

# We’re Calling an Intervention: Exploring Fundamental Hurdles in Adapting Language Models to Nonstandard Text

Aarohi Srivastava and David Chiang

Computer Science and Engineering

University of Notre Dame

Notre Dame, IN, USA

{asrivas2, dchiang}@nd.edu

## Abstract

We present a suite of experiments that allow us to understand the underlying challenges of language model adaptation to nonstandard text. We do so by designing *interventions* that approximate core features of user-generated text and their interactions with existing biases of language models. Applying our interventions during language model adaptation to nonstandard text variations, we gain important insights into when such adaptation is successful, as well as the aspects of text variation and noise that are particularly difficult for language models to handle. For instance, on text with character-level variation, out-of-the-box performance improves even with a few additional training examples but approaches a plateau, suggesting that more data is not the solution. In contrast, on text with variation involving new words or meanings, far more data is needed, but it leads to a massive breakthrough in performance. Our findings reveal that existing models lack the necessary infrastructure to handle diverse forms of nonstandard text, guiding the development of more resilient language modeling techniques. We make the code for our interventions, which can be applied to any English text data, publicly available.

## 1 Introduction

Nonstandard text is all around us. Whether a user adopts a regional dialect, follows different spelling conventions, or uses culturally-specific vocabulary, encountering text variation in most day-to-day NLP use cases is inevitable (Blodgett et al., 2016; Huang et al., 2020). Yet, recent work continues to find large gaps in performance between standard and nonstandard text and speech (Kantharuban et al., 2023; Faisal et al., 2024), and efforts to reduce this gap are typically case-, variety-, or task-dependent (Held et al., 2023; Joshi et al., 2024). The distinction between standard and nonstandard text is often contextual. In written English, *going to* is standard,

while *gonna* or *gunna* is nonstandard. Likewise, *I am not* (standard) contrasts with *I ain’t* (non-standard). Similarly, *color* (American English) vs. *colour* (British English) reflects orthographic variation. Whether deploying a large-scale system for diverse users or working towards language model personalization, it is important to understand the challenges of adapting a language model to different varieties. Specifically, how can we improve a model’s performance in a new domain (in this case, a new variety) by leveraging knowledge from an existing one (i.e., the standard one)?

This is tricky; it is difficult to tease apart the interactions between the complex and intertwined features that comprise linguistic variation, and the black-box nature of language models makes it even more difficult to do so. But we can reach some firmer conclusions if, on the one hand, we can devise ways of controlling different levels of text variation, and, on the other hand, we study the structure of the model (as opposed to its parameter values). We find that the model structure itself induces biases towards the standard variety of a language, and it does so in different ways at different levels of linguistic structure.

Current language models like BERT (Devlin et al., 2019) and GPT (Radford et al., 2019) are actually a hybrid of two models: a frequency-based subword tokenizer and a Transformer-based encoder or decoder. The tokenizer determines subword units, and the Transformer embeds the subwords into a vector space where it operates on the vectors. Frequent words are often kept together as a single token, while infrequent words are often broken up into several shorter-length tokens. When both the tokenizer and the Transformer are biased towards standard text, a rigid relationship develops between tokens and their vector representations. Small changes to the token sequence (e.g., resulting from spelling variation) can break the model’s ability to understand it properly (Kumar et al., 2020; Soper et al.,

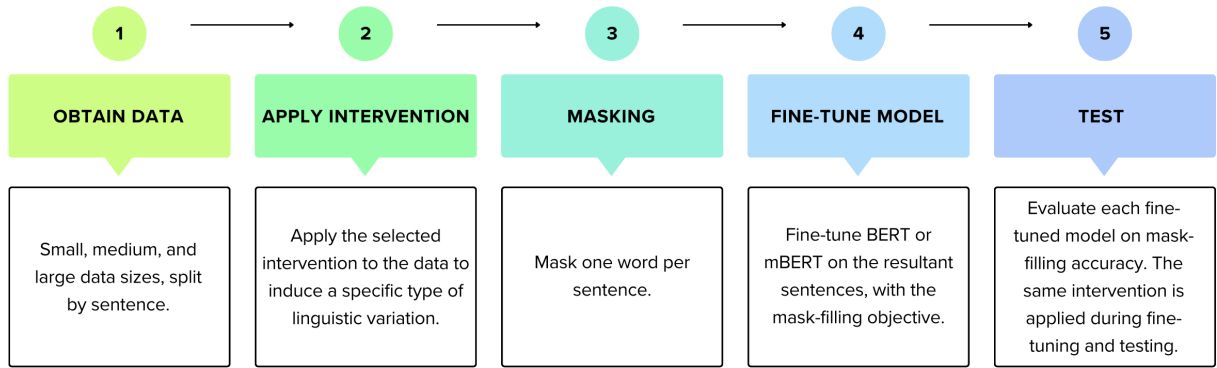


Figure 1: A sketch of our train/test pipeline.

2021; Blaschke et al., 2023; Srivastava and Chiang, 2023a; Chai et al., 2024). See the example below, where  $\overset{\text{tok}}{\rightsquigarrow}$  denotes tokenization:

$\text{coffee} \overset{\text{tok}}{\rightsquigarrow} \text{coffee}$   
 $\text{cofee} \overset{\text{tok}}{\rightsquigarrow} \text{co, fe, e}$

While the model would associate the token *coffee* with the appropriate word, one misspelling in the input means the model must figure out that the token sequence *co, fe, e* refers to the same thing as *coffee*, which is certainly not trivial. Moreover, since the subword tokenizer of a pre-trained model cannot be modified without large-scale retraining, such issues can typically only be addressed through model adaptation methods (e.g., fine-tuning). However, because this approach does not resolve the underlying problem, it is crucial to explore the challenges and success cases in adapting language models to different forms of nonstandard text.

To this end, we develop synthetic manipulations that exploit our knowledge of what happens when a Transformer-based language model interacts with nonstandard text. Our experiments isolate data-related factors that can play a role in language model adaptation (e.g., type, amount, and composition of training data), and we assemble a suite of nine interventions to synthetically induce tokenization and embedding disruption, grounded in traits of user-generated text and at different levels of linguistic structure (e.g., orthographic, morphological, lexical), in controlled settings. We make the code for these interventions publicly available.<sup>1</sup> Our experiments (outlined in Figure 1) evaluate and stress-test BERT’s ability to adapt to various types of nonstandard text under different conditions. Our findings inform important questions about the fundamental hurdles of adapting language models

to nonstandard text:

- Language models adapt more effectively to lexical variation at the subword and multi-subword levels (e.g., new words, word senses, and meanings) during fine-tuning but struggle with within-subword variation (e.g., character-level changes, unconventional spellings).
- When handling lexical variation, greater data availability is vital for successful knowledge transfer. Multilingual models are also more helpful in such cases.
- In contrast, for text with character-level variation (within-subword changes), increasing data offers limited benefits. Instead, achieving robustness likely requires alternative solutions. Monolingual models outperform multilingual ones in such cases.

## 2 Related Work

As larger language models with new capabilities emerge, engaging with nonstandard text (e.g., dialects, language varieties, noisy text) – a hallmark of content generated by today’s diverse user base – remains a challenge. Language models are not robust to noisy text, plummeting in performance when faced with seemingly simple issues like misspellings, typos, and grammatical errors, even though such issues naturally arise in almost any use case (Kumar et al., 2020; Yin et al., 2020; Aspillaga et al., 2020). These limitations have been documented even in large models. For instance, Pagnoni et al. (2024) find that Llama 3 and 3.1 perform poorly in robustness tests involving character-level noise, and Chai et al. (2024) highlight the sensitivity of large language models to character-level variations.

Improving model performance on nonstandard text is not trivial. For instance, Faisal et al. (2024)

<sup>1</sup><https://github.com/aarsri/interventions-linguistic-variation>

document a persistent performance gap between standard and dialectal text, even after in-variety fine-tuning. Approaches to increasing robustness to linguistic variation, though often successful, tend to be highly dependent on the language variety and task, typically requiring data for each dialect in question (Held et al., 2023) and exposing differences in performance across tasks and varieties (Srivastava and Chiang, 2023b). Given the vast, unpredictable nature of user-generated text, which often involves multiple types of linguistic variation at once, these challenges become even more pronounced. Thus, it is essential to take a step back and develop a deeper understanding of why such adaptation remains difficult and what is needed to facilitate effective learning.

Past work has explored challenges of cross-lingual transfer (Philippy et al., 2023). For instance, Wu et al. (2023) investigate three key factors – embedding space disruption, tokenization changes, and word order changes – by transforming GLUE datasets to induce each type of shift. They find that while language models can adapt to tokenization and word order changes, they struggle to recover from embedding space disruption (e.g., learning new alignments). Similar conclusions are drawn by Deshpande et al. (2022) and Jain et al. (2022); Deshpande et al. (2022) highlight the importance of subword overlap and token embedding alignment for successful knowledge transfer, while Jain et al. (2022) suggest leveraging word alignment and dictionary matching techniques. Building on these findings, our experiments examine core features of nonstandard text and explore their interactions with these modeling challenges.

### 3 Levels of Text Variation

Variation is a natural response to the productive nature of language. It can be observed at all levels of linguistic reasoning (Haber, 1976; Geeraerts et al., 1994): character-level change (e.g., spelling, abbreviation, orthography), morphological and syntactic variation (i.e., word and sentence structure), lexical and semantic variation (new word senses and meanings), and variation in style and tone, often related to language-external sociopolitical factors. Style and tone variation operate at a higher level of reasoning, integrating features from the first three categories. Viewing user-generated text in terms of these core features elucidates the challenges language models face in adapting to its variability.

When dealing with nonstandard text, we encounter infrequent strings far more often, which inherently challenges existing biases of language models and impacts downstream performance. The tokenizer produces longer sequences of tokens comprised of shorter subwords, a phenomenon called *oversegmentation* (Soper et al., 2021; Srivastava and Chiang, 2023a). Shorter subwords can appear in many more contexts and take on different meanings each time, a phenomenon called *subword replacement* (Srivastava and Chiang, 2023a). Even in the case of new word senses/meanings, for which the token sequence may not change as much, the model would lack the relevant knowledge.

Drawing on documented features of user-generated text, we have devised a suite of 9 interventions that operate at different levels of linguistic structure. We apply these interventions to all text used in our experiments. The interventions fall into four categories: character-level change, subword boundary manipulation, morphological variation, and lexical variation. All the interventions require the model to learn to map between standard (seen) and nonstandard (rare or unseen) text for effective knowledge transfer. However, differences in how the interventions affect tokenization can influence the difficulty of learning these mappings during fine-tuning. For this reason, the interventions are designed to cause a fair amount of disruption, with some acting as stress tests. All of our data and experiments are in English, but the algorithmic nature of many of our interventions can be extended to other languages.

#### 3.1 Character-Level Change

Character-level change can arise for several reasons, including phonological influences (e.g., accent, sound change), new methods of writing (e.g., social media), and typological errors (Condorelli and Rutkowska, 2023). We include two interventions under this category; the first is a reflection of phonological variation, and the second is more of a stress test.

1. **IPA:** Letters corresponding to consonants on the IPA chart that form a minimal feature pair (voiced vs. unvoiced) are swapped. For example, *p* and *b*, which differ only in the voicing feature but share the same place and manner of articulation, are swapped. Changes to consonants are a hallmark of modern English orthographic variation, particularly observed on social media (Eisenstein, 2015; Ilbury, 2020).

For instance, such variations are prevalent in corpora of noisy text like MultiLexNorm (Van Der Goot et al., 2021); examples include dese/these, dey/they, smyle/smile, and sadest/saddest.

boots  $\xrightarrow{\text{IPA}}$  poodz  $\xrightarrow{\text{tok}}$  p, ood, z

2. **Shift:** A Caesar cipher is applied to the letters of the alphabet:  $a \rightarrow b, b \rightarrow c, \dots, z \rightarrow a$ . This intervention is the most extreme form of orthographic change, in which every alphabetic symbol is renamed. It roughly approximates situations where a language variety uses a different alphabet than the standard variety or where social media trends using symbols that resemble letters. **Shift** serves as a stress test, as well as a means of comparison to milder interventions like **IPA**.

boots  $\xrightarrow{\text{Shift}}$  cpput  $\xrightarrow{\text{tok}}$  c, pp, ut

While orthographic changes like the ones above would ordinarily only require us to learn to map one character to another (or one spelling of a word to another, as in texting abbreviations), the structure imposed by the model makes this task more complex. As exemplified above, adapting to this category of variation would require the model to learn a one-to-many mapping from a single token to a list of tokens.

### 3.2 Subword Boundaries

As demonstrated above, linguistic variation often results in changes to subword token boundaries, which triggers a domino effect and ultimately results in lower quality contextual embeddings assigned to nonstandard text by the model (Kumar et al., 2020; Soper et al., 2021). Because of this, we include three interventions that overtly manipulate subword boundaries, two of which preserve the original spelling of the word and only split tokens. Successful adaptation would once again involve learning a one-to-many mapping, but with more of the surface level appearance (i.e., spelling) intact. Examples like “ammmazing” (amazing) in MultiLexNorm (Van Der Goot et al., 2021) resemble this situation.

1. **Reg:** Subword regularization using the Max-Match Dropout (Hiraoka, 2022) method for BERT’s WordPiece algorithm is applied with a dropout of 0.5. This means 50% of the subword vocabulary is randomly dropped with each tokenization call.

boots  $\xrightarrow{\text{tok}}$  boots  
boots  $\xrightarrow{\text{Reg}}$  boot, s

2. **Char:** Subword regularization using the Max-Match Dropout (Hiraoka, 2022) method for WordPiece is applied with a dropout of 1, meaning each letter of a word is its own token.

boots  $\xrightarrow{\text{Char}}$  b, o, o, t, s

3. **Pig:** Words are converted to Pig Latin, in which the word-initial consonant(s) is moved to the end, and a suffix (*ay* or *yay*) is added. For example, *pig latin* becomes *igpay atinlay*.

boots  $\xrightarrow{\text{Pig}}$  ootsbay  $\xrightarrow{\text{tok}}$  o, ots, bay

### 3.3 Morphological Variation

In English, morphological variation can occur in inflectional or derivational affixes (Neef, 2009; Zanuttini and Horn, 2014). Inflectional endings are morphemes with grammatical functions that typically change grammatical features like part of speech and number (e.g., plural *-s*). This is a type of morphosyntactic variation, which relates to grammatical acceptability and is often highly stigmatized – some utterances may be grammatical in one variety and not in the standard, or vice versa. In contrast, derivational affixes are used to change the meaning of a word, which may or may not also change its part of speech. Unlike inflectional endings, which are a functional category, derivational affixes have much more room for variation.

1. **–End:** Using MorphyNet (Batsuren et al., 2021), inflectional endings are dropped from words that have them.

boots  $\xrightarrow{\text{–End}}$  boot  $\xrightarrow{\text{tok}}$  boot

2. **Affix:** Using MorphyNet (Batsuren et al., 2021), derivational prefixes and suffixes are mapped cyclically. For instance, *non-* becomes *ab-*, *ab-* becomes *pre-*, etc. *Nonsense* is now *absense* (not “absence”), *absence* is now *presence*, and so on. In this way, we tinker with part of a word but change its whole meaning.

nonsense  $\xrightarrow{\text{Affix}}$  absense  $\xrightarrow{\text{tok}}$  a, bs, ense

Because the lemma is preserved in these cases, learning the appropriate knowledge will require a mix of recovering token mappings and transferring existing knowledge about the meaning of the word from the lemma.



There are several examples of morphosyntactic variation found in user-generated text (Zanuttini and Horn, 2014). A common feature in MultiLexNorm (Van Der Goot et al., 2021) seems to be informal contractions (e.g., “imma,” “finna,” “tryna,” “hella”). Because these forms have lower preservation of the lemma, the model may also see them as new words, bringing us to the final category: lexical variation.

### 3.4 Lexical Variation

Word-level variation can be just that – a new word introduced to refer to a specific, perhaps novel, referent, or to be used in a new context. But lexical variation can often introduce semantic variation; the typical process being that a word’s senses expand and eventually shift to the new meaning, through specialization, generalization, or subjectification (Kakharova, 2021; Geeraerts et al., 2024). Such occurrences are extremely common on social media and in colloquial settings.

1. **Hyp:** Exemplifying specialization, through which a word’s range of reference narrows, words with hyponyms found in WordNet (Princeton University, 2010) are replaced by their hyponym.

boot  $\overset{\text{Hyp}}{\rightsquigarrow}$  buskin  $\overset{\text{tok}}{\rightsquigarrow}$  bus, kin

2. **Ant:** Exemplifying subjectification, through which a word’s meaning becomes more positive (ameliorization) or negative (pejorization), words with antonyms found in WordNet (Princeton University, 2010) are replaced by their antonym.

nice  $\overset{\text{Ant}}{\rightsquigarrow}$  nasty

With the lexical interventions, the model must learn to map between contextual meanings of seen words. Variation is at the subword or multi-subword level.

### 3.5 Interventions in Action

Through these four categories of interventions, we are able to examine a range of underlying factors in modeling and adaptation capability for nonstandard text, from tokenization-specific issues (character-level change and subword boundaries), to token-embedding relationships (morphological variation), to contextual representation shift (lexical variation). Moreover, within each category, we include milder, more realistic interventions (e.g., **IPA**) and stress tests (e.g., **Shift**). Our experiments reveal whether a model has a chance at recovering each type of

Size	Sentences	Word Count Quartiles			
		Average	$P_{25}$	$P_{50}$	$P_{75}$
<b>S</b>	264	19.9	10	18	26
<b>M</b>	2641	18.5	10	16	24
<b>L</b>	26415	18.4	10	16	24

Table 1: Sentence and word count statistics ( $25^{th}$ ,  $50^{th}$ , and  $75^{th}$  percentiles) for each data split.

mapping during fine-tuning, and under which data-related conditions.

## 4 Data Preparation

To compare the model’s ability to adapt to each synthetic variety, we fine-tune and test it with the mask-filling objective on text with the appropriate intervention applied. We provide a sketch of our train/test pipeline in Figure 1. All of our data is sourced from Wikicorpus (Reese et al., 2010) to reduce external effects of choice of data. We reserve half the Wikicorpus articles for fine-tuning and half for testing. These are separated into sentences using the sentence tokenizer from NLTK (Loper and Bird, 2002). In our experiments, a *word* is any string that satisfies Python’s `isalpha` function and is an element of NLTK word tokenizer’s output. Sentences with 0 or 1 words are eliminated.

When fine-tuning, we vary the amount of data used. Each split is a fixed set of sentences sampled from the fine-tuning articles. The number of sentences in each split covers three orders of magnitude. We report statistics in Table 1 on sentence length (measured by number of *words*) for each split to demonstrate that they are comparable in this regard. In addition to varying the data size and intervention, we also vary the composition of the fine-tuning data. It could be *mixed*, meaning the intervention is only applied to half the sentences, or *full*, meaning the intervention is applied to all sentences.

Fine-tuning typically focuses on adapting a model to a specific task, making it challenging to simultaneously adapt the model to a new language variety. Task-specific factors can also complicate this process, especially in experiments involving disrupted text. For example, some tasks, like intent classification, rely on identifying key words in the input, while others, like sentiment analysis, require an overall understanding of the sentence, and more linguistically complex tasks, such as linguistic acceptability, demand deeper grammatical reasoning. These variations create a complex interaction between handling perturbed text and meeting the

requirements of each task, potentially introducing bias into the experiment. To avoid these complications, we select mask-filling as our fine-tuning task. Mask-filling aligns with the models’ pre-training objective, allowing fine-tuning resources to focus exclusively on adapting to the new language variety without interference from task-specific factors.

Our masking policy during fine-tuning is as follows. One *word* per sentence is randomly selected to be masked. We practice whole-word masking, meaning the model could be asked to fill one or more consecutive mask tokens, given the number of subword tokens that comprise the masked word. Depending on the word to be masked, it is possible the intervention does not actually change that word. While this is natural for fine-tuning, to make sure the comparison during testing is fair, we mandate that the word masked at test time is actually modified by all nine interventions. This filtering yields a test set of 931 sentences sampled from the testing articles. For each sentence, the same word is masked in all interventions/tests for consistency.

## 5 Experiments

There are four axes of variation in our experiments: *amount* of fine-tuning data (small, medium, large), *composition* of fine-tuning data (50% (mixed) or 100% (full) of sentences are noised), *intervention* to be applied (9 total), and *multilinguality* (monolingual or multilingual pre-trained model). We use BERT-base-cased<sup>2</sup> (BERT) as the monolingual English model, and BERT-base-multilingual-cased<sup>3</sup> (mBERT) as the multilingual model. We do not include character-level models like CharacterBERT (El Boukkouri et al., 2020) and CANINE (Clark et al., 2022) in our experiments; unlike BERT and mBERT, they cannot be used out-of-the-box for mask-filling and would not yield comparable results. For larger models, it is often unclear whether success stems from pre-training exposure, parameter-based knowledge retention, or true adaptation (our focus). Using BERT, we minimize such confounds.

For each possible combination of the four axes of variation, a pre-trained model is fine-tuned on the corresponding data with the masked language modeling objective. As described in Section 4, one word per sentence is masked (whole word masking), and the fine-tuning objective is to fill the masked

token(s) with the tokens of the original word.

We use Low-Rank Adaptation (LoRA, Hu et al. (2021)) for parameter-efficient fine-tuning, which adapts the attention weights of each Transformer encoder layer and freezes all other parameters. We follow Hu et al.’s guidelines for hyperparameter choice, using the AdamW optimizer (Loshchilov and Hutter, 2018) with a linear scheduler, LoRA rank of 8, and LoRA scaling factor  $\alpha$  of 8. We use learning rate  $7 \cdot 10^{-4}$  for the small data amount and  $5 \cdot 10^{-4}$  for the medium and large data amounts. On an NVIDIA A10 Tensor Core GPU, fine-tuning takes 12 seconds/epoch for the small data amount, 2 minutes/epoch for the medium data amount, and 18 minutes/epoch for the large data amount. LoRA was chosen over standard fine-tuning for two key reasons: (1) standard fine-tuning is highly susceptible to distribution shift issues, and (2) LoRA provides greater control over where learning occurs (in encoder parameters associated with attention).

We measure performance with three metrics: from most to least strict, exact match, 1-best, and 5-best accuracy. Exact match accuracy measures for how many masked words each token of the word is predicted correctly, divided by the test set size. 1-best accuracy measures the total number of masked tokens filled correctly (by the top probability prediction) divided by the total number of masked tokens in the test set. Similarly, 5-best accuracy measures the total number of masked tokens whose top five predictions include the correct answer, divided by the total number of masked tokens in the test set. The 1-best accuracy metric provides the best summary of the results Table 2; we include results using the other two metrics in Tables 5 and 6.

## 6 Results

The main 1-best results for our experiments are found in Table 2, and the normalized scores are reported in Table 3. Additional results using the other metrics are included in Tables 4-6. We reiterate the categories of interventions (see Section 3) below:

- Character-Level Change: **IPA, Shift**
- Subword Boundary Variation: **Reg, Char, Pig**
- Morphological Variation: **–End, Affix**
- Lexical Variation: **Hyp, Ant**

### 6.1 Baselines: We need an intervention!

We include a baseline row for each model (data amount 0), in which the pre-trained model, without any additional fine-tuning, is tested on each inter-

<sup>2</sup><https://huggingface.co/bert-base-cased>

<sup>3</sup><https://huggingface.co/bert-base-multilingual-cased>

vention. We also include a baseline intervention *None*, in which the models are evaluated on the original text without any intervention applied.

As expected, monolingual BERT is better out-of-the-box at the mask-filling task on English Wikipedia text. Fine-tuning without any intervention results in overfitting – an expected outcome, as the model has already been trained on the same task with similar data. Thus, we also include relative performance – each baseline score in Table 2 is normalized by the corresponding *None* performance and expressed as a percentage in Table 3. Out-of-the-box performance (data amount 0) is extremely low across the board, demonstrating that our experiments will provide clear findings as BERT cannot already solve these tasks.

## 6.2 Mixed vs. Full Composition: What is needed to learn new mappings?

The knowledge transfer that takes place when adapting a model to a new variety of the language is akin to learning how to map elements of the standard variety to the new one. One of the varied parameters of our experiments is whether the fine-tuning data is mixed (intervention is only applied to 50% of sentences) or full (intervention is applied to all sentences). Sometimes, we might expect model learning to benefit from seeing both standard and nonstandard versions of text during fine-tuning, while other times, this only makes learning the appropriate patterns more difficult.

Our results indicate that the latter holds true when dealing with linguistic variation; average performance (not including baseline scores) is about 4.2 points, or 36%, higher, with the full composition (Table 2) rather than the mixed composition (Table 4). Performance with full composition is also higher when comparing averages for each intervention category, so the mixed composition does not provide an advantage for any type of variation tested. Because of this difference in performance, we refer only to the full composition results when discussing and analyzing the remaining factors below.

## 6.3 Data Size: When is more data needed?

Across our experiments, particularly looking at the normalized performance relative to the baseline intervention scores (Table 3), more fine-tuning data helps. At the same time, the utility of adding more fine-tuning data differs depending on the type of mapping needed for a class of interventions.

When it comes to the orthographic interventions (**IPA**, **Shift**) and **–End**, the input sequence is affected by tokenization-related issues (i.e., over-segmentation and subword replacement), and the model must learn one-to-many mappings between tokens during fine-tuning for successful knowledge transfer. In this case, simply fine-tuning, even on the small or medium data amounts, results in a big improvement, but further improvements are much smaller. While fine-tuning helps, it is not sufficient to fully recover the mapping due to the model’s inability to draw on sub-atomic knowledge (i.e., within-subword character-level information).

Subword boundary manipulation still involves the issues of oversegmentation, but the closeness in spelling may result in a stronger association by the model between the contextual representations of pre- and post-intervention versions of the same words. When adapting to the synthetic varieties with subword boundary manipulation, performance improves gradually as more data is added.

Strikingly, there is an apparent breakthrough effect when fine-tuning with the largest data size for the lexical interventions (**Hyp**, **Ant**) and **Affix**. While the performance attained is comparatively low for the small and medium data sizes, there is a massive jump when the large data size is used for fine-tuning. This is most clearly observed in the exact-match performance (Table 5) for BERT and the relative performance for mBERT, which nears or exceeds 100% in these three tasks. These tasks require relearning the usage of words in new contexts. Mapping word-level information (i.e., new spellings) as well as contextual meaning evidently requires much more data, but when the data requirement is satisfied, the model is capable of recovering (multi-) subword level information.

## 6.4 Monolingual vs. Multilingual: Which type of knowledge is more helpful?

For the most part, the average performance (not including baseline performance) is extremely close between BERT and mBERT. Notably, mBERT provides a substantial advantage for adapting to variation in meaning (**Affix**, **Hyp**, and **Ant**), providing a 20% boost in absolute scores (Table 2) and a 74% relative improvement in normalized scores (Table 3). Because mBERT is trained on several languages, it is likely not as rigid in terms of relating words to specific meanings/contexts, providing an advantage here.

		Interventions									
Model	Data	None	IPA	Shift	Reg	Char	Pig	−End	Affix	Hyp	Ant
BERT	0	58.8	2.6	2.3	7.0	5.4	9.3	1.5	0.1	0.1	0.2
	S	52.4	3.4	5.6	9.0	12.2	15.3	32.3	0.3	2.0	2.6
	M	48.6	15.3	8.7	12.7	16.1	14.8	28.9	14.5	9.5	19.1
	L	47.7	18.8	9.5	11.6	26.5	23.1	37.6	35.7	29.6	29.6
mBERT	0	42.0	2.6	2.2	5.1	6.4	13.8	2.6	1.7	1.7	1.1
	S	38.8	10.4	4.4	4.7	11.6	13.6	23.1	1.7	5.2	7.1
	M	31.6	12.1	3.7	12.1	20.5	16.7	30.5	18.8	17.6	19.7
	L	34.4	13.2	5.2	11.1	29.7	21.3	30.0	41.7	28.9	33.6

Table 2: 1-best accuracy results (single run) for all experiments with the full data composition using the base version of the model. Data amount 0 denotes the out-of-the-box baseline performance compared to fine-tuning with the small (S), medium (M), or large (L) data sizes.

		Interventions									
Model	Data	None	IPA	Shift	Reg	Char	Pig	−End	Affix	Hyp	Ant
BERT	0	100.0	4.4	3.8	11.9	9.2	15.7	2.6	0.2	0.1	0.3
	S	100.0	6.5	10.8	17.2	23.3	29.2	61.7	0.5	3.8	4.9
	M	100.0	31.5	18.0	26.1	33.1	30.5	59.4	29.8	19.6	39.3
	L	100.0	39.4	19.9	24.4	55.4	48.3	78.8	74.9	62.0	62.1
mBERT	0	100.0	6.1	5.3	12.1	15.1	32.8	6.2	4.1	4.0	2.5
	S	100.0	26.9	11.4	12.1	30.0	35.2	59.6	4.3	13.3	18.3
	M	100.0	38.4	11.6	38.1	64.9	52.9	96.4	59.5	55.5	62.4
	L	100.0	38.4	15.0	32.2	86.2	61.8	87.0	120.9	83.9	97.6

Table 3: Relative performance (single run) normalized by the baseline (*None*), as percentages using the base version of the model. Data amount 0 denotes the out-of-the-box baseline performance compared to fine-tuning with the small (S), medium (M), or large (L) data sizes.

## 6.5 Implications for User-Generated Text

Our interventions provide insights into how language models adapt to the diverse noise patterns in user-generated text. We examine cases where token sequences are disrupted but meaning is preserved (e.g., character-level changes) and those where meanings shift with minimal impact on tokenization (e.g., lexical variation), as well as intermediate cases like subword boundary manipulation.

Our findings show that models struggle to process variations within subwords. Given the prevalence of character-level changes in user-generated text, particularly on social media, fine-tuning on additional data alone is insufficient for robust adaptation. This poses challenges for NLP applications in these domains, especially when users prefer to retain their stylistic choices rather than conform to standardized text. At the same time, we find that with sufficient data, models can learn certain morphological and lexical variations, making it possible to adapt to new words, slang, and evolving usage patterns. However, given the diversity of user-generated content, some phenomena may be too sparse for effective learning, highlighting the need for more targeted approaches

beyond data scaling.

## 7 Conclusion

We introduce a suite of interventions that synthetically modify English text to analyze interactions between features of nonstandard and user-generated text with underlying biases of language models. Our experiments isolate data-related factors that can contribute to language model adaptation, revealing critical insights into the limitations of adapting language models to nonstandard text. We explore several cases, ranging from character-level changes that over-segment the token sequence to lexical variations that alter contextual representations. Some interventions require learning one-to-many mappings within subwords, while others demand associations across multiple subwords.

Our findings highlight key adaptation challenges. BERT-like models struggle to adapt to character-level changes, even with additional data, but can successfully handle lexical shifts and new word senses with enough exposure to relevant fine-tuning data. Notably, interventions involving affix changes and antonym substitutions achieve performance com-



parable to or exceeding mBERT baselines. This suggests that while models can effectively learn word-to-word mappings, structural constraints hinder their ability to process finer-grained variations within subwords or at the character level. These limitations arise from how tokens are segmented and represented within the model, restricting its ability to capture within-subword (sub-atomic) variations.

Ultimately, while current models adjust well to meaning-related variations given enough data, they struggle with fine-grained structural disruptions, where constituents are smaller than a subword (e.g., character-level). Despite the prevalence of such variation (e.g., social media), current language models lack the capability to facilitate the required flexibility between tokens and embeddings for straightforward adaptation methods like fine-tuning, unless the relevant knowledge is already incorporated during pre-training. This underscores the need for more flexible tokenization and modeling approaches, especially for handling the complexities of user-generated text and enabling the model to effectively capture within-subword (e.g., character-level) information.

## Limitations

While our experiments explored numerous possibilities within the scope of our study, they are certainly not exhaustive. We recognized the vast array of potential variations and interventions that could be considered and aimed to curate a feasible selection that still offered a diverse representation of linguistic challenges and adaptation scenarios. Beyond the selection of interventions to design, it would be valuable to expand the scope of experiments in the other dimensions, as well: languages, data sizes, and models.

For instance, our interventions and experiments were developed and executed in English. While narrowing down the language dimension allowed us to develop the highly controlled experiments needed for this study, it also means insights are missing for languages in which the character-, subword-, and word-level paradigm used in this work may not apply in the same way. Examples include languages with richer morphology (e.g., Turkic languages) and languages for which characters correspond to syllables, words, or concepts (e.g., Sino-Tibetan languages). Similarly, while the code for the interventions can largely be extended to other Indo-European languages, it would require more modification before

it could be used for other language families.

Furthermore, while our choices for the small, medium, and large data sizes are informed by runtime and typical fine-tuning data sizes in NLP work, some of our results point to trends as data size increases. As a result, it would be valuable to extend these tables to see an even bigger picture as data size continues to increase. In addition, while we used LoRA for model adaptation in this study, there are other approaches that could be explored to better understand the nuances of parameter-efficient fine-tuning.

Finally, our work focuses on the widely-used BERT, which is an Encoder-only language model of relatively small size. While the small size is beneficial for our study, as it assures us that the model has not already seen the intervention tasks during pre-training, diversity in the type of model (e.g., encoder-decoder, decoder-only, LLMs) can help paint a bigger picture for our results and their implications in a wider range of settings.

## References

- Carlos Aspillaga, Andrés Carvallo, and Vladimir Araujo. 2020. [Stress test evaluation of Transformer-based models in natural language understanding tasks](#). In *Proc. LREC*, pages 1882–1894.
- Khuyagbaatar Batsuren, Gábor Bella, and Fausto Giunchiglia. 2021. [MorphyNet: a large multilingual database of derivational and inflectional morphology](#). In *Proc. SIGMORPHON*, pages 39–48.
- Verena Blaschke, Hinrich Schütze, and Barbara Plank. 2023. [Does manipulating tokenization aid cross-lingual transfer? A study on POS tagging for non-standardized languages](#). In *Proc. VarDial*, pages 40–54.
- Su Lin Blodgett, Lisa Green, and Brendan O’Connor. 2016. [Demographic dialectal variation in social media: A case study of African-American English](#). In *Proc. EMNLP*, pages 1119–1130.
- Yekun Chai, Yewei Fang, Qiwei Peng, and Xuhong Li. 2024. [Tokenization falling short: On subword robustness in large language models](#). In *Findings of the ACL: EMNLP*, pages 1582–1599.
- Jonathan H Clark, Dan Garrette, Iulia Turc, and John Wieting. 2022. [CANINE: Pre-training an efficient tokenization-free encoder for language representation](#). *Transactions of the Association for Computational Linguistics*, 10:73–91.
- Marco Condorelli and Hanna Rutkowska. 2023. *The Cambridge Handbook of Historical Orthography*. Cambridge University Press.

- Ameet Deshpande, Partha Talukdar, and Karthik Narasimhan. 2022. [When is BERT multilingual? isolating crucial ingredients for cross-lingual transfer](#). In *Proc. NAACL HLT*, pages 3610–3623.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proc. NAACL-HLT*, pages 4171–4186.
- Jacob Eisenstein. 2015. [Systematic patterning in phonologically-motivated orthographic variation](#). *Journal of Sociolinguistics*, 19(2):161–188.
- Hicham El Boukkouri, Olivier Ferret, Thomas Lavergne, Hiroshi Noji, Pierre Zweigenbaum, and Jun’ichi Tsujii. 2020. [CharacterBERT: Reconciling ELMo and BERT for word-level open-vocabulary representations from characters](#). In *Proc. ICLR*, pages 6903–6915.
- Fahim Faisal, Orevaoghene Ahia, Aarohi Srivastava, Kabir Ahuja, David Chiang, Yulia Tsvetkov, and Antonios Anastasopoulos. 2024. [DIALECTBENCH: An NLP benchmark for dialects, varieties, and closely-related languages](#). In *Proc. ACL*, pages 14412–14454.
- Dirk Geeraerts, Stefan Grondelaers, and Peter Bakema. 1994. *The structure of lexical variation: Meaning, naming, and context*. 5. Walter de Gruyter.
- Dirk Geeraerts, Dirk Speelman, Kris Heylen, Mariana Montes, Stefano De Pascale, Karlien Franco, and Michael Lang. 2024. [Lexical Variation and Change: A Distributional Semantic Approach](#). Oxford University Press.
- Lyn R. Haber. 1976. [Leaped and leapt: A theoretical account of linguistic variation](#). *Foundations of Language*, pages 211–238.
- William Held, Caleb Ziems, and Diyi Yang. 2023. [TADA: Task agnostic dialect adapters for English](#). In *Findings of the ACL*, pages 813–824.
- Tatsuya Hiraoka. 2022. [MaxMatch-Dropout: Subword regularization for WordPiece](#). In *Proc. ICLR*, pages 4864–4872.
- Edward J. Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. [LoRA: Low-rank adaptation of large language models](#). In *Proc. ICLR*.
- Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. 2020. [Challenges in building intelligent open-domain dialog systems](#). *ACM Transactions on Information Systems (TOIS)*, 38(3):1–32.
- Christian Ilbury. 2020. [“Sassy Queens”: Stylistic orthographic variation in Twitter and the enregisterment of AAVE](#). *Journal of Sociolinguistics*, 24(2):245–264.
- Neel Jain, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. 2022. [How to do a vocab swap? A study of embedding replacement for pre-trained transformers](#).
- Aditya Joshi, Raj Dabre, Diptesh Kanojia, Zhuang Li, Haolan Zhan, Gholamreza Haffari, and Doris Dippold. 2024. [Natural language processing for dialects of a language: A survey](#). *ACM Computing Surveys*.
- Nilufar Nuridinovna Kakharova. 2021. [On the nature of linguistic variation and its types](#). *Asian Journal of Research in Social Sciences and Humanities*, 11(11):507–510.
- Anjali Kantharuban, Ivan Vulić, and Anna Korhonen. 2023. [Quantifying the dialect gap and its correlates across languages](#). In *Findings of the ACL: EMNLP*, pages 7226–7245.
- Ankit Kumar, Piyush Makhija, and Anuj Gupta. 2020. [Noisy text data: Achilles’ heel of BERT](#). In *Proc. W-NUT*, pages 16–21.
- Edward Loper and Steven Bird. 2002. [NLTK: The natural language toolkit](#). In *Proc. ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, pages 63–70.
- Ilya Loshchilov and Frank Hutter. 2018. [Decoupled weight decay regularization](#). In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Martin Neef. 2009. [Morphological variation: A declarative approach](#). In *Describing and Modeling Variation in Grammar*, pages 117–133. Mouton de Gruyter Berlin.
- Artidoro Pagnoni, Ram Pasunuru, Pedro Rodriguez, John Nguyen, Benjamin Muller, Margaret Li, Chunting Zhou, Lili Yu, Jason Weston, Luke Zettlemoyer, et al. 2024. [Byte latent transformer: Patches scale better than tokens](#). *arXiv preprint arXiv:2412.09871*.
- Fred Philippy, Siwen Guo, and Shohreh Haddadan. 2023. [Towards a common understanding of contributing factors for cross-lingual transfer in multilingual language models: A review](#). In *Proc. ACL*.
- Princeton University. 2010. [About WordNet](#).
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. [Language models are unsupervised multitask learners](#).
- Samuel Reese, Gemma Boleda, Montse Cuadros, Lluís Padró, and German Rigau. 2010. [Wikicorpus: A word-sense disambiguated multilingual Wikipedia corpus](#). In *Proc. LREC*.
- Elizabeth Soper, Stanley Fujimoto, and Yen-Yun Yu. 2021. [BART for post-correction of OCR newspaper text](#). In *Proc. W-NUT*, pages 284–290.
- Aarohi Srivastava and David Chiang. 2023a. [BERTwch: Extending BERT’s capabilities to model dialectal and noisy text](#). In *Findings of the ACL: EMNLP*, pages 15510–15521.

- Aarohi Srivastava and David Chiang. 2023b. [Fine-tuning BERT with character-level noise for zero-shot transfer to dialects and closely-related languages](#). In *Proc. VarDial*, pages 152–162.
- Rob Van Der Goot, Alan Ramponi, Arkaitz Zubiaga, Barbara Plank, Benjamin Muller, Iñaki San Vicente Roncal, Nikola Ljubešić, Özlem Çetinoğlu, Rahmad Mahendra, Talha Çolakoglu, et al. 2021. [MultiLexNorm: A shared task on multilingual lexical normalization](#). In *Proc. W-NUT*, pages 493–509. Association for Computational Linguistics.
- Zhengxuan Wu, Alex Tamkin, and Isabel Papadimitriou. 2023. [Oolong: Investigating what makes transfer learning hard with controlled studies](#). In *Proc. EMNLP*, pages 3280–3289.
- Fan Yin, Quanyu Long, Tao Meng, and Kai-Wei Chang. 2020. [On the robustness of language encoders against grammatical errors](#). In *Proc. ACL*.
- Raffaella Zanuttini and Laurence Horn. 2014. *Micro-syntactic Variation in North American English*. Oxford University Press.

## A Appendix

		Interventions									
Model	Data	None	IPA	Shift	Reg	Char	Pig	–End	Affix	Hyp	Ant
BERT	0	58.8	2.6	2.3	6.9	5.4	9.3	1.5	0.1	0.1	0.2
	S	52.4	2.4	5.9	11.0	11.7	17.4	18.5	0.8	0.6	0.8
	M	48.6	7.5	8.3	12.5	13.1	16.3	26.7	10.6	8.3	3.0
	L	47.7	14.4	8.3	12.0	19.9	20.3	15.1	25.5	17.0	19.7
mBERT	0	42.0	2.6	2.2	5.1	6.4	13.8	2.6	1.7	1.7	1.1
	S	38.8	10.5	1.5	7.4	12.3	10.1	5.2	4.2	1.3	6.8
	M	31.6	9.3	4.8	12.8	14.1	16.2	7.6	24.0	9.7	8.5
	L	34.4	14.7	6.8	13.0	16.4	16.9	11.9	29.5	25.3	22.8

Table 4: 1-best accuracy results (single run) for all experiments with the mixed data composition using the base version of the model. Data amount 0 denotes the out-of-the-box baseline performance compared to fine-tuning with the small (S), medium (M), or large (L) data sizes.

		Interventions									
Model	Data	None	IPA	Shift	Reg	Char	Pig	–End	Affix	Hyp	Ant
BERT	0	62.0	0.1	0.0	12.1	0.0	0.0	1.6	0.1	0.0	0.2
	S	55.1	0.3	0.0	9.9	0.5	0.0	34.2	0.1	1.8	3.2
	M	51.3	2.4	2.3	12.5	1.2	0.2	30.7	3.2	10.1	18.1
	L	50.3	9.8	2.2	10.2	7.3	7.5	39.0	25.2	31.2	31.5
mBERT	0	44.8	1.2	0.9	6.8	0.9	1.4	2.9	0.5	1.7	0.8
	S	40.7	1.2	0.4	1.8	0.8	0.9	25.5	0.4	7.0	10.4
	M	33.6	2.7	0.2	9.9	8.1	2.8	33.3	9.0	13.5	17.1
	L	37.0	5.3	0.8	8.5	8.9	7.7	32.2	31.0	27.7	30.5

Table 5: Exact match accuracy results (single run) for all experiments with the full data composition using the base version of the model. Data amount 0 denotes the out-of-the-box baseline performance compared to fine-tuning with the small (S), medium (M), or large (L) data sizes.

		Interventions									
Model	Data	None	IPA	Shift	Reg	Char	Pig	–End	Affix	Hyp	Ant
BERT	0	77.0	10.5	10.0	13.9	26.4	23.9	21.5	2.2	1.2	0.5
	S	74.3	17.1	17.3	23.5	46.6	33.1	47.1	2.0	6.9	8.5
	M	74.9	29.3	28.3	27.9	55.6	40.2	46.6	31.7	23.8	31.0
	L	62.6	35.2	33.5	27.6	60.0	51.8	55.8	52.6	45.2	45.9
mBERT	0	64.1	6.9	8.6	14.3	25.3	27.4	21.5	4.1	2.6	2.3
	S	63.5	21.2	14.3	15.4	44.9	32.8	36.9	5.5	11.2	16.1
	M	61.6	24.9	14.4	29.3	55.2	35.2	46.9	31.2	32.5	36.6
	L	49.0	25.2	19.3	23.7	62.4	44.7	43.7	56.8	44.7	47.7

Table 6: 5-best accuracy results (single run) for all experiments with the full data composition using the base version of the model. Data amount 0 denotes the out-of-the-box baseline performance compared to fine-tuning with the small (S), medium (M), or large (L) data sizes.