

Local Structure Matters Most in Most Languages

Louis Clouâtre^{1,3} Prasanna Parthasarathi² Amal Zouaq¹ and Sarath Chandar^{1,3,4}

¹ Polytechnique Montréal

² Noah’s Ark Lab, Huawei Canada

³ Quebec Artificial Intelligence Institute (Mila)

⁴ Canada CIFAR AI Chair

Abstract

Many recent perturbation studies have found unintuitive results on what does and does not matter when performing Natural Language Understanding (NLU) tasks in English. Coding properties, such as the order of words, can often be removed through shuffling without impacting downstream performances. Such insight may be used to direct future research into English NLP models. As many improvements in multilingual settings consist of wholesale adaptation of English approaches, it is important to verify whether those studies replicate or not in multilingual settings. In this work, we replicate a study on the importance of local structure, and the relative unimportance of global structure, in a multilingual setting. We find that the phenomenon observed on the English language broadly translates to over 120 languages, with a few caveats.

1 Introduction

A recent research trend has explored the sensitivity, or insensitivity, of neural language models to different perturbations of texts (Pham et al., 2021; Sinha et al., 2020, 2021; Gupta et al., 2021; O’Connor and Andreas, 2021; Taktasheva et al., 2021; Clouatre et al., 2022). Their findings may be central in directing future NLP research by providing insight into which coding property (Kulmizev and Nivre, 2021) of language are most valuable to performing Natural Language Understanding (NLU) tasks. As research in English NLP tends to be adapted to other languages, such as through single language adaptation of BERT-style models (Devlin et al., 2019; Cui et al., 2019; Le et al., 2019; Martin et al., 2019; Antoun et al., 2020; Carmo et al., 2020; de Vries et al., 2019; Malmsten et al., 2020; Polignano et al., 2019; Nguyen and Tuan Nguyen, 2020) or multilingual adaptations of the same architecture (Lample and Conneau, 2019; Clark et al., 2021; Xue et al., 2020, 2021; Liu et al., 2020; Devlin et al., 2019),

it is vital that we verify how insights derived from the English language generalize to other languages.

One such coding property, the local structure of text, has recently been shown to be ubiquitously relied upon by both neural language models (Clouatre et al., 2022) and humans (Mollica et al., 2020) to understand text in English. The global structure of text only sometimes being necessary for a model to perform NLU tasks (Clouatre et al., 2022). Such results motivate hierarchical approaches to neural language model development, where one would first build meaning locally and then reason over the global context if necessary. However, we must verify that the importance of that coding property is not merely an artifact of the English language.

In this short paper, our contributions are as follows:

- We adapt and replicate the findings of Clouatre et al. (2022) in a multilingual setting to verify their generality and find that their conclusions regarding both local and global structure broadly apply to most of the 120 languages surveyed.
- We provide analysis for why text using Chinese Characters as its script may be more resilient to local perturbations and highlight the importance of testing improvements in English neural modeling in other languages.

2 Related Work

Text Perturbations and Structure Probing Several text perturbation schemes have been explored to probe what kind of structure does and does not matter for neural models performing NLU. Sankar et al. (2019) explores both shuffling and reversing utterances and words in a generative dialogue setting, highlighting models’ insensitivity to the order of conversational history. Pham et al. (2021) explores shuffling n -grams for different values of n , which highlights the insensitivity of pretrained

Transformer models. [Sinha et al. \(2020\)](#) explores shuffling of words on textual entailment tasks, highlighting models’ insensitivity to such perturbations. Finally, [Taktasheva et al. \(2021\)](#) extend perturbation studies to Swedish and Russian and performs perturbations by shuffling syntactic phrases, rotating sub-trees around the root of the syntactic tree of a sentence, or simply shuffling the words of the text.

These approaches share the main limitation of requiring automatic parsing tools or well-developed tokenizers to define words. This limits their applicability in a multilingual setting. Priors regarding the form of the text, such as the presence of white-space delimited words, limit the generalizability of most of these studies.

[Cloutre et al. \(2022\)](#) proposes a suite of controllable perturbations on characters and subwords, which should be compatible with almost any written language, as well as a metric quantifying perturbations to the local and global structure that measures perturbations on a character-level.

3 Experiments

We extend the perturbation studies of [Cloutre et al. \(2022\)](#) to a multilingual setting. We perform those experiments on eight popular cross-lingual tasks ([Hu et al., 2020](#); [Ponti et al., 2020](#); [Liang et al., 2020](#)) covering over 120 languages. This will shed light on what languages, if any, do not share the same sensitivity to local structure and insensitivity to global structure as English.

3.1 Metric and Perturbations

The **CHRF-2** (chrF) ([Popović, 2015](#)) metric measures the amount of character bi-gram overlap between a perturbed text and the original text. This measure represents the amount of *local* structure that has not been perturbed in a text.

The **Index Displacement Count (IDC)** ([Cloutre et al., 2022](#)) metric measures the average absolute distance traversed by every character in a perturbed text. An IDC of 0.3 would mean that, on average, every character has traversed 30% of the length of the text. This measure represents the amount of *global* perturbations applied to a text.

The **compression rate** (Comp) ([Xue et al., 2021](#)) represents the total length of the text in terms of characters divided by the total length of the text once tokenized. Since most of our models either use subwords or tokenize characters directly, there

are no out-of-vocabulary tokens to be counted. The compression rate is then used as a proxy for vocabulary destruction of pretrained models, an important confounder for the importance of local structure.

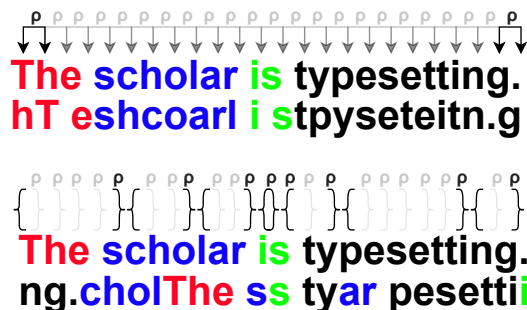


Figure 1: From top to bottom: Neighbor Flipping with $\rho = 0.5$, Phrase Shuffling with $\rho = 0.5$

We perform perturbations by altering the order of **subwords** and **characters** present in the text. Three types of perturbations are applied.

Full shuffling completely randomizes the order of the subword or characters.

Neighbor flipping flips a subword or character with its neighbor with a controllable probability ρ , providing local perturbations while maintaining much of the absolute position of the tokens.

Phrase shuffling randomly builds phrases of subwords or characters of controllable average length with a parameter ρ and shuffles those phrases, providing a minimal amount of local perturbations for a large amount of change in absolute position.

Simple examples of those perturbations are shown in Figure 1, pseudocode and details are present in the Appendix B.

Task	n Languages	Task Type	Metric
PAWS-X	7	Paraphrase Detection	ACC
XNLI	15	NLI	ACC
QAM	3	Text Classification	ACC
QADSM	3	Text Classification	ACC
WPR	7	Page Ranking	nDCG
XCopa	11	Commonsense Reasoning	ACC
BUCC	5	Sentence Retrieval	F1
Tatoeba	122	Sentence Retrieval	F1

Table 1: Summary information of the different tasks used.

3.2 Experimental Details

All experiments are conducted on three pretrained cross-lingual models. The XLM-RoBERTa-Base ([Lample and Conneau, 2019](#)), BERT-Base-Multilingual-Cased ([Devlin et al., 2019](#)) and the

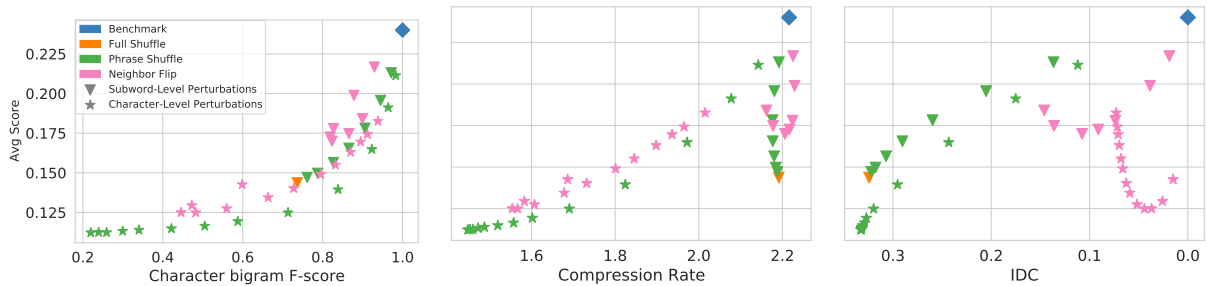


Figure 2: Plotted are the relations between the different choices of metrics measuring the amount of perturbation and the average performance of all 3 models on all tested datasets. Left is more perturbed, up is better performance. The X-axis of the IDC metric is inverted for clearer comparison.

Canine-S (Clark et al., 2021) model are used. The Canine model is a tokenization-free pretrained model, which lets us isolate the impact of subword destruction on the findings.

The zero-shot cross-lingual setting (Hu et al., 2020) is used for all experiments. The model is first finetuned on the English version of the dataset and evaluated without further tuning on all target languages.

The English version on which the model is finetuned is kept unperturbed, while the target language text on which the model is evaluated goes through several perturbations. We perform a total of 43 different perturbations on every task and language and obtain their performance. All models are finetuned on five different random seeds, and all perturbations are performed on five different random seeds, for a total of 25 evaluations for every model on every task, every language present in the tasks, and every perturbation setting.¹

A total of 8 cross-lingual tasks selected from the most popular cross-lingual benchmarks (Hu et al., 2020; Liang et al., 2020; Ponti et al., 2020) covering over 120 languages are used for evaluation.² Summary information of the tasks can be found in Table 1.³

3.3 Results and Discussion

In Figure 2, we observe the trends reported by Cloutre et al. (2022) to be broadly true in a cross-

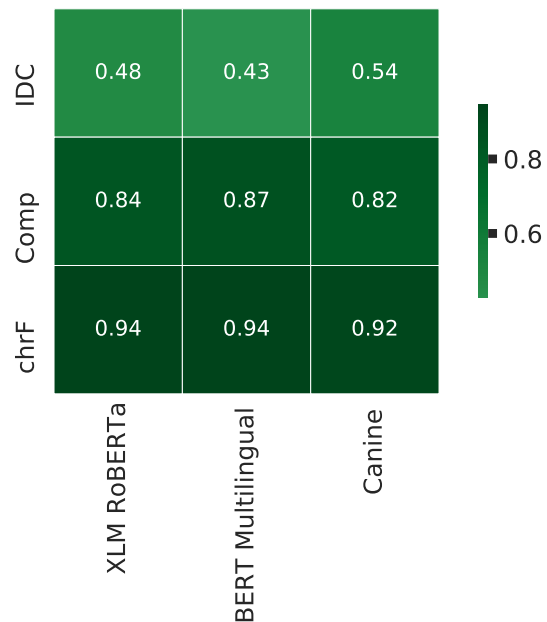


Figure 3: Rank-correlation matrix between the different models’ performance to perturbed samples on the and the perturbation quantified by the different metrics. The higher the value the better the metric explains the degradation in performance.

lingual setting. Specifically, the more local perturbations are applied to a text, the more degradation in the understanding of that text can be expected, which shows that model does rely on the local structure to build understanding. The perturbations to the global structure are shown to be a much poorer explanation for the degradation in performance than the perturbation to the local structure. The compression rate is highly correlated with a model’s performance and the local structure, making it a potential confounder for the degradation in performance. However, the trend in local structure holds with subword-level perturbations, unlike with the compression rate, which is not affected by

¹Detailed training and testing hyperparameters and process are present in the Appendix A and details on the specific perturbations in Appendix A.

²Extractive tasks such as extractive QA are not compatible with our perturbations, as the answer would also be perturbed and were not considered.

³As we use all 122 languages in the Tatoeba dataset, which vary from 100 to 1000 possible sentences to retrieve, the F1 score is more appropriate as an evaluation of performance than the accuracy used in the XTREME benchmark.

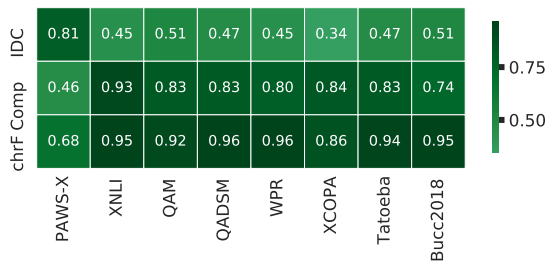


Figure 4: Rank-correlation matrix between the different task’s performance to perturbed samples and the perturbation quantified by the different metrics. The higher the value the better the metric explains the degradation in performance.

perturbations to the order of subwords, as well as holding for the vocabulary-free Canine model, as shown in Figure 3. This makes it more likely that the cause for the degradation in performance is the local structure perturbation, the destruction of the vocabulary being incidental.

3.3.1 PAWS-X

Figure 4 shows the rank-correlations of a model’s performance over the different tasks with the different measures of perturbation. The overall trends are stable in all but one task, PAWS-X. Much like the CoLA task (Warstadt et al., 2019) in the GLUE Benchmark (Wang et al., 2019), it is possible to build tasks that require the specific order of words to be successfully completed. The PAWS-X task comprises adversarial paraphrases containing a similar lexicon between paraphrase and non-paraphrases. The performance is then highly sensitive to perturbations causing displacement, such as shuffling words, even if the local structure is mostly kept intact. It is not that local structure is unnecessary, but that global structure is. This phenomenon is further explored by Mahowald et al. (2022); Ravishankar et al. (2022); Papadimitriou et al. (2022).

3.3.2 Chinese Character Script

Figure 5 show that the findings are consistent across almost all text scripts, with the exception of languages using Chinese Characters as script.

This is most likely caused by how semantically richer the smallest separable unit in Chinese tends to be compared to characters in different scripts. Where Chinese has a single indivisible character meaning "water" the English equivalent "water" can be perturbed to "rtawe". Even character-level shuffling cannot strip Chinese text of all meaning,

which would explain some the differences. It is to be noted that while weaker, the correlation between local structure perturbations and performance remains high.

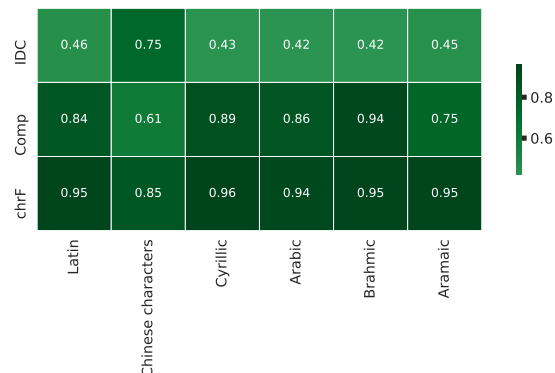


Figure 5: Rank-correlation matrix between the different language script’s containing at least 3 languages performance to perturbed samples on the and the perturbation quantified by the different metrics. The higher the value the better the metric explains the degradation in performance.

4 Conclusion

We first explored and confirmed the importance of local structure, the limited importance of global structure, and controlled for the potential of vocabulary destruction being the main explanatory factor in 8 NLU tasks covering over 120 languages. In aggregate, the findings of Cloutre et al. (2022) hold for many different pretrained cross-lingual models and NLU tasks in a multilingual setting. Local structure sensitivity and global structure insensitivity do not seem to be an artifacts of the English language.

A significant exception is when grammatical cues are essential to complete the task, such as in the PAWS-X task. While many tasks can be solved purely with the information obtained from the local structure, reasoning over the global context is necessary for many problems.

Languages using Chinese characters as their script also deviate from the norm. This is likely caused by how semantically rich their characters are.

It will be important that any NLP improvements derived from English experiments are verified to also generalize to other languages. As we have observed that languages written in Chinese Character Script are differently impacted by perturbations to different coding properties, it is possible that im-

provements to the way our model understand those properties in English will not generalize.

Acknowledgements

This research has been funded by the NSERC Discovery Grant Program.

References

- Wissam Antoun, Fady Baly, and Hazem M. Hajj. 2020. Arabert: Transformer-based model for arabic language understanding. *ArXiv*, abs/2003.00104.
- Diedre Carmo, Marcos Piau, Israel Campiotti, Rodrigo Nogueira, and Roberto de Alencar Lotufo. 2020. PTT5: pretraining and validating the T5 model on brazilian portuguese data. *CoRR*, abs/2008.09144.
- Jonathan H. Clark, Dan Garrette, Iulia Turc, and John Wieting. 2021. Canine: Pre-training an efficient tokenization-free encoder for language representation.
- Louis Clouatre, Prasanna Parthasarathi, Amal Zouaq, and Sarath Chandar. 2022. Local structure matters most: Perturbation study in NLU. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3712–3731, Dublin, Ireland. Association for Computational Linguistics.
- Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for chinese BERT. *CoRR*, abs/1906.08101.
- Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. Bertje: A dutch BERT model. *CoRR*, abs/1912.09582.
- J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.
- Ashim Gupta, Giorgi Kvernadze, and Vivek Sriku-mar. 2021. Bert & family eat word salad: Experiments with text understanding. *arXiv preprint arXiv:2101.03453*.
- Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization.
- Artur Kulmizev and Joakim Nivre. 2021. Schrödinger’s tree - on syntax and neural language models. *CoRR*, abs/2110.08887.
- Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *CoRR*, abs/1901.07291.
- Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoît Crabbé, Laurent Besacier, and Didier Schwab. 2019. Flaubert: Unsupervised language model pre-training for french. *CoRR*, abs/1912.05372.

- Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenge Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Bruce Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Rangan Majumder, and Ming Zhou. 2020. [XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation](#). *CoRR*, abs/2004.01401.
- Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. [Multilingual denoising pre-training for neural machine translation](#). *CoRR*, abs/2001.08210.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [Roberta: A robustly optimized bert pretraining approach](#). *ArXiv*, abs/1907.11692.
- Kyle Mahowald, Evgeniia Diachek, Edward Gibson, Evelina Fedorenko, and Richard Futrell. 2022. [Grammatical cues are largely, but not completely, redundant with word meanings in natural language](#).
- Martin Malmsten, Love Börjeson, and Chris Haffenden. 2020. [Playing with words at the national library of sweden - making a swedish BERT](#). *CoRR*, abs/2007.01658.
- Louis Martin, Benjamin Müller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villamonte de la Clergerie, Djamel Seddah, and Benoît Sagot. 2019. [Camembert: a tasty french language model](#). *CoRR*, abs/1911.03894.
- Francis Mollica, Matthew Siegelman, Evgeniia Diachek, Steven T. Piantadosi, Zachary Mineroff, Richard Futrell, Hope Kean, Peng Qian, and Evelina Fedorenko. 2020. [Composition is the Core Driver of the Language-selective Network](#). *Neurobiology of Language*, 1(1):104–134.
- Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. [PhoBERT: Pre-trained language models for Vietnamese](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1037–1042, Online. Association for Computational Linguistics.
- Joe O’Connor and Jacob Andreas. 2021. [What context features can transformer language models use?](#) In *ACL/IJCNLP*.
- Isabel Papadimitriou, Richard Futrell, and Kyle Mahowald. 2022. [When classifying grammatical role, bert doesn’t care about word order... except when it matters](#).
- Thang M. Pham, Trung Bui, Long Mai, and Anh M Nguyen. 2021. [Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks?](#) *ArXiv*, abs/2012.15180.
- Marco Polignano, Pierpaolo Basile, Marco Degemmis, Giovanni Semeraro, and Valerio Basile. 2019. [Alberto: Italian bert language understanding model for nlp challenging tasks based on tweets](#). In *CLiC-it*.
- Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. [Xcopa: A multilingual dataset for causal common-sense reasoning](#).
- Maja Popović. 2015. [chrF: character n-gram F-score for automatic MT evaluation](#). In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.
- Vinit Ravishankar, Mostafa Abdou, Artur Kulmizev, and Anders Søgaard. 2022. [Word order does matter \(and shuffled language models know it\)](#).
- Chinnadhurai Sankar, Sandeep Subramanian, Chris Pal, Sarath Chandar, and Yoshua Bengio. 2019. [Do neural dialog systems use the conversation history effectively? an empirical study](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 32–37, Florence, Italy. Association for Computational Linguistics.
- 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](#). *arXiv preprint arXiv:2104.06644*.
- Koustuv Sinha, Prasanna Parthasarathi, Joelle Pineau, and Adina Williams. 2020. [Unnatural language inference](#). *arXiv preprint arXiv:2101.00010*.
- Ekaterina Taktasheva, Vladislav Mikhailov, and Ekaterina Artemova. 2021. [Shaking syntactic trees on the sesame street: Multilingual probing with controllable perturbations](#). *CoRR*, abs/2109.14017.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. [GLUE: A multi-task benchmark and analysis platform for natural language understanding](#). In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.
- Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. [Neural network acceptability judgments](#). *Transactions of the Association for Computational Linguistics*, 7:625–641.
- Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2021. [Byt5: Towards a token-free future with pre-trained byte-to-byte models](#). *CoRR*, abs/2105.13626.
- Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. [mt5: A massively multilingual pre-trained text-to-text transformer](#). *CoRR*, abs/2010.11934.

A Experiment Details

Model Hyperparameters and Training We finetune each pretrained models on the English version of each dataset for a total of 10 epochs, checkpointing the model after each epochs. The English version is never perturbed, the finetuning is done on unperturbed data. This finetuning is done 5 times with different random seeds for each model and each datasets. For 8 datasets and 3 models we have a total of $3 * 8 * 5 = 120$ finetuning and 1200 checkpoints, one for each epoch. A learning rate of $2e-5$, a batch size of 32 and a weight decay of 0.1 is used in all finetuning. All experiments used a warmup ratio of 0.06, as described in Liu et al. (2019).

For the evaluation, we perform the same perturbations on the validation and testing data of the different target languages. We evaluate the perturbed validation data on each of the 10 checkpoints, chose the best checkpoint on the perturbed validation data, and evaluate that checkpoint on the perturbed test data. This process is repeated for each perturbations, each of the 5 random seed and 5 times with different perturbation random seeds for each finetuned models. In total, for each language in each task on each model for each perturbation setup we average results over 25 random seeds.

For the sentence retrieval tasks, such as Tatoeba, we do not perform any finetuning. We simply obtain the nearest neighbour using cosine similarity on the final hidden representation. (Hu et al., 2020) First, we obtain the representation of the unperturbed English side of the dataset. This is done by feeding the English text through the model and averaging the final layers hidden representation of the text. We then perform our perturbations on the target language text, feed those perturbed text through the same pretrained cross-lingual model and obtain it’s representation through the same process. We now have a set of English representation and a set of target language representation, on which we find the nearest neighbour as measured by the Cosine Distance on the pooled hidden representations. If the nearest neighbour is the sentence that was to be retrieved, we consider this an hit, else it is a miss. The reported results are over the average of 5 random seeds of those perturbations.

Perturbations A total of 43 perturbations are used for all experiments. The first one is the Benchmark, which is simply the unperturbed text. We perform a full-shuffling on both the subwords and characters. On the subword-level perturbations we perform phrase-shuffling with ρ values of: [0.9, 0.8, 0.65, 0.5, 0.35, 0.2, 0.1] and neighbour-flip shuffling with ρ values of: [0.9, 0.8, 0.6, 0.5, 0.4, 0.2, 0.1]. On the character-level perturbations we perform phrase-shuffling with ρ values of: [0.975, 0.95, 0.9, 0.8, 0.65, 0.5, 0.4, 0.3, 0.2, 0.15, 0.1, 0.075, 0.05] and neighbour-flip shuffling with ρ values of: [0.8, 0.65, 0.5, 0.4, 0.3, 0.2, 0.1, 0.075, 0.05, 0.035, 0.025, 0.01]. A total of 15 subword-level experiments, 27 character-level experiments and the unperturbed benchmark are evaluated for a grand total of 43 different perturbation settings .

B Pseudocode for Metric and Perturbations

```

Function IDC ( $X_p$ ):
   $X_p^{len} \leftarrow X_p.length()$ ;
  IDC_list  $\leftarrow list()$ 
  for  $i \leftarrow 0$  and  $i \leq X_p^{len}$  do
    | abs_distortion  $\leftarrow abs(i - X_p[i])$ ;
    | IDC_list.append(abs_distortion);
  end
  IDC_agg  $\leftarrow IDC\_list.mean()$ ;
  IDC  $\leftarrow \frac{IDC\_agg}{X_p^{len}}$ ;
return

```

Algorithm 1: Pseudocode to compute IDC metric.

```

Function PhrasePerturbation ( $\rho \leftarrow 0.5, text \leftarrow list$ ):
  all_phrases  $\leftarrow list()$ ;
  phrase  $\leftarrow list(text[0])$ 
  for token in text[1:] do
    |  $p \sim Unif([0, 1])$ ;
    | if  $p < \rho$  then
      | | all_phrases.append(phrase);
      | | phrase  $\leftarrow list(token)$ 
    | else
      | | phrase  $\leftarrow [phrase, token]$ ;
    | end
  end
  all_phrases.append(phrase);
  perturbed_text  $\leftarrow \text{.join}(shuffle(all\_phrases))$ 
return perturbed_text

```

Algorithm 2: Pseudocode for PhraseShuffle.

```

Function NeighborFlip ( $\rho \leftarrow 0.5, text \leftarrow list$ ):
  perturbed_tokens  $\leftarrow list()$ ;
  held_token  $\leftarrow list(text[0])$ 
  for token in text[1:] do
    |  $p \sim Unif([0, 1])$ ;
    | if  $p < \rho$  then
      | | perturbed_tokens.append(held_token);
      | | held_token  $\leftarrow list(token)$ 
    | else
      | | perturbed_tokens  $\leftarrow [perturbed\_tokens, token]$ ;
    | end
  end
  perturbed_tokens.append(held_token);
  perturbed_text  $\leftarrow \text{.join}(perturbed\_tokens)$ 
return perturbed_text

```

Algorithm 3: Pseudocode for NeighborFlip.

C Additional Results

Language Family Figure 6 shows the aggregated correlations between the different language families and the different metrics. Results seem to be consistent across all families, with the exception of Sino-Tibetan languages. This was generally addressed in Section 3.3.2.

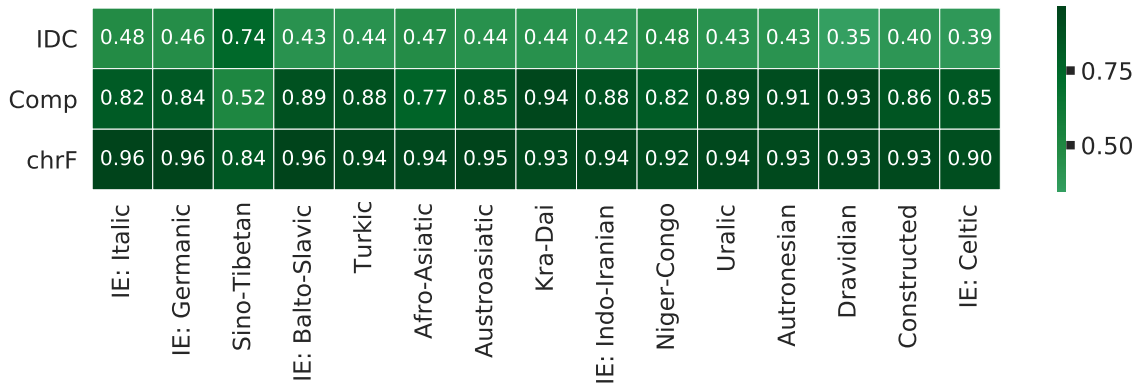


Figure 6: Rank-correlation matrix between the different language family’s containing at least 3 languages performance to perturbed samples on the and the perturbation quantified by the different metrics. The higher the value the better the metric explains the degradation in performance.

PAWS-X To determine whether it is that the local structure is not essential on PAWS-X, or simply that perturbations to the order of words are equally important, we observe the performance of models using only neighbor flipping perturbations, limiting the displacement of words to a minimum. In Figure 7, we show that if we only perturb the local structure, performance is highly correlated with the amount of local perturbations. This implies that it is not that the model is insensitive to local perturbations, rather for certain tasks where grammatical queues are necessary any change to the order of words will lead to failure.

Chinese Character Script Languages using Chinese characters and derivatives obtain a relatively weaker correlation with local perturbations. Figure 8 illustrates the perturbation to performance curve while only taking into account languages using Chinese characters as their script, compared to those using the Latin script in Figure 9.

A few major divergences from the global trend are present. First, the average compression ratio is under 1, meaning that the tokenizer *adds* to the sequence length on average. While counter-intuitive, this is caused by the fact that the vast majority of Chinese characters’ tokenization defaults to tokenizing the character directly, thus yielding almost no compression. The tokenizer adds a few special characters for the Transformer model to use, yielding *longer* sequences on average than the raw text. This can be verified by the fact that, unlike with other scripts, subword-perturbations are sufficient to explore almost the whole spectrum of local perturbations, which would only be possible if most subwords were of length 1.

While the phrase shuffling perturbations seem to behave as expected, it seems that text written in chinese script are especially resilient to neighbour flipping. We compare the performance of Chinese character scripts and Latin scripts in Figure 9 and find that Chinese scripts are, on average, more resilient to perturbations, going from an average score of 0.18 to 0.08 while the Latin Script performance drops all the way to an aggregate score of 0.03.

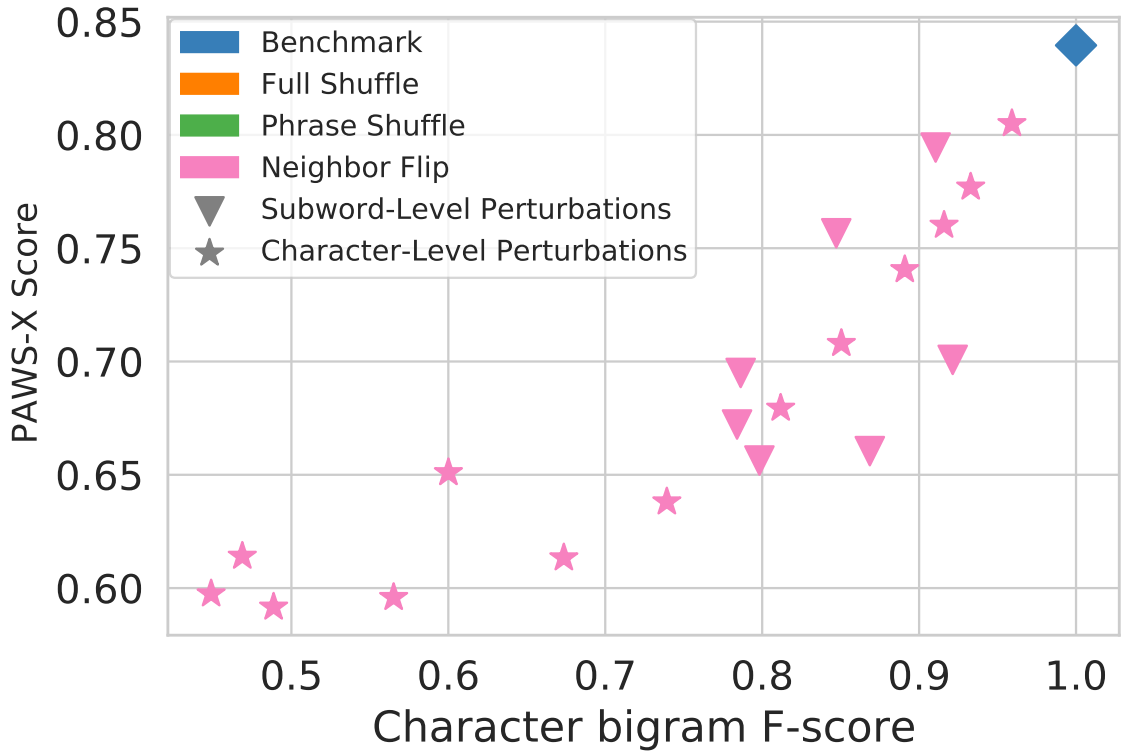


Figure 7: Plotted is the relations between the local structure perturbation and the average performance on the PAWS-X dataset. Only the neighbour flipped perturbations are shown to isolate the impact of perturbations to the local structure.

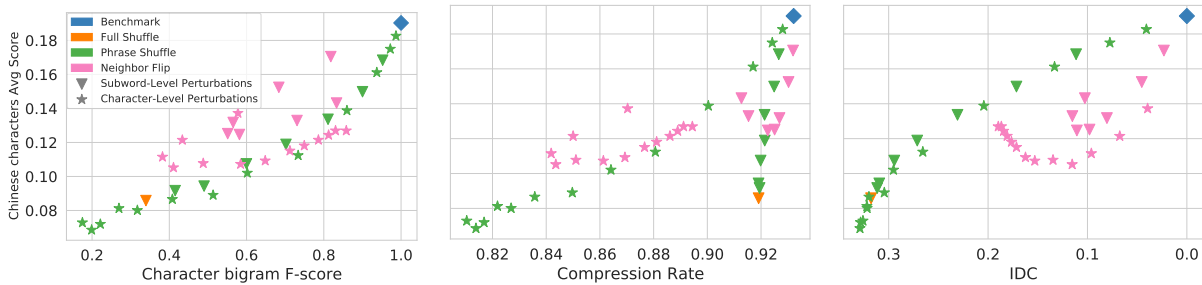


Figure 8: Plotted are the relations between the different metrics measuring the amount of perturbation and the average performance of all 3 models on all tested datasets on languages using chinese characters or derivatives as their scripts.

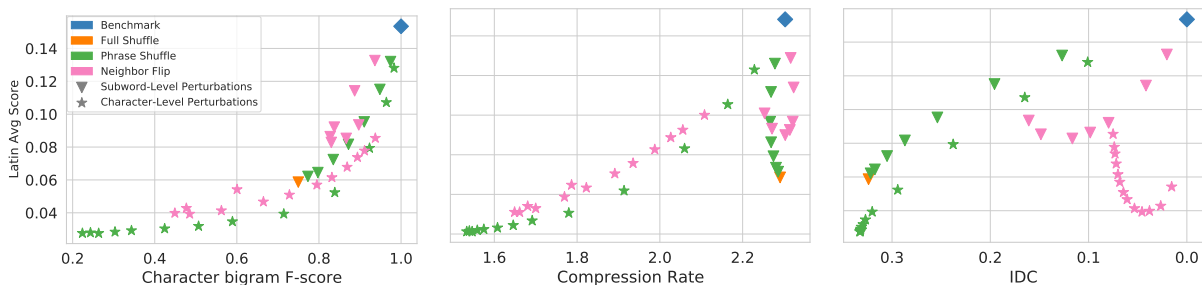


Figure 9: Plotted are the relations between the different metrics measuring the amount of perturbation and the average performance of all 3 models on all tested datasets on languages using a latin script.