Superbizarre Is Not Superb: Derivational Morphology Improves BERT's Interpretation of Complex Words

Valentin Hofmann[*‡], **Janet B. Pierrehumbert**[†*], **Hinrich Schütze**[‡]

[*]Faculty of Linguistics, University of Oxford
[†]Department of Engineering Science, University of Oxford
[‡]Center for Information and Language Processing, LMU Munich
valentin.hofmann@ling-phil.ox.ac.uk

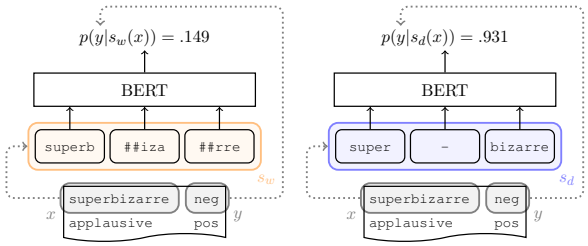
Abstract

How does the input segmentation of pretrained language models (PLMs) affect their interpretations of complex words? We present the first study investigating this question, taking BERT as the example PLM and focusing on its semantic representations of English derivatives. We show that PLMs can be interpreted as serial dual-route models, i.e., the meanings of complex words are either stored or else need to be computed from the subwords, which implies that maximally meaningful input tokens should allow for the best generalization on new words. This hypothesis is confirmed by a series of semantic probing tasks on which DelBERT (Derivation leveraging BERT), a model with derivational input segmentation, substantially outperforms BERT with WordPiece segmentation. Our results suggest that the generalization capabilities of PLMs could be further improved if a morphologically-informed vocabulary of input tokens were used.

1 Introduction

Pretrained language models (PLMs) such as BERT (Devlin et al., 2019), GPT-2 (Radford et al., 2019), XLNet (Yang et al., 2019), ELECTRA (Clark et al., 2020), and T5 (Raffel et al., 2020) have yielded substantial improvements on a range of NLP tasks. What linguistic properties do they have? Various studies have tried to illuminate this question, with a focus on syntax (Hewitt and Manning, 2019; Jawahar et al., 2019) and semantics (Ethayarajh, 2019; Ettinger, 2020; Vulić et al., 2020).

One common characteristic of PLMs is their input segmentation: PLMs are based on fixed-size vocabularies of words and subwords that are generated by compression algorithms such as byte-pair encoding (Gage, 1994; Sennrich et al., 2016) and WordPiece (Schuster and Nakajima, 2012; Wu et al., 2016). The segmentations produced by these algorithms are linguistically questionable at times (Church, 2020), which has been shown to worsen performance on certain downstream tasks (Bostrom and Durrett, 2020; Hofmann et al., 2020a). However, the wider implications of these findings, particularly with regard to the generalization capabilities of PLMs, are still poorly understood.



(a) BERT (s_w) (b) DelBERT (s_d)

Figure 1: Basic experimental setup. BERT with WordPiece segmentation (s_w) mixes part of the stem `bizarre` with the prefix `super`, creating an association with `superb` (left panel). DelBERT with derivational segmentation (s_d), on the other hand, separates prefix and stem by a hyphen (right panel). The two likelihoods are averaged across 20 models trained with different random seeds. The average likelihood of the true class is considerably higher with DelBERT than with BERT. While `superbizarre` has negative sentiment, `applausive` is an example of a complex word with positive sentiment.

Here, we address a central aspect of this issue, namely how the input segmentation affects the semantic representations of PLMs, taking BERT as the example PLM. We focus on derivationally complex words such as `superbizarre` since they exhibit systematic patterns on the lexical level, providing an ideal testbed for linguistic generalization. At the same time, the fact that low-frequency and out-of-vocabulary words are often derivationally complex (Baayen and Lieber, 1991) makes our work relevant in practical settings, especially when many one-word expressions are involved, e.g., in query processing (Kacprzak et al., 2017).

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 3594–3608
August 1–6, 2021.

The topic of this paper is related to the more fundamental question of how PLMs represent the meaning of complex words in the first place. So far, most studies have focused on methods of representation extraction, using ad-hoc heuristics such as averaging the subword embeddings (Pinter et al., 2020; Sia et al., 2020; Vulić et al., 2020) or taking the first subword embedding (Devlin et al., 2019; Heinzerling and Strube, 2019; Martin et al., 2020). While not resolving the issue, we lay the theoretical groundwork for more systematic analyses by showing that PLMs can be regarded as serial dual-route models (Caramazza et al., 1988), i.e., the meanings of complex words are either stored or else need to be computed from the subwords.

Contributions. We present the first study examining how the input segmentation of PLMs, specifically BERT, affects their interpretations of derivationally complex English words. We show that PLMs can be interpreted as serial dual-route models, which implies that maximally meaningful input tokens should allow for the best generalization on new words. This hypothesis is confirmed by a series of semantic probing tasks on which derivational segmentation substantially outperforms BERT's WordPiece segmentation. This suggests that the generalization capabilities of PLMs could be further improved if a morphologically-informed vocabulary of input tokens were used. We also publish three large datasets of derivationally complex words with corresponding semantic properties.[1]

2 How Are Complex Words Processed?

2.1 Complex Words in Psycholinguistics

The question of how complex words are processed has been at the center of psycholinguistic research over the last decades (see Leminen et al. (2019) for a recent review). Two basic processing mechanisms have been proposed: *storage*, where the meaning of complex words is listed in the mental lexicon (Manelis and Tharp, 1977; Butterworth, 1983; Feldman and Fowler, 1987; Bybee, 1988; Stemberger, 1994; Bybee, 1995; Bertram et al., 2000a), and *computation*, where the meaning of complex words is inferred based on the meaning of stem and affixes (Taft and Forster, 1975; Taft, 1979, 1981, 1988, 1991, 1994; Rastle et al., 2004; Taft, 2004; Rastle and Davis, 2008).

In contrasting with *single-route* frameworks, *dual-route* models allow for a combination of storage and computation. Dual-route models are further classified by whether they regard the processes of retrieving meaning from the mental lexicon and computing meaning based on stem and affixes as *parallel*, i.e., both mechanisms are always activated (Frauenfelder and Schreuder, 1992; Schreuder and Baayen, 1995; Baayen et al., 1997, 2000; Bertram et al., 2000b; New et al., 2004; Kuperman et al., 2008, 2009), or *serial*, i.e., the computation-based mechanism is only activated when the storage-based one fails (Laudanna and Burani, 1985; Burani and Caramazza, 1987; Caramazza et al., 1988; Burani and Laudanna, 1992; Laudanna and Burani, 1995; Alegre and Gordon, 1999).

Outside the taxonomy presented so far are recent models that assume multiple levels of representation as well as various forms of interaction between them (Rácz et al., 2015; Needle and Pierrehumbert, 2018). In these models, sufficiently frequent complex words are stored together with representations that include their internal structure. Complex-word processing is driven by analogical processes over the mental lexicon (Rácz et al., 2020).

2.2 Complex Words in NLP and PLMs

Most models of word meaning proposed in NLP can be roughly assigned to either the single-route or dual-route approach. Word embeddings that represent complex words as whole-word vectors (Deerwester et al., 1990; Mikolov et al., 2013a,b; Pennington et al., 2014) can be seen as single-route storage models. Word embeddings that represent complex words as a function of subword or morpheme vectors (Schütze, 1992; Luong et al., 2013) can be seen as single-route computation models. Finally, word embeddings that represent complex words as a function of subword or morpheme vectors as well as whole-word vectors (Botha and Blunsom, 2014; Qiu et al., 2014; Bhatia et al., 2016; Bojanowski et al., 2017; Athiwaratkun et al., 2018; Salle and Villavicencio, 2018) are most closely related to parallel dual-route approaches.

Where are PLMs to be located in this taxonomy? PLMs represent many complex words as whole-word vectors (which are fully stored). Similarly to how character-based models represent word meaning (Kim et al., 2016; Adel et al., 2017), they can also store the meaning of frequent complex words that are segmented into subwords, i.e., frequent sub-

[1] We make our code and data available at https://github.com/valentinhofmann/superbizarre.

word collocations, in their model weights. When the complex-word meaning is neither stored as a whole-word vector nor in the model weights, PLMs compute the meaning as a compositional function of the subwords. Conceptually, PLMs can thus be interpreted as serial dual-route models. While the parallelism has not been observed before, it follows logically from the structure of PLMs. The key goal of this paper is to show that the implications of this observation are borne out empirically.

As a concrete example, consider the complex words `stabilize`, `realize`, `finalize`, `mobilize`, `tribalize`, and `templatize`, which are all formed by adding the verbal suffix `ize` to a nominal or adjectival stem. Taking BERT, specifically $\text{BERT}_{\text{BASE}}$ (uncased) (Devlin et al., 2019), as the example PLM, the words `stabilize` and `realize` have individual tokens in the input vocabulary and are hence associated with whole-word vectors storing their meanings, including highly lexicalized meanings as in the case of `realize`. By contrast, the words `finalize` and `mobilize` are segmented into `final`, `##ize` and `mob`, `##ili`, `##ze`, which entails that their meanings are not stored as whole-word vectors. However, both words have relatively high absolute frequencies of 2,540 (`finalize`) and 6,904 (`mobilize`) in the English Wikipedia, the main dataset used to pretrain BERT (Devlin et al., 2019), which means that BERT can store their meanings in its model weights during pretraining.[2] Notice this is even possible in the case of highly lexicalized meanings as for `mobilize`. Finally, the words `tribalize` and `templatize` are segmented into `tribal`, `##ize` and `te`, `##mp`, `##lat`, `##ize`, but as opposed to `finalize` and `mobilize` they do not occur in the English Wikipedia. As a result, BERT cannot store their meanings in its model weights during pretraining and needs to compute them from the meanings of the subwords.

Seeing PLMs as serial dual-route models allows for a more nuanced view on the central research question of this paper: in order to investigate semantic generalization we need to investigate the representations of those complex words that activate the computation-based route. The words that do so are the ones whose meaning is neither stored as a whole-word vector nor in the model weights and hence needs to be computed compositionally as a function of the subwords (`tribalize` and `templatize` in the discussed examples). We hypothesize that the morphological validity of the segmentation affects the representational quality in these cases, and that the best generalization is achieved by maximally meaningful tokens. It is crucial to note this does not imply that the tokens have to be morphemes, but the segmentation boundaries need to coincide with morphological boundaries, i.e., groups of morphemes (e.g., `tribal` in the segmentation of `tribalize`) are also possible.[3] For `tribalize` and `templatize`, we therefore expect the segmentation `tribal`, `##ize` (morphologically valid since all segmentation boundaries are morpheme boundaries) to result in a representation of higher quality than the segmentation `te`, `##mp`, `##lat`, `##ize` (morphologically invalid since the boundaries between `te`, `##mp`, and `##lat` are not morpheme boundaries). On the other hand, complex words whose meanings are stored in the model weights (`finalize` and `mobilize` in the discussed examples) are expected to be affected by the segmentation to a much lesser extent: if the meaning of a complex word is stored in the model weights, it should matter less whether the specific segmentation activating that meaning is morphologically valid (`final`, `##ize`) or not (`mob`, `##ili`, `##ze`).[4]

3 Experiments

3.1 Setup

Analyzing the impact of different segmentations on BERT's semantic generalization capabilities is not straightforward since it is not clear a priori how to measure the quality of representations. Here, we devise a novel lexical-semantic probing task: we use BERT's representations for complex words to predict semantic dimensions, specifically sentiment and topicality (see Figure 1). For sentiment, given the example complex word `superbizarre`, the task is to predict that its sentiment is negative. For topicality, given the example complex word `isotopize`, the task is to predict that it is used in physics. We confine ourselves to binary predic-

[2] Previous research suggests that such lexical knowledge is stored in the lower layers of BERT (Vulić et al., 2020).

[3] This is in line with substantial evidence from linguistics showing that frequent groups of morphemes can be treated as semantic wholes (Stump, 2017, 2019).

[4] We expect the distinction between storage and computation of complex-word meaning for PLMs to be a continuum. While the findings presented here are consistent with this view, we defer a more in-depth analysis to future work.

			Class 1		Class 2	
Dataset	Dimension	$\lvert\mathcal{D}\rvert$	Class	Examples	Class	Example
Amazon	Sentiment	239,727	neg	overpriced, crappy	pos	megafavorite, applausive
ArXiv	Topicality	97,410	phys	semithermal, ozoneless	cs	autoencoded, rankable
Reddit	Topicality	85,362	ent	supervampires, spoilerful	dis	antirussian, immigrationism

Table 1: Dataset characteristics. The table provides information about the datasets such as the relevant semantic dimensions with their classes and example complex words. $|\mathcal{D}|$: number of complex words; neg: negative; pos: positive; phys: physics; cs: computer science; ent: entertainment; dis: discussion.

tion, i.e., the probed semantic dimensions always consist of two classes (e.g., positive and negative). The extent to which a segmentation supports a solution of this task is taken as an indicator of its representational quality.

More formally, let $\mathcal{D}$ be a dataset consisting of complex words x and corresponding classes y that instantiate a certain semantic dimension (e.g., sentiment). We denote with $s(x) = (t_1, \ldots, t_k)$ the segmentation of x into a sequence of k subwords. We ask how s impacts the capability of BERT to predict y, i.e., how $p(y|(s(x))$, the likelihood of the true semantic class y given a certain segmentation of x, depends on different choices for s. The two segmentation methods we compare in this study are BERT's standard WordPiece segmentation (Schuster and Nakajima, 2012; Wu et al., 2016), s_w, and a derivational segmentation that segments complex words into stems and affixes, s_d.

3.2 Data

Since existing datasets do not allow us to conduct experiments following the described setup, we create new datasets in a weakly-supervised fashion that is conceptually similar to the method proposed by Mintz et al. (2009): we employ large datasets annotated for sentiment or topicality, extract derivationally complex words, and use the dataset labels to establish their semantic classes.

For determining and segmenting derivationally complex words, we use the algorithm introduced by Hofmann et al. (2020b), which takes as input a set of prefixes, suffixes, and stems and checks for each word in the data whether it can be derived from a stem using a combination of prefixes and suffixes.[5] The algorithm is sensitive to morpho-orthographic rules of English (Plag, 2003), e.g., when the suffix ize is removed from isotopize, the result is isotope, not isotop. We follow Hofmann et al. (2020a) in using the prefixes, suffixes, and stems in BERT's WordPiece vocabulary as input to the algorithm. This means that all tokens used by the derivational segmentation are in principle also available to the WordPiece segmentation, i.e., the difference between s_w and s_d does not lie in the vocabulary per se but rather in the way the vocabulary is used. See Appendix A.1 for details about the derivational segmentation.

To get the semantic classes, we compute for each complex word which fraction of texts containing the word belongs to one of two predefined sets of dataset labels (e.g., reviews with four and five stars for positive sentiment) and rank all words accordingly. We then take the first and third tertiles of complex words as representing the two classes. We randomly split the words into 60% training, 20% development, and 20% test.

In the following, we describe the characteristics of the three datasets in greater depth. Table 1 provides summary statistics. See Appendix A.2 for details about data preprocessing.

Amazon. Amazon is an online e-commerce platform. A large dataset of Amazon reviews has been made publicly available (Ni et al., 2019).[6] We extract derivationally complex words from reviews with one or two (neg) as well as four or five stars (pos), discarding three-star reviews for a clearer separation (Yang and Eisenstein, 2017).

ArXiv. ArXiv is an open-access distribution service for scientific articles. Recently, a dataset of all papers published on ArXiv with associated metadata has been released.[7] For this study, we extract all articles from physics (phys) and computer science (cs), which we identify using ArXiv's subject classification. We choose physics and computer

[5] The distinction between inflectionally and derivationally complex words is notoriously fuzzy (Haspelmath and Sims, 2010; ten Hacken, 2014). We try to exclude inflection as far as possible (e.g., by removing problematic affixes such as ing) but are aware that a clear separation does not exist.

[6] https://nijianmo.github.io/amazon/index.html

[7] https://www.kaggle.com/Cornell-University/arxiv

	Amazon		ArXiv		Reddit	
Model	Dev	Test	Dev	Test	Dev	Test
DelBERT	.635 ± .001	.639 ± .002	.731 ± .001	.723 ± .001	.696 ± .001	.701 ± .001
BERT	.619 ± .001	.624 ± .001	.704 ± .001	.700 ± .002	.664 ± .001	.664 ± .003
Stem	.572 ± .003	.573 ± .003	.705 ± .001	.697 ± .001	.679 ± .001	.684 ± .002
Affixes	.536 ± .008	.539 ± .008	.605 ± .001	.603 ± .002	.596 ± .001	.596 ± .001

Table 2: Results. The table shows the average performance as well as standard deviation (F1) of 20 models trained with different random seeds. Best result per column highlighted in gray, second-best in light gray.

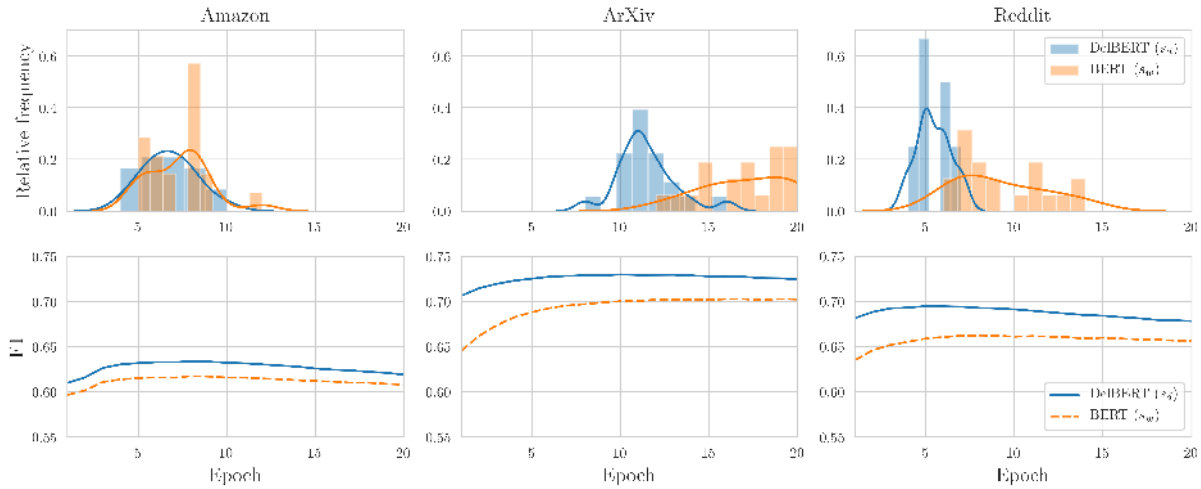


Figure 2: Convergence analysis. The upper panels show the distributions of the number of epochs after which the models reach their maximum validation performance. The lower panels show the trajectories of the average validation performance (F1) across epochs. The plots are based on 20 models trained with different random seeds. The convergence statistics for DelBERT and BERT are directly comparable because the optimal learning rate is the same (see Appendix A.3). DelBERT models reach their performance peak faster than BERT models.

science since we expect large topical distances for these classes (compared to alternatives such as mathematics and computer science).

Reddit. Reddit is a social media platform hosting discussions about various topics. It is divided into smaller communities, so-called subreddits, which have been shown to be a rich source of derivationally complex words (Hofmann et al., 2020c). Hofmann et al. (2020a) have published a dataset of derivatives found on Reddit annotated with the subreddits in which they occur.[8] Inspired by a content-based subreddit categorization scheme,[9] we define two groups of subreddits, an entertainment set (`ent`) consisting of the subreddits `anime`, `DestinyTheGame`, `funny`, `Games`, `gaming`, `leagueoflegends`, `movies`, `Music`, `pics`, and `videos`, as well as a discussion set (`dis`) consisting of the subreddits `askscience`, `atheism`, `conspiracy`, `news`, `Libertarian`, `politics`, `science`, `technology`, `TwoXChromosomes`, and `worldnews`, and extract all derivationally complex words occurring in them. We again expect large topical distances for these classes.

Given that the automatic creation of the datasets necessarily introduces noise, we measure human performance on 100 randomly sampled words per dataset, which ranges between 71% (Amazon) and 78% (ArXiv). These values can thus be seen as an upper bound on performance.

3.3 Models

We train two main models on each binary classification task: BERT with the standard WordPiece segmentation (s_w) and BERT using the derivational segmentation (s_d), a model that we refer to as DelBERT (**De**rivation **l**everaging **BERT**). BERT and DelBERT are identical except for the way in which they use the vocabulary of input tokens (but the vocabulary itself is also identical for both models).

[8] https://github.com/valentinhofmann/dagobert

[9] https://www.reddit.com/r/TheoryOfReddit/comments/1f7hqc/the_200_most_active_subreddits_categorized_by

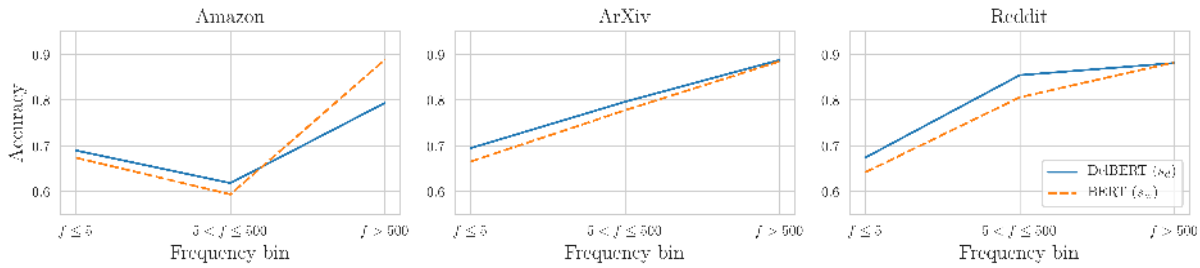


Figure 3: Frequency analysis. The plots show the average performance (accuracy) of 20 BERT and DelBERT models trained with different random seeds for complex words of low ($f \leq 5$), mid ($5 < f \leq 500$), and high ($f > 500$) frequency. On all three datasets, BERT performs similarly or better than DelBERT for complex words of high frequency but worse for complex words of low and mid frequency.

The specific BERT variant we use is $\text{BERT}_{\text{BASE}}$ (uncased) (Devlin et al., 2019). For the derivational segmentation, we follow previous work by Hofmann et al. (2020a) in separating stem and prefixes by a hyphen. We further follow Casanueva et al. (2020) and Vulić et al. (2020) in mean-pooling the output representations for all subwords, excluding BERT's special tokens. The mean-pooled representation is then fed into a two-layer feed-forward network for classification. To examine the relative importance of different types of morphological units, we train two additional models in which we ablate information about stems and affixes, i.e., we represent stems and affixes by the same randomly chosen input embedding.[10]

We finetune BERT, DelBERT, and the two ablated models on the three datasets using 20 different random seeds. We choose F1 as the evaluation measure. See Appendix A.3 for details about implementation and hyperparameters.

3.4 Results

DelBERT (s_d) outperforms BERT (s_w) by a large margin on all three datasets (Table 2). It is interesting to notice that the performance difference is larger for ArXiv and Reddit than for Amazon, indicating that the gains in representational quality are particularly large for topicality.

What is it that leads to DelBERT's increased performance? The ablation study shows that models using only stem information already achieve relatively high performance and are on par or even better than the BERT models on ArXiv and Reddit. However, the DelBERT models still perform substantially better than the stem models on all three datasets. The gap is particularly pronounced for Amazon, which indicates that the interaction between the meaning of stem and affixes is more complex for sentiment than for topicality. This makes sense from a linguistic point of view: while stems tend to be good cues for the topical associations of a complex word, sentiment often depends on semantic interactions between stems and affixes. For example, while the prefix `un` turns the sentiment of `amusing` negative, it turns the sentiment of `biased` positive. Such effects involving negation and antonymy are known to be challenging for PLMs (Ettinger, 2020; Kassner and Schütze, 2020) and might be one of the reasons for the generally lower performance on Amazon.[11] The performance of models using only affixes is much lower.

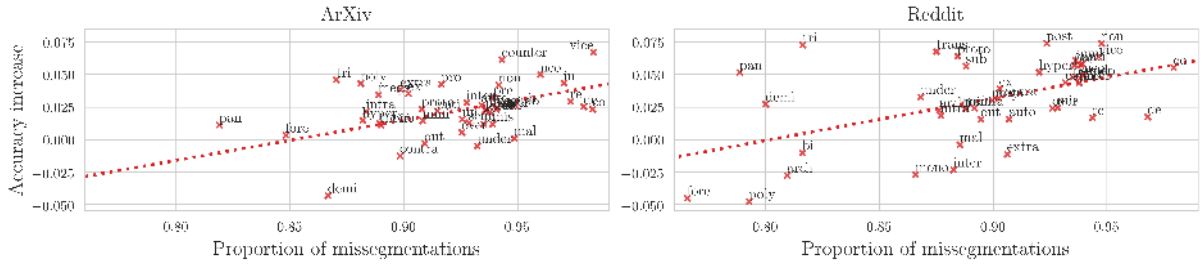
3.5 Quantitative Analysis

To further examine how BERT (s_w) and DelBERT (s_d) differ in the way they infer the meaning of complex words, we perform a convergence analysis. We find that the DelBERT models reach their peak in performance faster than the BERT models (Figure 2). This is in line with our interpretation of PLMs as serial dual-route models (see Section 2.2): while DelBERT operates on morphological units and can combine the subword meanings to infer the meanings of complex words, BERT's subwords do not necessarily carry lexical meanings, and hence the derivational patterns need to be stored by adapting the model weights. This is an additional burden, leading to longer convergence times and substantially worse overall performance.

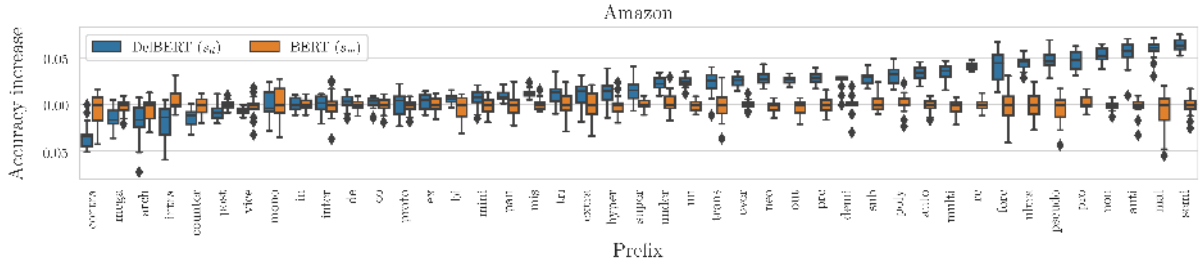
Our hypothesis that PLMs can use two routes

[10] For affix ablation, we use two different input embeddings for prefixes and suffixes.

[11] Another reason for the lower performance on sentiment is that the datasets were created automatically (see Section 3.2), and hence many complex words do not directly carry information about sentiment or topicality. The density of such words is higher for sentiment than topicality since the topic of discussion affects the likelihoods of most content words.



(a) Topicality prediction



(b) Sentiment prediction

Figure 4: Accuracy increase of DelBERT compared to BERT for prefixes. The plots show the accuracy increase as a function of the proportion of morphologically incorrect WordPiece segmentations (topicality prediction) and as ordered boxplot pairs centered on the median accuracy of BERT (sentiment prediction). Negative values mean that the DelBERT models have a lower accuracy than the BERT models for a certain prefix.

to process complex words (storage in weights and compositional computation based on input embeddings), and that the second route is blocked when the input segmentation is not morphological, suggests the existence of frequency effects: BERT might have seen frequent complex words multiple times during pretraining and stored their meaning in the model weights. This is less likely for infrequent complex words, making the capability to compositionally infer the meaning (i.e., the computation route) more important. We therefore expect the difference in performance between DelBERT (which should have an advantage on the computation route) and BERT to be larger for infrequent words. To test this hypothesis, we split the complex words of each dataset into three bins of low ($f \leq 5$), mid ($5 < f \leq 500$), and high ($f > 500$) absolute frequencies, and analyze how the performance of BERT and DelBERT differs on the three bins. For this and all subsequent analyses, we merge development and test sets and use accuracy instead of F1 since it makes comparisons across small sets of data points more interpretable. The results are in line with our hypothesis (Figure 3): BERT performs worse than DelBERT on complex words of low and mid frequencies but achieves very similar (ArXiv, Reddit) or even better (Amazon) accuracies on high-frequency complex words. These results strongly suggest that two different mechanisms are involved, and that BERT has a disadvantage for complex words that do not have a high frequency. At the same time, the slight advantage of BERT on high-frequency complex words indicates that it has high-quality representations of these words in its weights, which DelBERT cannot exploit since it uses a different segmentation.

We are further interested to see whether the affix type has an impact on the relative performance of BERT and DelBERT. To examine this question, we measure the accuracy increase of DelBERT as compared to BERT for individual affixes, averaged across datasets and random seeds. We find that the increase is almost twice as large for prefixes ($\mu = .023$, $\sigma = .017$) than for suffixes ($\mu = .013$, $\sigma = .016$), a difference that is shown to be significant by a two-tailed Welch's t-test ($d = .642$, $t(82.97) = 2.94$, $p < .01$).[12] Why is having access to the correct morphological segmentation more advantageous for prefixed than suffixed complex words? We argue that there are two key factors at play. First, the WordPiece tokenization sometimes generates the morphologically correct segmenta-

[12] We use a Welch's instead of Student's t-test since it does not assume that the distributions have equal variance.

Dataset	x	y	$s_d(x)$	μ_p	$s_w(x)$	μ_p
Amazon	applausive	pos	applause, ##ive	.847	app, ##laus, ##ive	.029
	superannoying	neg	super, -, annoying	.967	super, ##ann, ##oy, ##ing	.278
	overseasoned	neg	over, -, seasoned	.956	overseas, ##oned	.219
ArXiv	isotopize	phy	isotope, ##ize	.985	iso, ##top, ##ize	.039
	antimicrosoft	cs	anti, -, microsoft	.936	anti, ##mic, ##ros, ##oft	.013
	inkinetic	phy	in, -, kinetic	.983	ink, ##ine, ##tic	.035
Reddit	prematuration	dis	premature, ##ation	.848	prem, ##at, ##uration	.089
	nonmultiplayer	ent	non, -, multiplayer	.950	non, ##mu, ##lt, ##ip, ##layer	.216
	promosque	dis	pro, -, mosque	.961	promo, ##sque	.066

Table 3: Error analysis. The table gives example complex words that are consistently classified correctly by DelBERT and incorrectly by BERT. x: complex word; y: semantic class; $s_d(x)$: derivational segmentation; μ_p: average likelihood of true semantic class across 20 models trained with different random seeds; $s_w(x)$: WordPiece segmentation. For the complex words shown, μ_p is considerably higher with DelBERT than with BERT.

tion, but it does so with different frequencies for prefixes and suffixes. To detect morphologically incorrect segmentations, we check whether the WordPiece segmentation keeps the stem intact, which is in line with our definition of morphological validity (Section 2.2) and provides a conservative estimate of the error rate. For prefixes, the WordPiece tokenization is seldom correct (average error rate: $\mu = .903$, $\sigma = .042$), whereas for suffixes it is correct about half the time ($\mu = .503$, $\sigma = .213$). Hence, DelBERT gains a greater advantage for prefixed words. Second, prefixes and suffixes have different linguistic properties that affect the prediction task in unequal ways. Specifically, whereas suffixes have both syntactic and semantic functions, prefixes have an exclusively semantic function and always add lexical-semantic meaning to the stem (Giraudo and Grainger, 2003; Beyersmann et al., 2015). As a result, cases such as `unamusing` where the affix boundary is a decisive factor for the prediction task are more likely to occur with prefixes than suffixes, thus increasing the importance of a morphologically correct segmentation.[13]

Given the differences between sentiment and topicality prediction, we expect variations in the relative importance of the two identified factors: (i) in the case of sentiment the advantage of s_d should be maximal for affixes directly affecting sentiment; (ii) in the case of topicality its advantage should be the larger the higher the proportion of incorrect segmentations for a particular affix, and hence the more frequent the cases where DelBERT has access to the stem while BERT does not. To test this hypothesis, we focus on predictions for prefixed complex words. For each dataset, we measure for individual prefixes the accuracy increase of the DelBERT models as compared to the BERT models, averaged across random seeds, as well as the proportion of morphologically incorrect segmentations produced by WordPiece. We then calculate linear regressions to predict the accuracy increases based on the proportions of incorrect segmentations. This analysis shows a significant positive correlation for ArXiv ($R^2 = .304$, $F(1, 41) = 17.92$, $p < 0.001$) and Reddit ($R^2 = .270$, $F(1, 40) = 14.80$, $p < 0.001$) but not for Amazon ($R^2 = .019$, $F(1, 41) = .80$, $p = .375$), which is in line with our expectations (Figure 4a). Furthermore, ranking the prefixes by accuracy increase for Amazon confirms that the most pronounced differences are found for prefixes that can change the sentiment such as `non`, `anti`, `mal`, and `pseudo` (Figure 4b).

3.6 Qualitative Analysis

Besides quantitative factors, we are interested in identifying qualitative contexts in which DelBERT has a particular advantage compared to BERT. To do so, we filter the datasets for complex words that are consistently classified correctly by DelBERT and incorrectly by BERT. Specifically, we compute for each word the average likelihood of the true semantic class across DelBERT and BERT models, respectively, and rank words according to the likelihood difference between both model types. Examining the words with the most extreme differences, we observe three classes (Table 3).

First, the addition of a suffix is often connected with morpho-orthographic changes (e.g., the deletion of a stem-final `e`), which leads to a segmentation of the stem into several subwords

[13] Notice that there are suffixes with similar semantic effects (e.g., `less`), but they are less numerous.

since the truncated stem is not in the WordPiece vocabulary (`applausive`, `isotopize`, `prematuration`). The model does not seem to be able to recover the meaning of the stem from the subwords. Second, the addition of a prefix has the effect that the word-internal (as opposed to word-initial) form of the stem would have to be available for proper segmentation. Since this form rarely exists in the WordPiece vocabulary, the stem is segmented into several subwords (`superannoying`, `antimicrosoft`, `nonmultiplayer`). Again, it does not seem to be possible for the model to recover the meaning of the stem. Third, the segmentation of prefixed complex words often fuses the prefix with the first characters of the stem (`overseasoned`, `inkinetic`, `promosque`). This case is particularly detrimental since it not only makes it difficult to recover the meaning of the stem but also creates associations with unrelated meanings, sometimes even opposite meanings as in the case of `superbizarre`. The three classes thus underscore the difficulty of inferring the meaning of complex words from the subwords when the whole-word meaning is not stored in the model weights and the subwords are not morphological.

4 Related Work

Several recent studies have examined how the performance of PLMs is affected by their input segmentation. Tan et al. (2020) show that tokenizing inflected words into stems and inflection symbols allows BERT to generalize better on non-standard inflections. Bostrom and Durrett (2020) pretrain RoBERTa with different tokenization methods and find tokenizations that align more closely with morphology to perform better on a number of tasks. Ma et al. (2020) show that providing BERT with character-level information also leads to enhanced performance. Relatedly, studies from automatic speech recognition have demonstrated that morphological decomposition improves the perplexity of language models (Fang et al., 2015; Jain et al., 2020). Whereas these studies change the vocabulary of input tokens (e.g., by adding special tokens), we show that even when keeping the pretrained vocabulary fixed, employing it in a morphologically correct way leads to better performance.[14]

[14]There are also studies that analyze morphological aspects of PLMs without a focus on questions surrounding segmentation (Edmiston, 2020; Klemen et al., 2020).

Most NLP studies on derivational morphology have been devoted to the question of how semantic representations of derivationally complex words can be enhanced by including morphological information (Luong et al., 2013; Botha and Blunsom, 2014; Qiu et al., 2014; Bhatia et al., 2016; Cotterell and Schütze, 2018), and how affix embeddings can be computed (Lazaridou et al., 2013; Kisselew et al., 2015; Padó et al., 2016). Cotterell et al. (2017), Vylomova et al. (2017), and Deutsch et al. (2018) propose sequence-to-sequence models for the generation of derivationally complex words. Hofmann et al. (2020a) address the same task using BERT. In contrast, we analyze how different input segmentations affect the semantic representations of derivationally complex words in PLMs, a question that has not been addressed before.

5 Conclusion

We have examined how the input segmentation of PLMs, specifically BERT, affects their interpretations of derivationally complex words. Drawing upon insights from psycholinguistics, we have deduced a conceptual interpretation of PLMs as serial dual-route models, which implies that maximally meaningful input tokens should allow for the best generalization on new words. This hypothesis was confirmed by a series of semantic probing tasks on which DelBERT, a model using derivational segmentation, consistently outperformed BERT using WordPiece segmentation. Quantitative and qualitative analyses further showed that BERT's inferior performance was caused by its inability to infer the complex-word meaning as a function of the subwords when the complex-word meaning was not stored in the weights. Overall, our findings suggest that the generalization capabilities of PLMs could be further improved if a morphologically-informed vocabulary of input tokens were used.

Acknowledgements

This work was funded by the European Research Council (#740516) and the Engineering and Physical Sciences Research Council (EP/T023333/1). The first author was also supported by the German Academic Scholarship Foundation and the Arts and Humanities Research Council. We thank the reviewers for their helpful comments.

References

Heike Adel, Ehsaneddin Asgari, and Hinrich Schütze. 2017. Overview of character-based models for natural language processing. In *International Conference on Computational Linguistics and Intelligent Text Processing (CICLing) 18*.

Maria Alegre and Peter Gordon. 1999. Frequency effects and the representational status of regular inflections. *Journal of Memory and Language*, 40:41–61.

Ben Athiwaratkun, Andrew Wilson, and Anima Anandkumar. 2018. Probabilistic fasttext for multi-sense word embeddings. In *Annual Meeting of the Association for Computational Linguistics (ACL) 56*.

R. Harald Baayen, Ton Dijkstra, and Robert Schreuder. 1997. Singulars and plurals in Dutch: Evidence for a parallel dual-route model. *Journal of Memory and Language*, 37:94–117.

R. Harald Baayen and Rochelle Lieber. 1991. Productivity and English derivation: A corpus-based study. *Linguistics*, 29(5).

R. Harald Baayen, Robert Schreuder, and Richard Sproat. 2000. Morphology in the mental lexicon: A computational model for visual word recognition. In Frank van Eynde and Dafydd Gibbon, editors, *Lexicon development for speech and language processing*, pages 267–293. Springer, Dordrecht.

Raymond Bertram, Matti Laine, R. Harald Baayen, Robert Schreuder, and Jukka Hyönä. 2000a. Affixal homonymy triggers full-form storage, even with inflected words, even in a morphologically rich language. *Cognition*, 74:B13–B25.

Raymond Bertram, Robert Schreuder, and R. Harald Baayen. 2000b. The balance of storage and computation in morphological processing: The role of word formation type, affixal homonymy, and productivity. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 26(2):489–511.

Elisabeth Beyersmann, Johannes C. Ziegler, and Jonathan Grainger. 2015. Differences in the processing of prefixes and suffixes revealed by a letter-search task. *Scientific Studies of Reading*, 19(5):360–373.

Parminder Bhatia, Robert Guthrie, and Jacob Eisenstein. 2016. Morphological priors for probabilistic neural word embeddings. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2016*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining. In *Findings of Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Jan A. Botha and Phil Blunsom. 2014. Compositional morphology for word representations and language modelling. In *International Conference on Machine Learning (ICML) 31*.

Cristina Burani and Alfonso Caramazza. 1987. Representation and processing of derived words. *Language and Cognitive Processes*, 2(3-4):217–227.

Cristina Burani and Alessandro Laudanna. 1992. Units of representation for derived words in the lexicon. In Ram Frost and Leonard Katz, editors, *Orthography, phonology, morphology, and meaning*, pages 361–376. North-Holland, Amsterdam.

Brian Butterworth. 1983. Lexical representation. In Brian Butterworth, editor, *Language production: Development, writing and other language processes*, pages 257–294. Academic Press, London.

Joan Bybee. 1988. Morphology as lexical organization. In Michael Hammond and Michael Noonan, editors, *Theoretical approaches to morphology: Approaches in modern linguistics*, pages 119–141. Academic Press, San Diego, CA.

Joan Bybee. 1995. Regular morphology and the lexicon. *Language and Cognitive Processes*, 10(425-455).

Alfonso Caramazza, Alessandro Laudanna, and Cristina Romani. 1988. Lexical access and inflectional morphology. *Cognition*, 28(297-332).

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Efficient intent detection with dual sentence encoders. In *Workshop on Natural Language Processing for Conversational AI 2*.

Kenneth Church. 2020. Emerging trends: Subwords, seriously? *Natural Language Engineering*, 26(3):375–382.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pretraining text encoders as discriminators rather than generators. In *International Conference on Learning Representations (ICLR) 8*.

Ryan Cotterell and Hinrich Schütze. 2018. Joint semantic synthesis and morphological analysis of the derived word. *Transactions of the Association for Computational Linguistics*, 6:33–48.

Ryan Cotterell, Ekaterina Vylomova, Huda Khayrallah, Christo Kirov, and David Yarowsky. 2017. Paradigm completion for derivational morphology. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2017*.

Scott Deerwester, Susan T. Dumais, George Furnas, Thomas Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Daniel Deutsch, John Hewitt, and Dan Roth. 2018. A distributional and orthographic aggregation model for English derivational morphology. In *Annual Meeting of the Association for Computational Linguistics (ACL) 56*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HTL) 2019*.

Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A. Smith. 2019. Show your work: Improved reporting of experimental results. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2019*.

Daniel Edmiston. 2020. A systematic analysis of morphological content in BERT models for multiple languages. In *arXiv 2004.03032*.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2019*.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Hao Fang, Mari Ostendorf, Peter Baumann, and Janet B. Pierrehumbert. 2015. Exponential language modeling using morphological features and multitask learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(12):2410–2421.

Laurie B. Feldman and Carol A. Fowler. 1987. The inflected noun system in Serbo-Croatian: Lexical representation of morphological structure. *Memory and Cognition*, 15(1):1–12.

Uli H. Frauenfelder and Robert Schreuder. 1992. Constraining psycholinguistic models of morphological processing and representation: The role of productivity. In Geert Booij and Jaap van Marle, editors, *Yearbook of morphology 1991*, volume 26, pages 165–183. Kluwer, Dordrecht.

Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal*, 12(2):23–38.

Hélène Giraudo and Jonathan Grainger. 2003. On the role of derivational affixes in recognizing complex words: Evidence from masked priming. In R. Harald Baayen and Robert Schreuder, editors, *Morphological structure in language processing*, pages 209–232. De Gruyter, Berlin.

Pius ten Hacken. 2014. Delineating derivation and inflection. In Rochelle Lieber and Pavol Štekauer, editors, *The Oxford handbook of derivational morphology*, pages 10–25. Oxford University Press, Oxford.

Bo Han and Timothy Baldwin. 2011. Lexical normalisation of short text messages: Makn sens a #twitter. In *Annual Meeting of the Association for Computational Linguistics (ACL) 49*.

Martin Haspelmath and Andrea D. Sims. 2010. *Understanding morphology*. Routledge, New York, NY.

Benjamin Heinzerling and Michael Strube. 2019. Sequence tagging with contextual and non-contextual subword representations: A multilingual evaluation. In *Annual Meeting of the Association for Computational Linguistics (ACL) 57*.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HTL) 2019*.

Valentin Hofmann, Janet B. Pierrehumbert, and Hinrich Schütze. 2020a. DagoBERT: Generating derivational morphology with a pretrained language model. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Valentin Hofmann, Janet B. Pierrehumbert, and Hinrich Schütze. 2020b. Predicting the growth of morphological families from social and linguistic factors. In *Annual Meeting of the Association for Computational Linguistics (ACL) 58*.

Valentin Hofmann, Hinrich Schütze, and Janet B. Pierrehumbert. 2020c. A graph auto-encoder model of derivational morphology. In *Annual Meeting of the Association for Computational Linguistics (ACL) 58*.

Abhilash Jain, Aku Rouhe, Stig-Arne Grönroos, and Mikko Kurimo. 2020. Finnish ASR with deep transformer models. In *Conference of the International Speech Communication Association (INTERSPEECH) 21*.

Ganesh Jawahar, Benoit Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Annual Meeting of the Association for Computational Linguistics (ACL) 57*.

Emilia Kacprzak, Laura M. Koesten, Luis-Daniel Ibáñez, Elena Simperl, and Jeni Tennison. 2017. A query log analysis of dataset search. In *International Conference on Web Engineering (ICWE) 17*.

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Annual Meeting of the Association for Computational Linguistics (ACL) 58*.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M. Rush. 2016. Character-aware neural language models. In *Conference on Artificial Intelligence (AAAI) 30*.

Diederik P. Kingma and Jimmy L. Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR) 3*.

Max Kisselew, Sebastian Padó, Alexis Palmer, and Jan Šnajder. 2015. Obtaining a better understanding of distributional models of german derivational morphology. In *International Conference on Computational Semantics (IWCS) 11*.

Matej Klemen, Luka Krsnik, and Marko Robnik-Šikonja. 2020. Enhancing deep neural networks with morphological information. In *arXiv 2011.12432*.

Victor Kuperman, Raymond Bertram, and R. Harald Baayen. 2008. Morphological dynamics in compound processing. *Language and Cognitive Processes*, 23(7-8):1089–1132.

Victor Kuperman, Robert Schreuder, Raymond Bertram, and R. Harald Baayen. 2009. Reading of polymorphemic Dutch compounds: Towards a multiple route model of lexical processing. *Journal of Experimental Psychology: Human Perception and Performance*, 35(3):876–895.

Alessandro Laudanna and Cristina Burani. 1985. Address mechanisms to decomposed lexical entries. *Linguistics*, 23(5).

Alessandro Laudanna and Cristina Burani. 1995. Distributional properties of derivational affixes: Implications for processing. In Laurie B. Feldman, editor, *Morphological aspects of language processing*, pages 345–364. Lawrence Erlbaum, Hillsdale, NJ.

Angeliki Lazaridou, Marco Marelli, Roberto Zamparelli, and Marco Baroni. 2013. Compositional-ly derived representations of morphologically complex words in distributional semantics. In *Annual Meeting of the Association for Computational Linguistics (ACL) 51*.

Alina Leminen, Eva Smolka, Jon Duñabeitia, and Christos Pliatsikas. 2019. Morphological processing in the brain: The good (inflection), the bad (derivation) and the ugly (compounding). *Cortex*, 116:4–44.

Minh-Thang Luong, Richard Socher, and Christopher D. Manning. 2013. Better word representations with recursive neural networks for morphology. In *Conference on Computational Natural Language Learning (CoNLL) 17*.

Wentao Ma, Yiming Cui, Chenglei Si, Ting Liu, Shijin Wang, and Guoping Hu. 2020. CharBERT: Character-aware pre-trained language model. In *International Conference on Computational Linguistics (COLING) 28*.

Leon Manelis and David A. Tharp. 1977. The processing of affixed words. *Memory and Cognition*, 5(6):690–695.

Louis Martin, Benjamin Muller, Pedro J. Suárez, Yoann Dupont, Laurent Romary, de la Clergerie, Éric V., Djamé Seddah, and Benoit Sagot. 2020. CamemBERT: A tasty French language model. In *Annual Meeting of the Association for Computational Linguistics (ACL) 58*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *arXiv 1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS) 26*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Annual Meeting of the Association for Computational Linguistics (ACL) 47*.

Jeremy M. Needle and Janet B. Pierrehumbert. 2018. Gendered associations of english morphology. *Journal of the Association for Laboratory Phonology*, 9(1):119.

Boris New, Marc Brysbaert, Juan Segui, Ludovic Ferrand, and Kathleen Rastle. 2004. The processing of singular and plural nouns in french and english. *Journal of Memory and Language*, 51(4):568–585.

Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fined-grained aspects. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2019*.

Sebastian Padó, Aurélie Herbelot, Max Kisselew, and Jan Šnajder. 2016. Predictability of distributional semantics in derivational word formation. In *International Conference on Computational Linguistics (COLING) 26*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2014*.

Yuval Pinter, Cassandra L. Jacobs, and Jacob Eisenstein. 2020. Will it unblend? In *Findings of Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Ingo Plag. 2003. *Word-formation in English*. Cambridge University Press, Cambridge, UK.

Siyu Qiu, Qing Cui, Jiang Bian, Bin Gao, and Tie-Yan Liu. 2014. Co-learning of word representations and morpheme representations. In *International Conference on Computational Linguistics (COLING) 25*.

Péter Rácz, Clay Beckner, Jennifer Hay, and Janet B. Pierrehumbert. 2020. Morphological convergence as on-line lexical analogy. *Language*, 96(4):735–770.

Péter Rácz, Janet B. Pierrehumbert, Jennifer Hay, and Viktória Papp. 2015. Morphological emergence. In Brian MacWhinney and William O'Grady, editors, *The handbook of language emergence*, pages 123–146. Wiley, Hoboken, NJ.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

Kathleen Rastle and Matthew H. Davis. 2008. Morphological decomposition based on the analysis of orthography. *Language and Cognitive Processes*, 23(7-8):942–971.

Kathleen Rastle, Matthew H. Davis, and Boris New. 2004. The broth in my brother's brothel: Morpho-orthographic segmentation in visual word recognition. *Psychonomic Bulletin and Review*, 11(6):1090–1098.

Alexandre Salle and Aline Villavicencio. 2018. Incorporating subword information into matrix factorization word embeddings. In *Workshop on Subword/Character LEvel Models 2*.

Robert Schreuder and R. Harald Baayen. 1995. Modeling morphological processing. In Laurie B. Feldman, editor, *Morphological aspects of language processing*, pages 131–154. Lawrence Erlbaum, Hillsdale, NJ.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and Korean voice search. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 37*.

Hinrich Schütze. 1992. Word space. In *Advances in Neural Information Processing Systems (NIPS) 5*, pages 895–902.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Annual Meeting of the Association for Computational Linguistics (ACL) 54*.

Suzanna Sia, Ayush Dalmia, and Sabrina J. Mielke. 2020. Tired of topic models? Clusters of pretrained word embeddings make for fast and good topics too! In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Joseph P. Stemberger. 1994. Rule-less morphology at the phonology-lexicon interface. In Susan D. Lima, Roberta Corrigan, and Gregory Iverson, editors, *The reality of linguistic rules*, pages 147–169. John Benjamins, Amsterdam.

Gregory Stump. 2017. Rule conflation in an inferential-realizational theory of morphotactics. *Acta Linguistica Academica*, 64(1):79–124.

Gregory Stump. 2019. Some sources of apparent gaps in derivational paradigms. *Morphology*, 29(2):271–292.

Marcus Taft. 1979. Recognition of affixed words and the word frequency effect. *Memory and Cognition*, 7(4):263–272.

Marcus Taft. 1981. Prefix stripping revisited. *Journal of Verbal Learning and Verbal Behavior*, 20:289–297.

Marcus Taft. 1988. A morphological-decomposition model of lexical representation. *Linguistics*, 26:657–667.

Marcus Taft. 1991. *Reading and the mental lexicon*. Lawrence Erlbaum, Hove, UK.

Marcus Taft. 1994. Interactive-activation as a framework for understanding morphological processing. *Language and Cognitive Processes*, 9(3):271–294.

Marcus Taft. 2004. Morphological decomposition and the reverse base frequency effect. *The Quarterly Journal of Experimental Psychology*, 57(4):745–765.

Marcus Taft and Kenneth I. Forster. 1975. Lexical storage and retrieval of prefixed words. *Journal of Verbal Learning and Verbal Behavior*, 14:638–647.

Samson Tan, Shafiq Joty, Lav R. Varshney, and Min-Yen Kan. 2020. Mind your inflections! Improving NLP for non-standard Englishes with base-inflection encoding. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Ivan Vulić, Edoardo M. Ponti, Robert Litschko, Goran Glavaš, and Anna Korhonen. 2020. Probing pretrained language models for lexical semantics. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Ekaterina Vylomova, Ryan Cotterell, Timothy Baldwin, and Trevor Cohn. 2017. Context-aware prediction of derivational word-forms. In *Conference of the European Chapter of the Association for Computational Linguistics (EACL) 15*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc Le V, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith

Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. In *arXiv 1609.08144*.

Yi Yang and Jacob Eisenstein. 2017. Overcoming language variation in sentiment analysis with social attention. *Transactions of the Association for Computational Linguistics*, 5:295–307.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems (NeurIPS) 33*.

A Appendices

A.1 Derivational Segmentation

Let A be a set of derivational affixes and S a set of stems. To determine the derivational segmentation of a word w, we employ an iterative algorithm. Define the set B_1^A of w as the words that remain when one derivational affix from A is removed from w. For example, `unlockable` can be segmented into `un`, `lockable` and `unlock`, `able` so $B_1^A(\texttt{unlockable}) = \{\texttt{lockable}, \texttt{unlock}\}$ (we assume that `un` and `able` are in A). We then iteratively create $B_{i+1}^A(w) = \bigcup_{b \in B_i^A(w)} B_1^A(b)$, i.e., we iteratively remove affixes from w. We stop as soon as $B_{i+1}^A(w) \cap S \neq \emptyset$. The element in this intersection, together with the used affixes from A, forms the derivational segmentation of w.[15] If there is no i such that $B_{i+1}^A(w) \cap S \neq \emptyset$, w does not have a derivational segmentation. The algorithm is sensitive to most morpho-orthographic rules of English (Plag, 2003), e.g., when the suffix `ize` is removed from `isotopize`, the resulting word is `isotope`, not `isotop`.

In this paper, we follow Hofmann et al. (2020a) in using BERT's prefixes, suffixes, and stems as input to the algorithm. Specifically, we assign 46 productive prefixes and 44 productive suffixes in BERT's vocabulary to A and all fully alphabetic words with more than 3 characters in BERT's vocabulary (excluding stopwords and affixes) to S, resulting in a total of 20,259 stems. This means that we only consider derivational segmentations that are possible given BERT's vocabulary.

[15] If $|B_{i+1}^A(w) \cap S| > 1$ (rarely the case in practice), the element with the lowest number of suffixes is chosen.

A.2 Data Preprocessing

We exclude texts written in a language other than English and remove strings containing numbers as well as hyperlinks. We follow Han and Baldwin (2011) in reducing repetitions of more than three letters (`niiiiice`) to three letters.

A.3 Hyperparameters

The feed-forward network has a ReLU activation after the first layer and a sigmoid activation after the second layer. The first layer has 100 dimensions. We apply dropout of 0.2 after the first layer. All other hyperparameters are as for $\text{BERT}_{\text{BASE}}$ (uncased) (Devlin et al., 2019). The number of trainable parameters is 109,559,241.

We use a batch size of 64 and perform grid search for the number of epochs $n \in \{1, \ldots, 20\}$ and the learning rate $l \in \{1 \times 10^{-6}, 3 \times 10^{-6}, 1 \times 10^{-5}, 3 \times 10^{-5}\}$ (selection criterion: F1 score). We tune l on Reddit (80 hyperparameter search trials per model type) and use the best configuration (which is identical for all model types) for 20 training runs with different random seeds on all three datasets (20 hyperparameter search trials per model type, dataset, and random seed). Models are trained with binary cross-entropy as the loss function and Adam (Kingma and Ba, 2015) as the optimizer. Experiments are performed on a GeForce GTX 1080 Ti GPU (11GB).

Table 4 lists statistics of the validation performance over hyperparameter search trials and provides information about best hyperparameter configurations as well as runtimes.[16] See also Section 3.5 and particularly Figure 2 in the main text, where we present a detailed analysis of the convergence behavior of the two main model types examined in this study (DelBERT and BERT).

[16] Since expected validation performance (Dodge et al., 2019) may not be correct for grid search, we report mean and standard deviation of the performance instead.

	Amazon					ArXiv					Reddit				
Model	μ	σ	n	l	τ	μ	σ	n	l	τ	μ	σ	n	l	τ
DelBERT	.627	.007	6.75	3e-06	67.73	.725	.006	11.45	3e-06	28.69	.687	.006	5.45	3e-06	25.56
BERT	.612	.006	7.30	3e-06	66.18	.693	.015	17.05	3e-06	28.04	.657	.007	9.25	3e-06	25.06
Stem	.556	.016	9.85	3e-06	67.43	.699	.005	8.15	3e-06	28.56	.670	.006	6.00	3e-06	25.39
Affixes	.519	.008	5.55	3e-06	67.70	.599	.004	7.50	3e-06	28.43	.593	.003	9.35	3e-06	25.49

Table 4: Validation performance statistics and hyperparameter search details. The table shows the mean (μ) and standard deviation (σ) of the validation performance (F1) on all hyperparameter search trials, the number of epochs (n) and learning rate (l) with the best validation performance, and the runtime (τ) in minutes for one full hyperparameter search (20 trials). The numbers are averaged across 20 training runs with different random seeds.