, Jennifer Foster, and Josef van Genabith. 2009. Judging grammaticality: Experiments in sentence classification. *CALICO Journal*, 26(3):474–490.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2018. Neural network acceptability judgments. *arXiv preprint arXiv:1805.12471*.

Ethan Wilcox, Roger Levy, Takashi Morita, and Richard Futrell. 2018. What do rnn language models learn about filler–gap dependencies? In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 211–221.

Ethan Wilcox, Peng Qian, Richard Futrell, Miguel Ballesteros, and Roger Levy. 2019. Structural supervision improves learning of non-local grammatical dependencies. *arXiv preprint arXiv:1903.00943*.