Reconsidering Token Embeddings with the Definitions for Pre-trained Language Models

Anonymous ACL submission

Abstract

Learning token embeddings based on token co-occurrence statistics has proven effective for both pre-training and fine-tuning in natural language processing. However, recent studies have pointed out the distribution of learned embeddings degenerates into anisotropy, and even pre-trained language models (PLMs) suffer from a loss of semantics-related information in embeddings for low-frequency tokens. This study first analyzes fine-tuning dynamics of a PLM, BART-large, and demonstrates its robustness against degeneration. On the basis of this finding, we propose DefinitionEMB, a method that utilizes definitions to construct isotropically distributed and semantics-related token embeddings for PLMs while maintaining original robustness during fine-tuning. Our experiments demonstrate the effectiveness of leveraging definitions from Wiktionary to construct such embeddings for RoBERTa-base and BART-large. Furthermore, the constructed embeddings for low-frequency tokens improve the performance of these models across various GLUE and four text summarization datasets.[1]

1 Introduction

Learning word embeddings, also known as word representations, is a fundamental challenge in natural language processing (NLP), given that embedding comprehensive word information serves as the initial step in many NLP tasks (Turian et al., 2010). Since the introduction of the Skip-gram model by Mikolov et al. (2013), the predominant approach for learning precise syntactic and semantic word embeddings in neural models is to train the models to predict words within given contexts based on word co-occurrence statistics.

Recent studies highlight representation degeneration issues that arise from learning word embeddings: the distribution of learned embeddings suffers from frequency bias (Yu et al., 2022) and exhibits a narrow cone-shaped anisotropy (Gao et al., 2019). Specifically, Tissier et al. (2017) claimed that, when using the co-occurrence principle, words that frequently appear together within contexts tend to have close representations. It aligns well with the anisotropy phenomenon observed in word embeddings of Word2Vec and GloVe (Mu and Viswanath, 2018). That is, embeddings tend to share a common direction and occupy a low-dimensional subspace, instead of being angularly uniformly distributed (i.e., isotropic). Biś et al. (2021) claimed that this phenomenon causes the shape of the word embedding matrix to degenerate into a narrow cone in a low-dimensional embedding space, when weight tying is applied during training; thereby, reducing the network's ability for effective generalization, especially for low-frequency (rare) words. Furthermore, Gong et al. (2018) observed that high-frequency (popular) and rare words occupy different areas in the embedding space and that embeddings for rare words contain more frequency-related information than semantics-related information, which raises concerns when replacing popular words with their rare counterparts (or vice versa).

To address the representation degeneration issues, previous studies have proposed solutions to post-process embeddings or enhance model optimization, for example, by eliminating specific directions from the embeddings (Mu and Viswanath, 2018), merging popular and rare words during training (Gong et al., 2018), and gating the gradients of embeddings (Yu et al., 2022). While these methods have improved the distribution of embeddings in task-specific models and the distribution of pre-trained word embeddings (e.g., Word2Vec), they cannot ensure learning of semantics-related embeddings for rare or out-of-vocabulary (OOV) words, due to underfitting caused by limited corresponding training data. Discounting the gradients or eliminating directions from embeddings may even lead

[1] Our code will be available at GitHub.

to a loss of semantics-related information and aggravate the underfitting problem. Considering that a vast majority of words are rare, as indicated by the Zipfian distribution (Zipf, 1949), it is worth exploring effective strategies to construct semantics-related embeddings for rare or OOV words while ensuring the isotropic distribution across the entire embedding matrix. Additionally, pre-trained language models (PLMs) with vocabularies consisting of subwords as tokens have become dominant in many NLP tasks due to their plug-and-play convenience and outstanding efficacy. However, experiments by Schick and Schütze (2020) and Biś et al. (2021) revealed that even BERT (Devlin et al., 2019) failed to understand many rare words and suffered from frequency bias in token embeddings.

Given the current situation, a natural question arises: **how can embeddings be constructed for PLMs to better understand rare tokens?** In educational psychology, achieving a deep understanding of a new word involves linking it to relevant concepts that students already possess, a process known as meaningful learning (Ausubel, 1968). Specifically, in classroom settings, these concepts can be conveyed through easily understood content to describe an aspect of the semantics of the new word, such as using dictionary definitions to elucidate the meaning of foreign language words. Throughout this paper, we refer to such explanatory content as "definitions." Inspired by the meaningful learning, we propose an architecture-agnostic method, DefinitionEMB, to construct token embeddings through denoising corresponding definitions. Our main contributions are summarized as follows:

- To the best of our knowledge, DefinitionEMB is the first method for constructing isotropically distributed and semantics-related token embeddings to improve PLMs' fine-tuned performance.
- Our experiments demonstrated that constructed embeddings for rare tokens, as a plug-and-play component, improve the fine-tuned performance for RoBERTa-base (Liu et al., 2019b) and BART-large (Lewis et al., 2020) across various GLUE and four text summarization datasets.
- We observed that using Mu and Viswanath (2018) for BART results in degeneration during fine-tuning, while BART with and without DefinitionEMB are robust against the degeneration. Additionally, BART with DefinitionEMB mixes embeddings with different frequencies and related semantics more thoroughly than the original BART.

2 Related Work

Previous studies have attempted to improve the distribution of word/token embeddings. Both Gong et al. (2018) and Yu et al. (2022) observed that popular and rare word/token embeddings tend to occupy different subregions within the embedding space, even when they are semantically similar. Gong et al. (2018) further found that embeddings of popular words in Word2Vec usually have semantically related neighbors, while rare words do not, with many of the nearest neighbors also being rare words. To merge popular and rare word embeddings geometrically, Gong et al. (2018) utilized a discriminator of generative adversarial networks to classify embeddings as belonging to either popular or rare words. These word embeddings were concurrently trained with a task-dependent loss to fool the discriminator. To achieve an isotropically distributed embedding matrix, Gao et al. (2019) proposed minimizing the cosine similarity between any two word embeddings during training and Zhang et al. (2020) followed to minimize the squared Euclidean distance between word embeddings with large context similarity. Mu and Viswanath (2018) eliminated the common mean vector and several dominating directions from embeddings, and Rajaee and Pilehvar (2021) followed Mu and Viswanath (2018) to eliminate principal components within clustered embeddings. Biś et al. (2021) simply removed the mean vector to improve the performance of GPT-2, BERT, and RoBERTa on the word similarity task. Subsequent empirical findings by Yu et al. (2022) indicated that the gradients of rare token embeddings push these token embeddings away from popular token embeddings, causing the word embedding matrix to degenerate into a narrow cone when weight tying is applied. To address this issue, Yu et al. (2022) proposed gating the gradients for rare token embeddings.

In contrast to the previous studies, our research considers both semantic and distributional information for embeddings. It is also the first to apply such considerations to improve the fine-tuned performance of PLMs, whose vocabulary consists of tokens created using byte-pair-encoding (Sennrich et al., 2016), WordPiece (Wu et al., 2016), or similar subword tokenization algorithms.

The OOV word issue has long been discussed in NLP. Previous studies have explored various solutions, such as leveraging surface-form information (Luong et al., 2013; Pinter et al., 2017;

Sasaki et al., 2019; Schick and Schütze, 2020) or the contexts in which these words occur (Khodak et al., 2018; Liu et al., 2019a; Schick and Schütze, 2019). Lexical definitions have also been considered. For instance, Tissier et al. (2017) augmented co-occurrence information with terms extracted from definitions to bring semantically related words closer. Bahdanau et al. (2018) trained a network to predict word embeddings based on definitions. Ruzzetti et al. (2022) extracted crucial words from each definition and fed them into a neural network to produce embeddings for the corresponding target words. Existing studies that utilize definitions to mimic embeddings for full words with a one-to-one mapping inspire us to explore a one-to-many mapping for the use of definitions for mimicking token embeddings. Thus, our constructed new embeddings can be easily applied to transformer-based (Vaswani et al., 2017) PLMs.

3 Preliminaries

Let $\mathcal{V} = \{v_n\}_{n=1}^{N}$ denote the predefined restricted vocabulary of a PLM, where N tokens are ranked in descending order according to their frequencies in the dataset for pre-training. $\mathbf{E} \in \mathbb{R}^{N \times H_e}$ denotes the pre-trained token embedding matrix of the PLM for $\mathcal{V}$, where H_e is the embedding size. Given a word w, we assume that w is tokenized into I tokens $(v^{w_1}, \ldots, v^{w_I})$, and $e(v) \in \mathbb{R}^{H_e}$ denotes the embedding of token v in the PLM.

The geometry of $\mathbf{E}$ is assumed to capture linguistic regularities: the similarity between token embeddings reflects the semantic similarity of the corresponding tokens. Therefore, researchers expect $\mathbf{E}$ to exhibit a uniform distribution, denoted as isotropic, to maximize the containment of linguistic information for distinguishing tokens. Improving isotropy has been proven effective in enhancing performance for text classification and word similarity tasks (Mu and Viswanath, 2018; Biś et al., 2021). To estimate isotropy, Mu and Viswanath (2018) proposed the metric $I(\mathbf{E}) \in [0, 1]$, where a value closer to 1 indicates higher isotropy. Appendix A describes the detail. Mu and Viswanath also proposed a post-processing technique to improve isotropy for $\mathbf{E}$ by eliminating the common mean vector and top-β dominating directions from $\mathbf{E}$. We denote this method as **DelDirection** and consider it as a baseline, with β set to 10.[2]

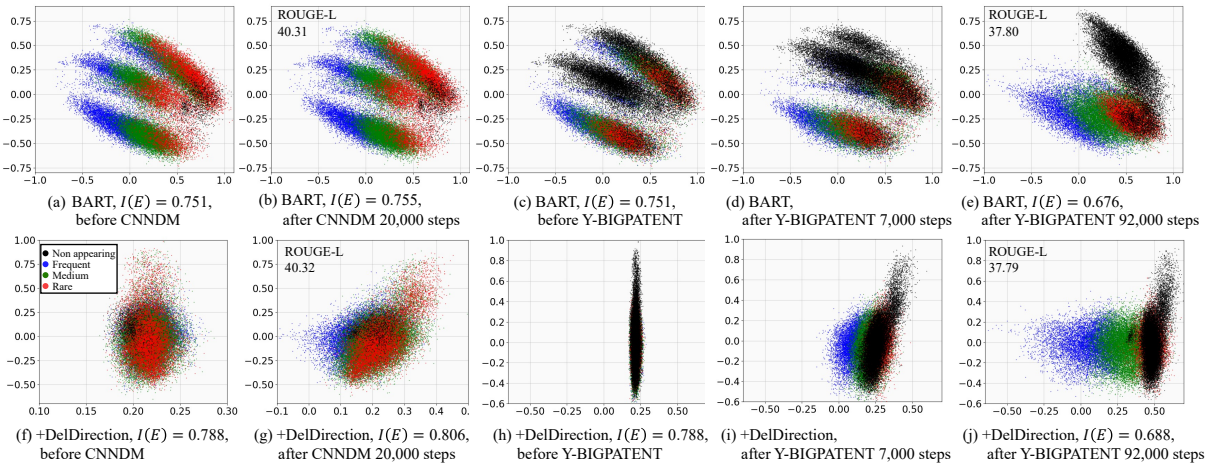
[2]According to Mu and Viswanath (2018), $\beta \approx H_e/100$.

4 Token Embedding Dynamics: An Experimental Investigation

Inspired by Yu et al. (2022)'s observation that the presence of a degenerated narrow cone is influenced by token frequency, we investigate whether the token frequency influences the embedding dynamics of PLMs during fine-tuning. To explore this phenomenon, we first examined the vocabulary distribution in downstream training datasets used for fine-tuning. We classified tokens in $\mathcal{V}$ into appearing and non appearing groups based on their appearance in the corresponding fine-tuning dataset. We observed that the CNNDM dataset has the most uniform distribution and the greatest variety of appearing tokens among CNNDM, Y-BIGPATENT,[3] XSum, and Billsum text summarization datasets. The token frequency in the Y-BIGPATENT dataset is significantly higher than in the other datasets, despite having the smallest variety of appearing tokens. Appendix B shows the detailed distributions.

We continue to uncover the embedding dynamics of PLMs, focusing on BART as an empirical study. We concluded that **BART does not degenerate into a narrow cone; instead, it exhibits drift patterns influenced by token frequency. Additionally, merely improving isotropy for BART does not guarantee semantically distributed embeddings and improve performance for downstream tasks.** We describe the details below. We visualized the distribution of $\mathbf{E}$ and measured the isotropy of $\mathbf{E}$ in the BART-large model before and after fine-tuning on the CNNDM and Y-BIGPATENT datasets. For detailed results, 30%, 50%, and 20% of the appearing tokens in $\mathcal{V}$ are assigned to the frequent, medium, and rare groups, respectively, based on their frequency in the fine-tuning dataset. We made several observations by visualizing $\mathbf{E}$ using singular value decomposition (SVD) projection in Figure 1: (1) Using DelDirection before fine-tuning yields higher $I(\mathbf{E})$ and more thoroughly mixed embeddings with different token frequencies for BART, as shown in Figures 1 (a), (f), (c), and (h). (2) However, these benefits do not persist during fine-tuning. Instead, using DelDirection yields a fragile distribution, where more updating steps result in greater degeneration (Figures 1 (f) to (g) and (h) to (j)) and no further improved ROUGE scores on the downstream datasets (Figures 1 (g) and (j)). Specifically, the popular tokens drift away from the original overlapping position, as seen in

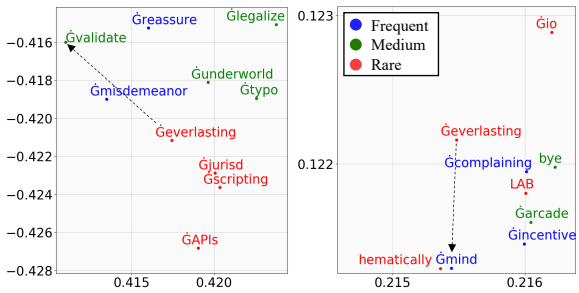
[3]The "y" category of BIGPATENT.



(a) BART, $I(E) = 0.751$, before CNNDM (b) BART, $I(E) = 0.755$, after CNNDM 20,000 steps (c) BART, $I(E) = 0.751$, before Y-BIGPATENT (d) BART, after Y-BIGPATENT 7,000 steps (e) BART, $I(E) = 0.676$, after Y-BIGPATENT 92,000 steps

(f) +DelDirection, $I(E) = 0.788$, before CNNDM (g) +DelDirection, $I(E) = 0.806$, after CNNDM 20,000 steps (h) +DelDirection, $I(E) = 0.788$, before Y-BIGPATENT (i) +DelDirection, after Y-BIGPATENT 7,000 steps (j) +DelDirection, $I(E) = 0.688$, after Y-BIGPATENT 92,000 steps

Figure 1: Projected token embeddings of BART with and without DelDirection on the CNNDM and Y-BIGPATENT datasets. Appendix C provides additional examples for BART and RoBERTa.



(a) BART (b) + DelDirection

Figure 2: Case study of the token embeddings before fine-tuning on the CNNDM dataset. "Ġ" denotes whitespace. The dashed lines from "Ġeverlasting" point to its semantics-related tokens, recognized by both ChatGPT 3.5 (Achiam et al., 2023) and Claude 3 Haiku (Anthropic, 2024). Appendix D lists their recognitions.

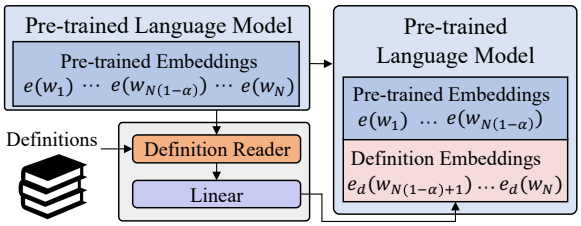


Figure 3: Overview of constructing definition embeddings to replace $\alpha\%$ of pre-trained embeddings.

Figures 1 (g) and (i). After more updates, tokens degenerate into a narrow cone with frequency bias, as seen in Figure 1 (j). (3) As for BART, it appears to be highly robust to the degeneration with the increasing number of updates, as seen in Figures 1 (b), (d), and (e). From Figure 2, we also observed that (4) the majority of neighbors of the rare token "Ġeverlasting" in both BART with and without DelDirection are not semantically related to "Ġeverlasting".

5 Methodology

This research assumes that there exists a function f, referred to as **DefinitionEMB**, which consists of a definition reader g and a linear mapping o for constructing token embeddings for a PLM from the corresponding word definition. Building on previous empirical studies, we expect new token embeddings for the PLM could preserve the robustness of pre-trained embeddings against degeneration while achieving a semantic and isotropic distribution. Hereafter, we denote the constructed embeddings as definition embeddings $e_d(v) \in \mathbb{R}^{H_e}$. Figure 3 provides an overview.

5.1 Embedding Construction

To establish a mapping from the definition of the word w to the target tokens $(v^{w_1}, \ldots, v^{w_I})$, we base our approach on prompt learning (Petroni et al., 2019) and denoising autoencoders (Vincent et al., 2010) to design prompts for f to trigger the corresponding target tokens. This involves corrupting the tokens in the prompt and training f to construct the embeddings of these corrupted tokens. The prompt incorporates the definition, part-of-speech, capitalization, and case sensitivity of the word w, along with the tokenizer's specific settings. Figure 4 shows examples of a prompt and its corrupted version. Let $p(w) = (v^{p_1}, \ldots, v^{p_J})$ be the tokenized corrupted prompt for the word w. In this scenario, given $p(w)$ with the j-th token $v^{p_j} \in w$ being corrupted, the definition reader

g utilizes a BERT-style denoising objective (Devlin et al., 2019) to generate the last hidden state $\mathbf{s}_j \in \mathbb{R}^{H_s}$ at the j-th time step, where H_s is the hidden size. Then, we linearly map $\mathbf{s}_j$ to $e_d(v^{p_j})$ as $e_d(v^{p_j}) = o(\mathbf{s}_j) = \mathbf{A}\mathbf{s}_j$, where $\mathbf{A} \in \mathbb{R}^{H_e \times H_s}$ is a weight matrix. $\mathbf{s}_j$ is computed as follows.

To ensure that the definition reader g performs effectively for PLMs, we initialize g using the given PLM. Because definitions contain popular, easily understandable words to explain their corresponding target words, and g is initialized from the PLM, we assume that the pre-trained embeddings of most tokens in $p(w)$ are isotropically distributed and semantics-related. Consequently, g can capture the directional aspects of these embeddings to create token embeddings for unknown tokens or tokens with anisotropically distributed or semantically unrelated pre-trained embeddings. Considering the most widely known PLM architectures, we propose two possible models for g: an encoder-only model and an encoder-decoder model. For the encoder-only reader, the training process involves randomly corrupting tokens within the corruption spans as follows: 50% of the tokens are replaced with a special mask token <MASK>, 25% are replaced with a random token, and the remaining 25% are left unchanged.[4] During inference, only one token within the corruption spans is replaced with <MASK> at a time, and this procedure is repeated until all tokens in the corruption spans have been replaced. With the corrupted prompt as an input, $\mathbf{s}_j$ is computed using the masked language modeling objective (Devlin et al., 2019). As for the encoder-decoder reader, we follow the approach by Raffel et al. (2020) to replace each corruption span with a mask token to construct the source sequence. Subsequently, we construct the target sequence using the replaced tokens delimited by the mask tokens used for replacement. $\mathbf{s}_j$ is computed using the causal language modeling objective (Hyndman and Athanasopoulos, 2018) by giving the source and previous tokens of v^{p_j} in the target.

5.2 Objective Function

Let $\mathcal{D} = \{(w_m, p(w_m))\}_{m=1}^{M}$ denote a corpus with M word-definition pairs. Given that f is optimized to find the overall optimal embedding space for $\mathcal{V}$, and pre-trained embeddings $e(v)$ for rare tokens in $\mathcal{V}$ may also contain corresponding representations, we incorporate definitions involving all tokens in $\mathcal{V}$ for training. Thus, the model parameters for f are optimized by minimizing the mean squared error (MSE) between pre-trained embeddings and definition embeddings as follows:

$$\mathcal{L} = \sum_{(w,p(w))\in\mathcal{D}} \frac{\sum_{i=1}^{I}(e(v^{w_i}) - e_d(v^{w_i}))^2}{MI}. \quad (1)$$

5.3 Replacing Strategy in Inference

Because we use Eq. (1) for model training, we hypothesize that $e(v)$, which contains definition information, would yield a low MSE, indicating that $e(v)$ is nearly equivalent to $e_d(v)$. Conversely, a high error in $e(v)$ suggests missing definition information or excessive noise, and should be replaced by $e_d(v)$. Therefore, in inference, we straightforwardly replace pre-trained embeddings with definition embeddings as $e(v) = e_d(v)$. Next, we discuss which tokens should be replaced. Given a specific downstream task, we remove tokens from $\mathcal{V}$ that do not appear in the corresponding fine-tuning dataset. The remaining set of tokens is denoted as $\mathcal{V}_{[task]}$.[5] Our preliminary experiments, presented in Appendix F, demonstrated that, when replacing the last tokens in $\mathcal{V}_{[task]}$, BART achieves the highest accuracy compared to replacing random or top tokens in $\mathcal{V}_{[task]}$. Therefore, as an initial study on constructing token embeddings for PLMs, this study focuses on definition embeddings for only low-frequency tokens, in line with previous

(a) Merged prompt

The definition of [word] is [definition] . Its part-of-speech , bpe-form without space , capitalization , and uppercase are [pos] , [wospace-word] , [cap-word] , and [upper-word] , respectively .

(b) Corrupted prompt

The definition of {corruption} is [definition] . Its part-of-speech , bpe-form without space , capitalization , and uppercase are [pos] , {corruption} , {corruption} , and {corruption} , respectively .

Figure 4: Constructed prompts. Brackets [] are a placeholder for the given word and its corresponding information. Texts with the same color indicate positions of a prompt and corresponding word information. {corruption} indicates the span for corrupted tokens. The bpe-form without space refers to the word's surface-form without the symbol "Ġ" when using the BART's tokenizer. Appendix E lists detailed examples.

[4] Our pre-experiments demonstrated that a large mask ratio would result in slow convergence, while a small ratio would cause limited change between $e(v)$ and $e_d(v)$. To ensure computational efficiency and to prevent the model from relying too heavily on unmasked tokens, we manually set these ratios.

[5] Tokens in $\mathcal{V}_{[task]}$ are also ranked in descending order according to their frequency in the dataset for pre-training.

Model	$I(\mathbf{E})\uparrow$			
	Frequent	Medium	Rare	All Tokens
RoBERTa	**0.694**	0.501	0.315	0.504
+ DelDirection	0.639	**0.641**	**0.599**	**0.624**
+ DefinitionEMB	0.649	0.470	0.382	0.519
BART	**0.851**	0.668	0.515	0.751
+ DelDirection	0.790	0.775	**0.731**	0.788
+ DefinitionEMB	0.834	**0.800**	0.603	**0.876**

Table 1: Isotropy of $\mathbf{E}$. The frequent (30%), medium (50%), and rare (20%) groups are determined based on the token index in $\mathcal{V}$. Appendix I shows projected $\mathbf{E}$.

Model	Spearman Score $\uparrow$				
	RG65	RW	SimLex	SimVerb	Ave
RoBERTa	16.05	18.89	26.67	11.81	18.36
+ DefinitionEMB	**18.88**	**18.96**	**27.15**	**11.91**	**19.23**
BART	15.32	19.66	28.56	**13.09**	19.16
+ DefinitionEMB	**15.67**	**19.76**	**28.63**	12.72	**19.20**

Table 2: Experimental results on the word similarity task with dot product. DefinitionEMB completely replaces $\mathbf{E}$. Appendix J shows the results with cosine similarity.

studies.[6] Considering diverse data distributions and task requirements for downstream tasks, we replace $\min(\alpha\% * N, |\mathcal{V}_{[task]}|)$ of the last tokens in $\mathcal{V}_{[task]}$, where α serves as a hyperparameter.

6 Experiments

In addition to evaluating the isotropy of $\mathbf{E}$, we evaluate the performance of PLMs before and after replacing embeddings by DefinitionEMB on various benchmark tasks, following Mu and Viswanath (2018) and Lewis et al. (2020), including a word similarity task, two natural language understanding tasks on General Language Understanding Evaluation (GLUE), and a text summarization task.

6.1 Experimental Settings

We adopted the RoBERTa-base and BART-large models as our baseline PLMs with encoder-only and encoder-decoder architectures, respectively. Both models used the same vocabulary $\mathcal{V}$ with the size $N = 50,265$. BART utilizes the weight tying (Press and Wolf, 2017) technique when predicting texts, which involves using $\mathbf{E}$ as the weight matrix for computing logits.

We utilized the English-language Wiktionary (1.5GB) as definitions for training DefinitionEMB. The 1,464,327 extracted definitions were randomly divided into 1,454,327 for training and 10,000 for validation. Additionally, we manually added definitions for 1,252 numbers in $\mathcal{V}$ by translating numbers into their corresponding words, such as "2" to "two". Furthermore, we added definitions for 136 named entity tokens in $\mathcal{V}$, such as "ĠNVIDIA", based on their Wikipedia pages or Google search results. Overall, 1,455,715 examples were used for training DefinitionEMB.[7] During inference, these examples were reused for loading definition embeddings.[8] Appendix G lists the hyperparameter settings for training DefinitionEMB and fine-tuning all models for downstream tasks. Experimental results were averaged over three trials. We tuned α (the ratio for replacing pre-trained token embeddings) based on the model performance on the validation set. The experimental results for tuning α are described in Appendix H.

6.2 Quantitative Evaluation

➢ **Initial Isotropy.** We measured the initial isotropy of token embeddings $\mathbf{E}$ in PLMs and the isotropy after completely replacing $\mathbf{E}$. Table 1 shows that both RoBERTa and BART exhibit high isotropy in the frequent group but low isotropy in the rare group. DelDirection helps achieve a uniformly distributed isotropy across frequency groups and results in the highest isotropy for the rare group, although it decreases isotropy in the frequent group. DefinitionEMB also showcases a more uniform isotropy distribution than the original PLMs, displaying lower isotropy for the frequent group but higher isotropy for the rare group. Additionally, DefinitionEMB for BART achieves the highest isotropy for the medium group as well as for the entire $\mathbf{E}$.

➢ **Word Similarity.** Compared to BART and RoBERTa, which utilize over 160 GB of contexts to learn semantic embeddings, DefinitionEMB utilizes definition information at only 1% of their contexts' size. To investigate whether DefinitionEMB can maintain original semantic relationships given such limited information, we adopted the word similarity task, following a previous study (Mu and Viswanath, 2018), to assess whether the similarity between the embeddings of two given words aligns with the ground truth, in terms of Spearman's rank correlation. We used dot product

[6] Following Bahdanau et al. (2018), handling unknown tokens $v \notin \mathcal{V}$ is deferred for future investigation.

[7] When using DefinitionEMB, 2305 tokens from $\mathcal{V}$ were always excluded from replacement because they do not have a corresponding definition, such as ")=(".

[8] Once a token embedding has been replaced, it will not be replaced during the rest of the procedure.

Model	SST	MRPC	STS	QQP	MNLI m	MNLI mm	QNLI	RTE	Average
RoBERTa	95.7	87.5	**89.6 / 89.0**	**89.6**	87.3	86.7	**93.1**	73.9	88.0
+ DelDefinition	**95.9**	86.9	89.0 / 88.3	89.3	87.3	86.8	93.0	72.3	87.6
+ DefinitionEMB	**95.9**	**87.7**	**89.6 / 89.0**	89.4	**87.6**	**87.0**	93.0	**75.3**	**88.3**
BART	**96.5**	87.8	91.2 / 90.6	**90.1**	**90.0**	**89.2**	94.7	82.4	90.3
+ DelDefinition	96.4	87.3	90.9 / 90.4	89.9	89.9	**89.2**	94.7	78.7	89.7
+ DefinitionEMB	96.4	**88.3**	**91.3 / 90.7**	**90.1**	**90.0**	**89.2**	**94.9**	**83.3**	**90.5**

Table 3: Experimental results on GLUE. For the STS dataset, we report the Pearson/Spearman's rank correlation, while for other datasets, we report accuracy scores. For the MNLI dataset, we report results for Matched (m) and Mismatched (mm) sets. Appendix K describes corresponding $I(\mathbf{E})$.

Model	CNNDM	Y-BIGPATENT	XSum	Billsum	Average
BART	43.57 / 20.93 / 40.31	43.96 / 18.92 / 37.80	43.76 / 20.40 / 34.65	**51.02** / 32.44 / 39.11	45.58 / 23.17 / 37.97
+DelDirection	43.59 / 20.93 / 40.32	43.91 / 18.85 / 37.79	43.90 / 20.58 / 34.86	50.89 / 32.22 / 38.97	45.57 / 23.15 / 37.99
+DefinitionEMB	**43.78**† / **20.94** / **40.52**†	**44.16**‡ / **19.06**‡ / **38.01**‡	**43.96**† / **20.61**† / **34.87**†	50.96 / **32.64**‡ / **39.28**	**45.72 / 23.31 / 38.17**

Table 4: Experimental results (ROUGE1-F1 / ROUGE2-F1 / ROUGEL-F1) on the text summarization task. † and ‡ indicate that the score is significantly superior to BART with a p-value < 0.01 and < 0.05, respectively.

to measure the similarity between word embeddings across four datasets: RG65 (Rubenstein and Goodenough, 1965), rare-words (RW) (Luong et al., 2013), SimLex-999 (Hill et al., 2015), and SimVerb-3500 (Gerz et al., 2016). We estimated the word embeddings by summing the embeddings of the corresponding tokens. As Table 2 shows, using DefinitionEMB yields higher Spearman scores than the original PLMs on the RG65, SimLex-999, and RW datasets. Notably, the results on the RW dataset, which consists of only rare words, underscore the effectiveness of DefinitionEMB in capturing semantic information for these words.

➤ **GLUE.** To evaluate the performance of PLMs with replaced embeddings in natural language understanding, we followed a previous study (Lewis et al., 2020) and conducted experiments on the GLUE benchmark (Wang et al., 2018) across seven datasets: SST, MRPC, STS, QQP, MNLI, QNLI, and RTE. We also report the test set results obtained from the public leaderboard.[9] Table 3 demonstrates that using DefinitionEMB improved the performance of both RoBERTa and BART on average, particularly for the MRPC and RTE datasets. This may be because the frequency of tokens in the two classification datasets is low, causing rare tokens to be insufficiently fine-tuned, thus enhancing the necessity for their embeddings to be replaced.

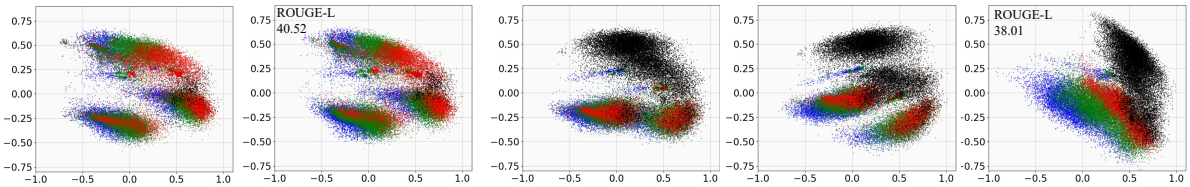
➤ **Text Summarization.** For the downstream summarization task, we used public abstractive summarization datasets, including CNN/DailyMail (CNNDM) (Hermann et al., 2015), Extreme Summarization (XSum) (Narayan et al., 2018), BillSum (Kornilova and Eidelman, 2019), and Y-BIGPATENT (Sharma et al., 2019). We evaluate the model performance on these datasets using the ROUGE scores (Lin, 2004) and compare BART + DefinitionEMB with the original BART using paired bootstrap resampling (Koehn, 2004) for the significance test. Table 4 shows that using DelDirection improved the ROUGEL-F1 score by 0.21 points for BART on the XSum dataset. However, the difference between BART and DelDirection is very limited on other datasets; DelDirection achieves even a lower ROUGEL-F1 score, with a decrease of 0.14 points, than BART on the Billsum dataset. In contrast, by improving semantics-related information and frequency-aware $I(\mathbf{E})$ for rare tokens, DefinitionEMB achieves the highest ROUGEL-F1 scores, with improvements of 0.21, 0.21, 0.22, and 0.17 points on the CNNDM, Y-BIGPATENT, XSum, and Billsum datasets, respectively, for BART.

7 Analysis of DefinitionEMB

7.1 Ablation Study for Replacing Tokens

We conducted an ablation study to analyze the effectiveness of replacing only appearing tokens instead of all tokens. The index range of replaced tokens is denoted as $[X, N]$, and the number of tokens appearing in $[X, N]$ satisfies $\min(\alpha\% *$

[9] https://gluebenchmark.com/

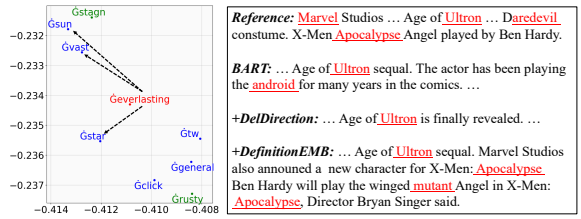


(a) +DefinitionEMB, $I(E) = 0.808$, before CNNDM (b) + DefinitionEMB, $I(E) = 0.792$, after CNNDM 20,000 steps (c) + DefinitionEMB, $I(E) = 0.764$, before Y-BIGPATENT (d) + DefinitionEMB, after Y-BIGPATENT 7,000 steps (e) + DefinitionEMB, $I(E) = 0.676$, after Y-BIGPATENT 92,000 steps

Figure 5: Projected token embeddings in BART+DefinitionEMB before and after fine-tuning. The embeddings in (a) and (c) exhibit different shapes due to the different α.

Replaced tokens	**Y-BIGPATENT** ($X = 23,000$)	**Billsum** ($X = 41,000$)
Appearing	**44.16 / 19.06 / 38.01**	50.96 / **32.64 / 39.28**
Both	43.93 / 18.79 / 37.76	**51.23** / 32.44 / 39.20

Table 5: Replacing appearing tokens only vs. replacing both appearing and non appearing tokens for BART on the Y-BIGPATENT and Billsum test sets. Appendix L describes the results for RoBERTa and BART on the MRPC test set.



(a) Token Embeddings before fine-tuning. (b) Generated summarization.

Figure 6: Case study on the CNNDM dataset. (a) visualizes the projected token embeddings in BART+DefinitionEMB. Underline in (b) indicates the rare tokens with index larger than 40,000 in $\mathcal{V}$. See Appendix M for a full version.

$N, |\mathcal{V}_{[task]}|$), as required in Section 5.3.

Results for BART+DefinitionEMB are reported in Table 5. When replacing only appearing tokens, the model achieved higher ROUGE scores than when replacing all tokens. Specifically, ROUGEL-F1 improved by 0.25 and 0.08 for Y-BIGPATENT and Billsum, respectively. This difference may be caused by the varying token frequencies in the training sets.

7.2 Embedding Dynamics

Figure 5 depicts the projected token embeddings of BART+DefinitionEMB on the CNNDM and Y-BIGPATENT datasets. On the CNNDM dataset, the BART+DefinitionEMB tokens exhibit minimal drift from before to after fine-tuning. Conversely, on the Y-BIGPATENT dataset, BART+DefinitionEMB tokens within the same group move together after fine-tuning. These findings align with those of BART in Figure 1, indicating that using DefinitionEMB helps maintain BART's robustness to degeneration into a narrow cone. Additionally, using DefinitionEMB before fine-tuning increases $I(\mathbf{E})$ for BART across the CNNDM and Y-BIGPATENT datasets, supporting our observation that, compared with BART, embeddings in BART+DefinitionEMB with different frequencies are more thoroughly mixing with each other. Figure 6 (a) provides a closer view of Figure 5 (a), revealing that the token "Ġeverlasting" is surrounded by more semantically related tokens, including "Ġsun", "Ġstar", and "Ġvast", than BART with and without DelDirection.

8 Conclusion

In this study, we found that during fine-tuning on text summarization tasks, the embeddings of the BART-large model do not degenerate into a narrow cone in a low-dimensional space. However, eliminating specific directions from BART's embeddings using Mu and Viswanath (2018) leads to degeneration, where embeddings with different token frequencies are pushed away from each other. Experimental results demonstrated that using DefinitionEMB for RoBERTa and BART improves the distribution of embeddings and enables low-frequency token embeddings to retain semantics-related information. DefinitionEMB also maintained BART's robustness to degeneration. While PLMs suffer from stale information, DefinitionEMB provided access to external data to construct embeddings for OOV tokens. Furthermore, our observed embedding dynamics paved the way for future work to explore why PLMs, such as BART, are robust to degeneration.

9 Limitations

The scope of this paper is limited to the investigation of two early transformer-based pre-trained language models, RoBERTa and BART, with encoder-only and encoder-decoder architectures, respectively. The effectiveness of using definitions for decoder-only transformer-based or transformer-unrelated pre-trained language models was not discussed. Additionally, our experiments focused exclusively on embeddings for tokens within the predefined vocabulary, and the effectiveness of utilizing DefinitionEMB for unknown tokens remains unexplored. Furthermore, while we directly employed a 2:1:1 strategy for corrupting tokens in the encoder-only reader, the optimal value for this strategy is worth exploring. Lastly, tuning α will be inefficient if the fine-tuning process for a given dataset is slow, and the improvements would be limited for tasks mainly comprising popular tokens.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. *Technique Report from Website*.

David Paul Ausubel. 1968. *Educational psychology: A cognitive view*. Holt, Rinehart and Winston.

Dzmitry Bahdanau, Tom Bosc, Stanisław Jastrzębski, Edward Grefenstette, Pascal Vincent, and Yoshua Bengio. 2018. Learning to compute word embeddings on the fly. *arXiv preprint arXiv:1706.00286*.

Daniel Biś, Maksim Podkorytov, and Xiuwen Liu. 2021. Too much in common: Shifting of embeddings in transformer language models and its implications. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5117–5130.

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. 2021. Isotropy in the contextual embedding space: Clusters and manifolds. In *International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

Jun Gao, Di He, Xu Tan, et al. 2019. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations*.

Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. SimVerb-3500: A large-scale evaluation set of verb similarity. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2173–2182.

Chengyue Gong, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2018. Frage: Frequency-agnostic word representation. In *Advances in Neural Information Processing Systems*, volume 31.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, et al. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, volume 28.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Rob J Hyndman and George Athanasopoulos. 2018. *Forecasting: principles and practice*. OTexts.

Mikhail Khodak, Nikunj Saunshi, Yingyu Liang, Tengyu Ma, Brandon Stewart, and Sanjeev Arora. 2018. A la carte embedding: Cheap but effective induction of semantic feature vectors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 12–22.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395.

Anastassia Kornilova and Vladimir Eidelman. 2019. BillSum: A corpus for automatic summarization of US legislation. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 48–56.

Mike Lewis, Yinhan Liu, Naman Goyal, et al. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81.

Qianchu Liu, Diana McCarthy, and Anna Korhonen. 2019a. Second-order contexts from lexical substitutes for few-shot learning of word representations. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM 2019)*, pages 61–67.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, et al. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Thang Luong, Richard Socher, and Christopher Manning. 2013. Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113.

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

Jiaqi Mu and Pramod Viswanath. 2018. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations*.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807.

Myle Ott, Sergey Edunov, Alexei Baevski, et al. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 48–53.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, et al. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 2463–2473.

Yuval Pinter, Robert Guthrie, and Jacob Eisenstein. 2017. Mimicking word embeddings using subword RNNs. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 102–112.

Ofir Press and Lior Wolf. 2017. Using the output embedding to improve language models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 157–163.

Colin Raffel, Noam Shazeer, Adam Roberts, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1).

Sara Rajaee and Mohammad Taher Pilehvar. 2021. A cluster-based approach for improving isotropy in contextual embedding space. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 575–584.

Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Elena Sofia Ruzzetti, Leonardo Ranaldi, Michele Mastromattei, Francesca Fallucchi, Noemi Scarpato, and Fabio Massimo Zanzotto. 2022. Lacking the embedding of a word? look it up into a traditional dictionary. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2651–2662.

Shota Sasaki, Jun Suzuki, and Kentaro Inui. 2019. Subword-based Compact Reconstruction of Word Embeddings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3498–3508.

Timo Schick and Hinrich Schütze. 2019. Attentive mimicking: Better word embeddings by attending to informative contexts. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 489–494.

Timo Schick and Hinrich Schütze. 2020. Rare words: A major problem for contextualized embeddings and how to fix it by attentive mimicking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8766–8774.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Eva Sharma, Chen Li, and Lu Wang. 2019. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213.

Julien Tissier, Christophe Gravier, and Amaury Habrard. 2017. Dict2vec : Learning word embeddings using lexical dictionaries. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 254–263.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394.

Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.*, 11:3371–3408.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing*

and Interpreting Neural Networks for NLP, pages 353–355.

Thomas Wolf, Lysandre Debut, Victor Sanh, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Yonghui Wu, Mike Schuster, Zhifeng Chen, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Sangwon Yu, Jongyoon Song, Heeseung Kim, Seongmin Lee, Woo-Jong Ryu, and Sungroh Yoon. 2022. Rare tokens degenerate all tokens: Improving neural text generation via adaptive gradient gating for rare token embeddings. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 29–45.

Zhong Zhang, Chongming Gao, Cong Xu, Rui Miao, Qinli Yang, and Junming Shao. 2020. Revisiting representation degeneration problem in language modeling. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 518–527.

George Kingsley Zipf. 1949. *Human behavior and the principle of least effort.* Addison-Wesley Press.

A Isotropy Metric

$\mathbf{E}$ denotes the pre-trained token embedding matrix of the PLM. Following previous studies (Mu and Viswanath, 2018; Biś et al., 2021; Yu et al., 2022), we compute the isotropy of $\mathbf{E}$ using Equation (1) from Mu and Viswanath (2018), which is given by:

$$I(\mathbf{E}) = \frac{\min_{\mathbf{b}\in\mathcal{B}} Z(\mathbf{b})}{\max_{\mathbf{b}\in\mathcal{B}} Z(\mathbf{b})}, \quad (1)$$

where $Z(\mathbf{b})$ is approximately constant, $\mathcal{B}$ is the set of eigenvectors of $\mathbf{E}^T\mathbf{E}$ with T represents transposition operation.

B Datasets

Table 6 provides an overview of the data statistics for each task. For the GLUE task, we utilize Stanford Sentiment Treebank (SST), Microsoft Research Paraphrase Corpus (MRPC), Semantic Textual Similarity Benchmark (STS), Quora Question Pairs (QQP), MultiNLI (MNLI), Question NLI (QNLI), and Recognizing Textual Entailment (RTE) datasets. For the text summarization task, CNNDM comprises articles from the CNN and the Daily Mail newspapers, while Xsum consists of BBC articles paired with single-sentence summaries. BillSum contains summaries of US Congressional and California state bills, and Y-BIGPATENT contains U.S. patent documents covering new or cross-sectional technology.

Figures 7 and 8 show the distribution of tokens appearing in the GLUE and text summarization datasets. As these two figures show, the token frequency in text summarization datasets is much higher than that in the GLUE datasets. This findings suggest a potential difference when fine-tuning embeddings with respect to the task requirements.

C Additional Embedding Dynamics

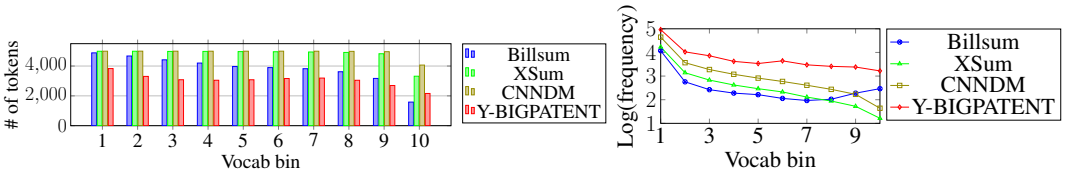
Figures 9 (e) and (g) show that before fine-tuning, the embeddings of DelDirection form a narrow ellipse. After fine-tuning, the minor axis of the ellipse lengthens, and the frequent, medium, and rare groups gradually disperse, eventually forming a square, as shown in Figures 9 (f) and (h). From Figures 9 (f) and (h), we also observe that the more updating steps, the longer the original minor axis of the ellipse, and the higher $I(\mathbf{E})$ achieved by DelDirection. However, if DelDirection is fine-tuned further, the shape may resemble that in Figure 1 (h), resulting in a much smaller $I(\mathbf{E})$. Comparing Figures 9 (a) to (d) with Figures 9 (i) to (ℓ), we observe that BART + DefinitionEMB performs similarly to BART. Specifically, on the XSum dataset, there is minimal drift in embeddings from before to after fine-tuning. Conversely, for the Billsum dataset, after fine-tuning, the embeddings of two ellipses move closer together.

We do not observe drift for BART-related models on the QQP and RTE datasets, as shown in Figure 10, which is different from text summarization datasets. This may be because the BART for classification tasks does not use the weight tying technique, and the token frequency in QQP and RTE are much lower than in text summarization datasets. Using DelDefinition for RoBERTa shows the spread of popular tokens from the original center as shown in Figures 11 (e) and (f).

Figures 12, 13, and 14 show case studies of specific tokens before and after replacing their token embeddings. In BART, the central tokens are surrounded by tokens of the same frequency, rather than those with related semantics. In the case of BART+DelDirection, we observe tokens with different token frequencies surrounding the central word. However, using DelDirection also does not guarantee the semantically related neighbors. After replacing embeddings with DefinitionEMB, semantically related tokens appear in the surrounding of the central tokens.

Figure 15 depicts the projection of contextual embeddings[10]. Although BART + DelDirection (Figure 1 (g)) exhibits a totally different token embedding distribution from BART (Figure 1 (b)), it yield a similar contextual embeddings as BART. Specifically, Figures 15 (a) and (b) show that high-frequency tokens are closely grouped together based on their frequency, while low-frequency tokens are spread out in specific directions. However, when using DefinitionEMB, we observe a more concentrated distribution than BART. The projection resembles concentric ellipses, where tokens with similar frequencies are placed in the same ellipse. This finding may be related to DefinitionEMB yielding higher ROUGE scores than BART and BART + DelDirection.

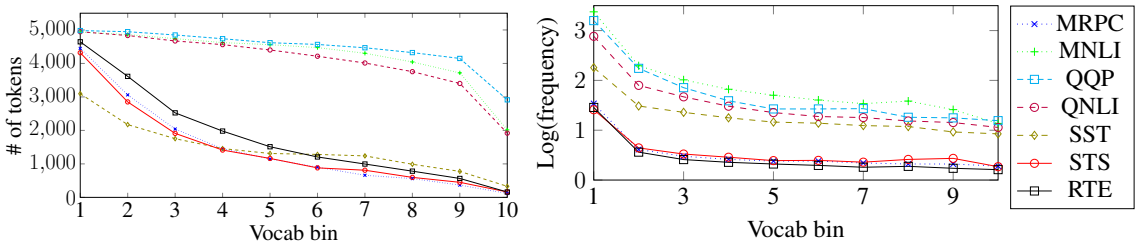
[10]The specific decoder layer hidden states of the token in a given context (Cai et al., 2021).



(a) Number of tokens appearing in $\mathcal{V}$.

(b) Logarithmic averaged frequency of tokens appearing in the corresponding training set.

Figure 7: Distribution of the BART vocabulary in the training sets of text summarization datasets, considering both source and target tokens. The first 50,000 tokens in $\mathcal{V}$ are grouped into bins of 5,000 according to their index in $\mathcal{V}$ (e.g., [0:4,999]).



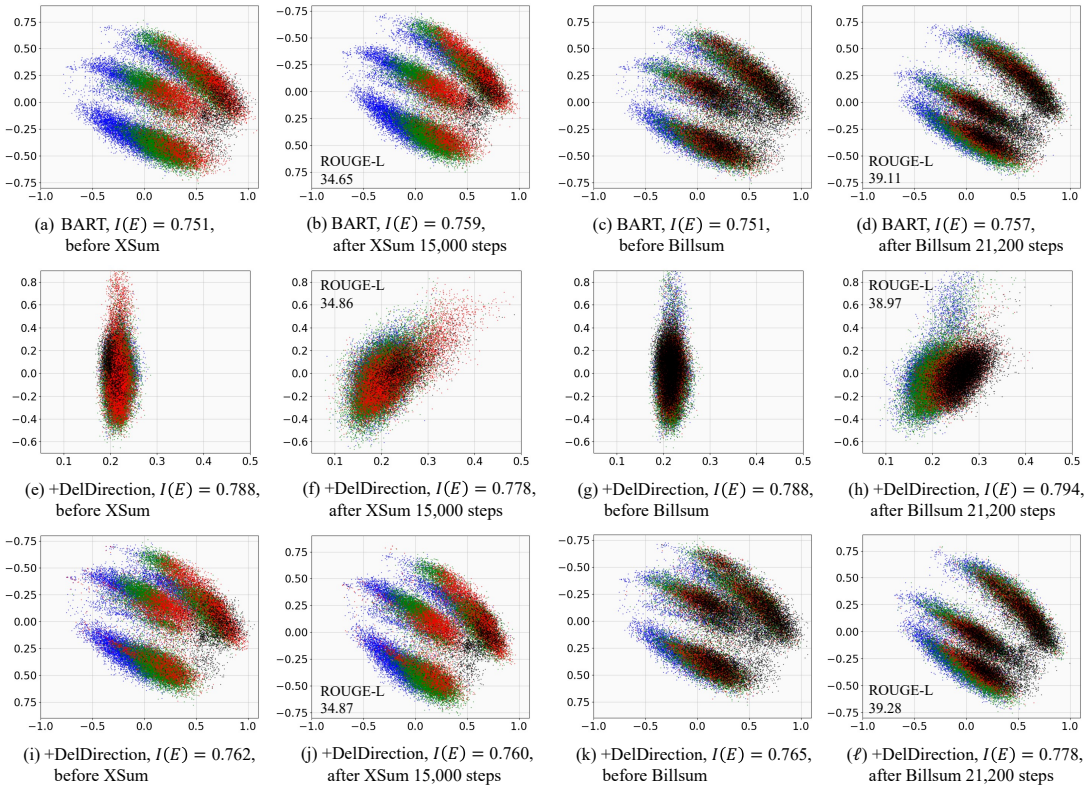
(a) Number of tokens appearing in $\mathcal{V}$.

(b) Logarithmic averaged frequency of tokens appearing in the corresponding training set.

Figure 8: Distribution of the BART vocabulary in the training sets of GLUE datasets.

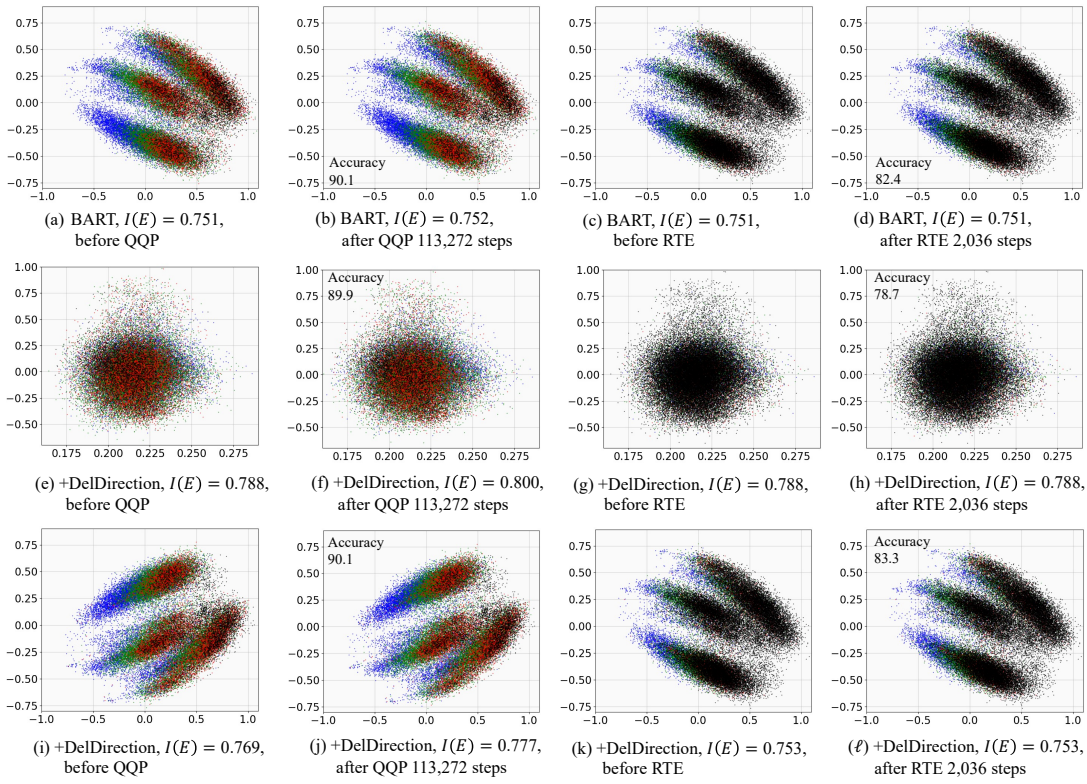
Task	**Dataset**	**# of train**	**# of validation**	**# of test**
Word similarity	RG65	-	-	65
	SimLex-999	-	-	999
	RW	-	-	2,034
	SimVerb-3500	-	-	3,500
GLUE	RTE	2, 490	277	3, 000
	MRPC	3, 668	408	1, 725
	STS	5, 749	1, 500	1, 379
	SST	67, 349	872	1, 821
	QNLI	104, 743	5, 463	5, 463
	QQP	363, 846	40, 430	390, 965
	MNLI	392, 702	9, 815 (m) + 9, 832 (mm)	9, 796 (m) + 9, 847 (mm)
Text summarization	BillSum	17, 054	1, 895	3, 269
	Y-BIGPATENT	124, 397	6, 911	6, 911
	XSum	204, 045	11, 332	11, 334
	CNNDM	287, 227	13, 368	11,490

Table 6: Detailed statistic of train, validation and test Datasets. For the MNLI dataset, we report Matched (m) and Mismatched (mm) sets.



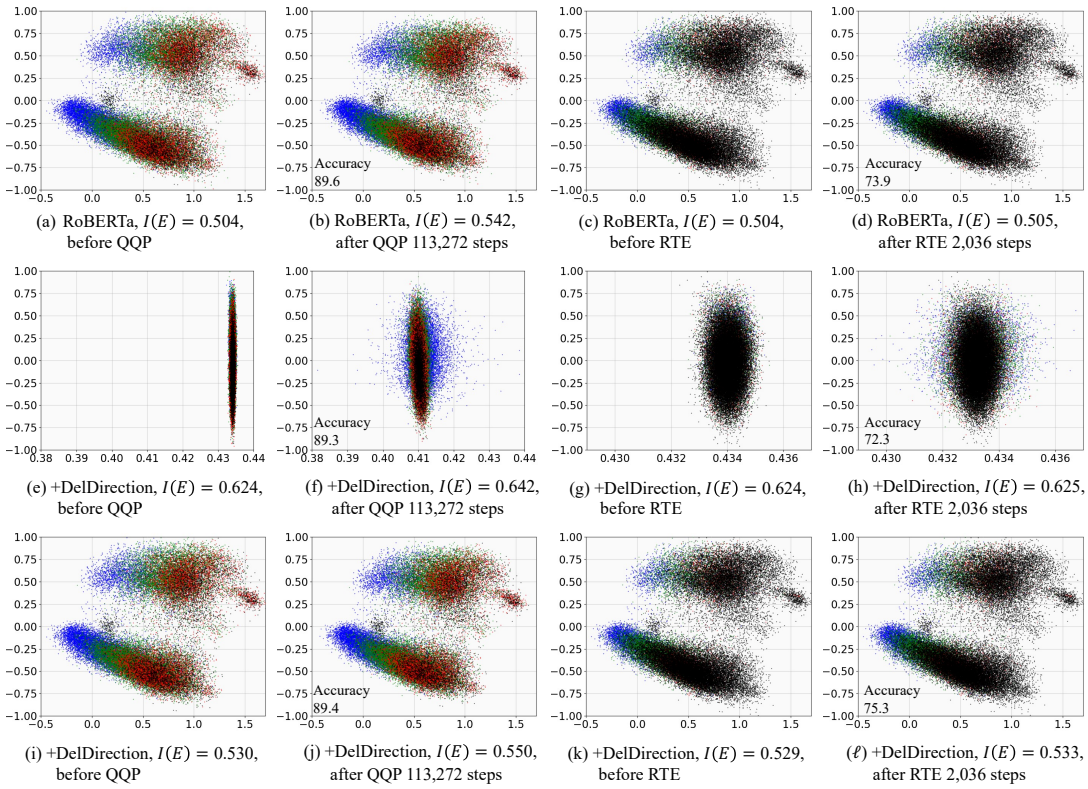
(a) BART, $I(E) = 0.751$, before XSum
(b) BART, $I(E) = 0.759$, after XSum 15,000 steps
(c) BART, $I(E) = 0.751$, before Billsum
(d) BART, $I(E) = 0.757$, after Billsum 21,200 steps
(e) +DelDirection, $I(E) = 0.788$, before XSum
(f) +DelDirection, $I(E) = 0.778$, after XSum 15,000 steps
(g) +DelDirection, $I(E) = 0.788$, before Billsum
(h) +DelDirection, $I(E) = 0.794$, after Billsum 21,200 steps
(i) +DelDirection, $I(E) = 0.762$, before XSum
(j) +DelDirection, $I(E) = 0.760$, after XSum 15,000 steps
(k) +DelDirection, $I(E) = 0.765$, before Billsum
(ℓ) +DelDirection, $I(E) = 0.778$, after Billsum 21,200 steps

Figure 9: Projected token embeddings of BART-related models before and after fine-tuning on the XSum and Billsum datasets. 30%, 50%, and 20% of the appearing tokens in $\mathcal{V}$ are assigned to the frequent, medium, and rare groups, respectively, based on their frequency in the fine-tuning set.



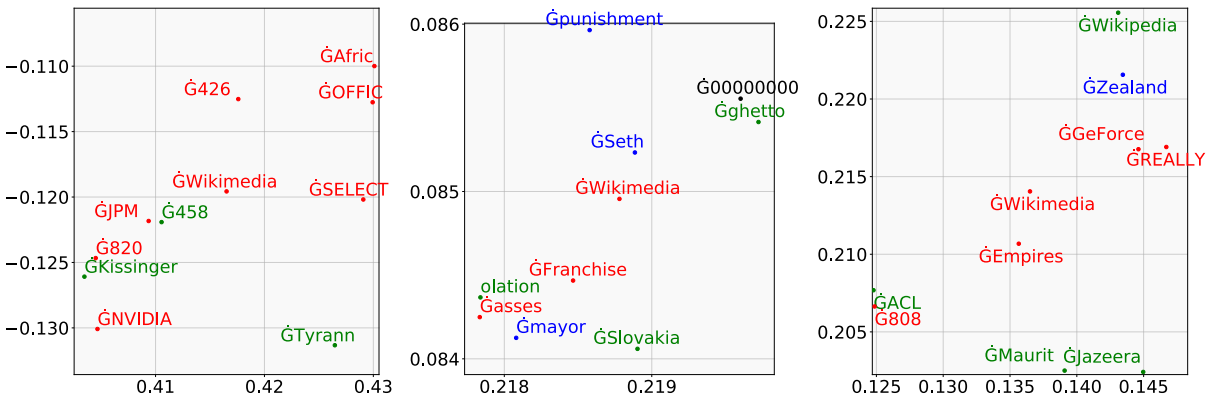
(a) BART, $I(E) = 0.751$, before QQP
(b) BART, $I(E) = 0.752$, after QQP 113,272 steps
(c) BART, $I(E) = 0.751$, before RTE
(d) BART, $I(E) = 0.751$, after RTE 2,036 steps
(e) +DelDirection, $I(E) = 0.788$, before QQP
(f) +DelDirection, $I(E) = 0.800$, after QQP 113,272 steps
(g) +DelDirection, $I(E) = 0.788$, before RTE
(h) +DelDirection, $I(E) = 0.788$, after RTE 2,036 steps
(i) +DelDirection, $I(E) = 0.769$, before QQP
(j) +DelDirection, $I(E) = 0.777$, after QQP 113,272 steps
(k) +DelDirection, $I(E) = 0.753$, before RTE
(ℓ) +DelDirection, $I(E) = 0.753$, after RTE 2,036 steps

Figure 10: Projected token embeddings of BART-related models before and after fine-tuning on the QQP and RTE datasets.



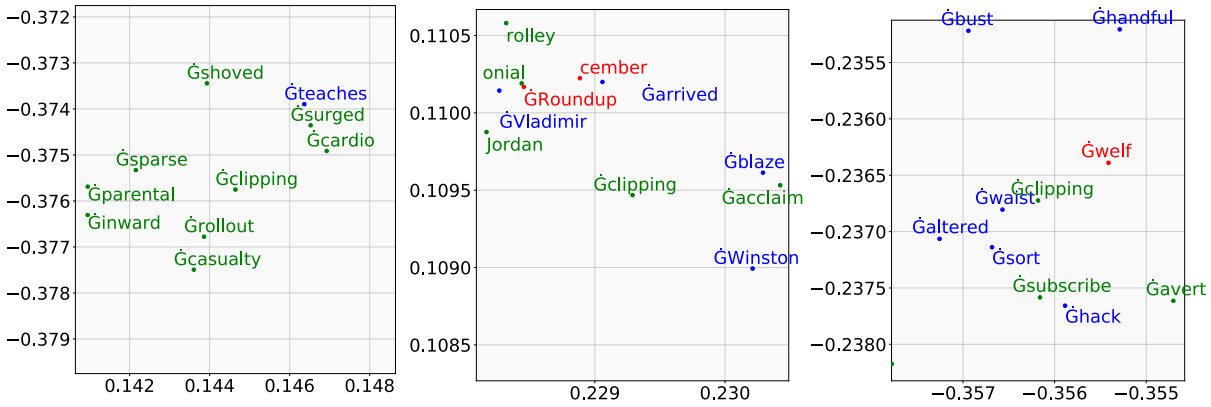
(a) RoBERTa, $I(E) = 0.504$, before QQP
(b) RoBERTa, $I(E) = 0.542$, after QQP 113,272 steps
(c) RoBERTa, $I(E) = 0.504$, before RTE
(d) RoBERTa, $I(E) = 0.505$, after RTE 2,036 steps
(e) +DelDirection, $I(E) = 0.624$, before QQP
(f) +DelDirection, $I(E) = 0.642$, after QQP 113,272 steps
(g) +DelDirection, $I(E) = 0.624$, before RTE
(h) +DelDirection, $I(E) = 0.625$, after RTE 2,036 steps
(i) +DelDirection, $I(E) = 0.530$, before QQP
(j) +DelDirection, $I(E) = 0.550$, after QQP 113,272 steps
(k) +DelDirection, $I(E) = 0.529$, before RTE
(ℓ) +DelDirection, $I(E) = 0.533$, after RTE 2,036 steps

Figure 11: Projected token embeddings of RoBERTa-related models before and after fine-tuning on the QQP and RTE datasets.



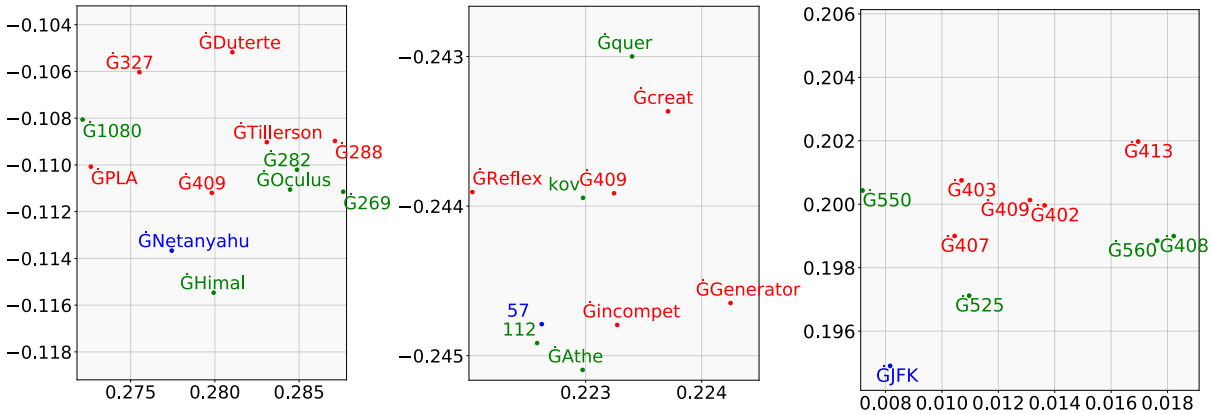
(a) BART
(b) + DelDirection
(c) + DefinitionEMB

Figure 12: Case study of the token "ĠWikimedia" and its surrounding tokens. Non appearing, rare, medium, and frequent groups in the CNNDM dataset are represented by black, red, green, and blue points respectively.



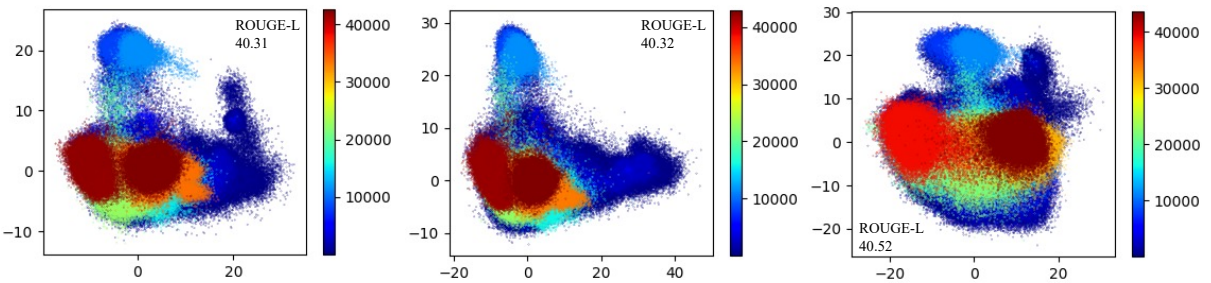
(a) BART (b) + DelDirection (c) + DefinitionEMB

Figure 13: Case study of the token embeddings of the token "Ġclipping" and its surrounding tokens.



(a) BART (b) + DelDirection (c) + DefinitionEMB

Figure 14: Case study of the token embeddings of the token "Ġ409" and its surrounding tokens.



(a) BART (b) + DelDirection (c) + DefinitionEMB

Figure 15: Projected contextual embedding in the 12th (final) decoder layer of BART-related models after fine-tuning on the CNNDM dataset. Colors indicate the token frequencies in corresponding test set.

D Semantically Related Tokens Recognized by Large Language Models

Table 7 shows the query we used to ask ChatGPT 3.5 and Claude 3 Haiku about the semantic relationship between "Ġeverlasting" and other tokens, along with their corresponding responses for semantics-related tokens only.

E Corrupted Prompts

Figure 16 shows examples of constructing merged and corrupted prompts for the word "discomfort".

Query	Please help me to check the semantic relationship between the token “everlasting” and other tokens, including reassure, legalize, validate, underworld, typo, misdemeanor, jurisd, scripting, APIs, io, bye, complaining, LAB, arcade, incentive, hematically, mind, stagn, sun, vast, star, tw, general, click, rusty. Please return scores in the range [0,1] to indicate their semantic relationship, and tell me the reason.
ChatGPT 3.5	validate: 0.3 - "Validate" can mean to confirm or verify the accuracy or truth of something, which could have a moderate semantic relationship with "everlasting" if used in a context related to eternal validity or truth. mind: 0.2 - "Mind" refers to the faculty of consciousness and thought, which could have a moderate semantic relationship with "everlasting" if used in a context related to eternal consciousness or existence. sun: 0.3 - "Sun" refers to the star around which the Earth orbits and has a moderate semantic relationship with "everlasting" due to its association with eternal cycles and longevity. vast: 0.3 - "Vast" refers to something extremely large or extensive, which could have a moderate semantic relationship with "everlasting" if used in a context related to eternal expansiveness. star: 0.3 - "Star" refers to a celestial body that emits light and has a moderate semantic relationship with "everlasting" due to its association with eternal cycles and longevity.
Claude 3 Haiku	"everlasting" and "validate": Score: 0.4 Reason: "Everlasting" and "validate" share a sense of permanence and confirmation, as "validate" means to confirm or corroborate something as true or correct, which is somewhat related to the idea of something being eternal or lasting forever. "everlasting" and "mind": Score: 0.4 Reason: "Everlasting" suggests something that lasts forever, while the "mind" is a key part of human consciousness and experience, which are somewhat related in the sense that the human mind may be seen as an eternal or enduring aspect of the self. "everlasting" and "sun": Score: 0.4 Reason: "Everlasting" suggests something that lasts forever, and the "sun" is a celestial body that has existed for billions of years and is expected to continue shining for a very long time, so there is a connection between the idea of something being "everlasting" and the sun’s enduring presence. "everlasting" and "vast": Score: 0.5 Reason: "Everlasting" suggests something that is immense or boundless in its duration, while "vast" describes something that is immense in size or scope, so there is a semantic overlap between the two terms in terms of conveying a sense of immensity or grandeur. "everlasting" and "star": Score: 0.4 Reason: Similar to the connection with the "sun", the "star" is another celestial body that exists for long durations, so the idea of something being "everlasting" is somewhat related to the enduring nature of stars.

Table 7: The query to ChatGPT 3.5 and Claude 3 Haiku, along with part of their responses.

(a) Merged prompt

The _ definition _ of _ **discomfort** _ is _ **To** _ **cause** _ **annoyance** _ **or** _ **distress** _ **to** _ . _ Its _ part-of-speech _ , _ bpe-form _ without _ space _ , _ capitalization _ , _ and _ uppercase _ are _ **verb** _ , **discomfort** _ , _ **Discomfort** _ , _ and _ **DISCOMFORT** _ , _ respectively _ .

(b) Source for the encoder-decoder reader

The _ definition _ of <MASK$_1$> _ is _ **To** _ **cause** _ **annoyance** _ **or** _ **distress** _ **to** _ . _ Its _ part-of-speech _ , _ bpe-form _ without _ space _ , _ capitalization _ , _ and _ uppercase _ are _ **verb** _ , <MASK$_2$> _ , <MASK$_3$> _ , _ and <MASK$_4$> _ , _ respectively _ .

(c) Target for the encoder-decoder reader

<MASK$_1$> _ discomfort <MASK$_2$> discomfort <MASK$_3$> _ Discomfort <MASK$_4$> _ DISCOMFORT

(d) Source for the encoder-only reader during training

The _ definition _ of <MASK> _ is _ **To** _ **cause** _ **annoyance** _ **or** _ **distress** _ **to** _ . _ Its _ part-of-speech _ , _ bpe-form _ without _ space _ , _ capitalization _ , _ and _ uppercase _ are _ **verb** _ , <MASK>**comfort** _ , **ĠDis** <MASK> _ , _ and <MASK> **COM** *BS ĠNations* _ , _ respectively _ .

(e) Source for the encoder-only reader during inference

The _ definition _ of <MASK> _ is _ **To** _ **cause** _ **annoyance** _ **or** _ **distress** _ **to** _ . _ Its _ part-of-speech _ , _ bpe-form _ without _ space _ , _ capitalization _ , _ and _ uppercase _ are _ **verb** _ , **discomfort** _ , _ **Discomfort** _ , _ and _ **DISCOMFORT** _ , _ respectively _ .

Figure 16: Example of constructing prompts for the word “discomfort”. “_” is whitespace. <MASK$_i$> denotes the ith mask token. *Italic* indicates randomly replaced tokens.

Strategy	RoBERTa	BART
None	87.5	87.8
Random	87.4	87.3
Top	86.0	87.6
Last	**87.7**	**88.3**

Table 8: Experimental results of replacing tokens in $\mathcal{V}_{MRPC}$ with DefinitionEMB ($\alpha = 5$) for RoBERTa-base and BART-large on the MRPC test set using three different strategies.

F Pre-experiment for Replacing Strategy

Considering the appearance bias as indicated in Figures 1 (e) and (j), we conducted pre-experiments to investigate which type of tokens in $\mathcal{V}_{[task]}$ should be replaced. DefinitionEMB replaced the embeddings of $\min(\alpha\% * N, |\mathcal{V}_{[task]}|)$ of tokens in $\mathcal{V}_{[task]}$ using one of the following strategies:

Random: Randomly replace tokens.

Top: Replace tokens with smallest indexes, where index ≥ 5000.

Last: Replace tokens with largest indexes.

As shown in Table 8, when replacing the last tokens in $\mathcal{V}_{[task]}$ results in the highest accuracy for both RoBERTa and BART on the MRPC test set, among all strategies considered.

G Hyperparameters

Table 9 lists the artifacts utilized in the study. We used the Fairseq (Ott et al., 2019) and HuggingFace (Wolf et al., 2020) to reproduce all models and run the downstream tasks.

We adhered to the original fine-tuning settings of RoBERTa and BART on the GLUE task and CNNDM dataset. Details of the hyperparameter settings for training DefinitionEMB and fine-tuning models are outlined in Tables 10, 11 and 12. During the training of DefinitionEMB, we utilized the Adam optimizer with a batch size of 4096 tokens and a 0.0001 learning rate with an "inverse square root" schedule. For fine-tuning models on GLUE and text summarization tasks, we employed the Adam optimizer and utilized a "polynomial decay" learning rate schedule. To reduce computation, we freeze the embedding layer of DefinitionEMB during training for BART.

H Tuning α

For each downstream task, we tuned α with a single trial on the corresponding validation set. Tables 13 to 23 display the performance of Basline+DefinitonEMB with various α. Tables 24 and 25 provide the tuned α for each downstream dataset. Overall, datasets in the GLUE task exhibit smaller α than those in the text summarization task. This may be because the text summarization task involves a larger number of input tokens than the GLUE task. Additionally, the text summarization task involves predicting tokens, whereas BART employs a weight tying technique to connect token embeddings for prediction, enabling token embeddings to be updated more during fine-tuning than in the GLUE task. Among all datasets, the SST dataset has the smallest tuned α, set at 1. This could be attributed to the dataset having the fewest unique tokens. The STS, RTE, and MPRC datasets have a similar number of unique tokens, slightly higher than that of the SST dataset, resulting in α values of 3 and 5. In the GLUE task, the QQP, QNLI, and MNLI datasets have the highest number of unique tokens, leading to α values for BART of 3, 5, and 10, respectively. However, tuned α values of the three datasets for RoBERTa are all set to 3: this may be because MSE is larger in RoBERTa than in BART as discussed in Appendix N. Among the text summarization datasets, Billsum has the lowest token frequencies, resulting in the smallest α value of 7. Despite having the highest token frequencies, the Y-BIGPATENT dataset has the lowest number of unique tokens, resulting in an α value of only 30. The CNNDM dataset yields the highest α value, set at 100, possibly due to its most uniformly distributed and largest number of unique tokens, and larger training corpus. Although the XSum dataset also contains a large number of unique tokens, it has a smaller training corpus than CNNDN, resulting in a smaller α value of 10. These findings suggest a potential relationship between the distribution of training data and the value of α. Specifically, datasets with a larger number of unique tokens, along with more training examples, tend to result in higher α values.

Used artifacts	Note
RoBERTa	https://huggingface.co/roberta-base
BART	https://huggingface.co/facebook/bart-large
BART for GLUE	https://github.com/facebookresearch/fairseq/blob/main/examples/bart/README.glue.md
BART for CNNDM	https://github.com/facebookresearch/fairseq/blob/main/examples/bart/README.summarization.md
Wiktionary	https://en.wiktionary.org/wiki/Wiktionary:Main_Page
Wiktionary(extracted)	https://github.com/tatuylonen/wiktextract/tree/master
Isotropy metric	https://github.com/danielbis/tooMuchInCommon/blob/main/src/isotropy.py
View contextual embedding	https://github.com/TideDancer/iclr21_isotropy_contxt
Files2rouge	https://github.com/pltrdy/files2rouge
Paired bootstrap resampling	https://github.com/neubig/util-scripts/blob/master/paired-bootstrap.py
Fairseq	https://github.com/facebookresearch/fairseq/
HuggingFace	https://github.com/huggingface/transformers/
ChatGPT 3.5	https://chat.openai.com/
Claude 3 Haiku	https://claude.ai/chat/

Table 9: Used artifacts.

Hyperparameters	SST	MRPC	STS	QQP	MNLI	QNLI	RTE
# of updates	20,935	2,296	3,598	113,272	123,873	33,112	2,036
# of warm-up updates	1,256	137	214	28,318	7,432	1,986	122
Batch size (sentences)	32	16	16	32	32	32	16
Learning rate	1e-5	1e-5	2e-5	1e-5	1e-05	1e-05	2e-05

Table 10: Hyperparameters used for fine-tuning RoBERTa-related models across different datasets.

Hyperparameters	SST	MRPC	STS	QQP	MNLI	QNLI	RTE	CNNDM	Y-BIGPATENT	XSUM	Billsum
# of updates	7,150	700	1,800	113,920	43,210	33,290	1020	20,000	92,880	15,000	21,320
# of warm-up updates	429	42	108	6,835	2,593	1,997	61	500	7,430	500	1,705
Batch size (sentences)	128	64	32	32	256	32	32	-	-	-	-
Batch size (tokens)	-	-	-	-	-	-	-	65,536	8,192	32,768	8,192
Learning rate	5e-6	2e-5	2e-5	1e-5	5e-6	1e-5	1e-5	3e-05	3e-5	3e-05	3e-5

Table 11: Hyperparameters used for fine-tuning BART-related models across different datasets.

Hyperparameters	RoBERTa	BART
# of updates	400,000	250,000
# of warm-up updates	24,000	20,000

Table 12: Hyperparameters used for training DefinitionEMB.

Baseline \ α	1	5	10	20
RoBERTa	89.2	**90.2**	89.5	89.7
BART	88.2	**90.2**	88.7	89.2

Table 13: Accuracy for Baseline+DefinitionEMB with various α on the MRPC validation set.

Baseline \ α	1	5	10	20
RoBERTa	**95.1**	94.8	94.8	94.4
BART	**96.2**	96.1	95.6	95.3

Table 14: Accuracy for Baseline+DefinitionEMB with various α on the SST validation set.

Baseline \ α	3	5	10	20
RoBERTa	77.3	**79.1**	76.2	76.5
BART	**85.9**	**85.9**	84.8	84.1

Table 15: Accuracy for Baseline+DefinitionEMB with various α on the RTE validation set.

Baseline \ α	3	5	7	10
RoBERTa	**90.7**	90.5	90.3	90.3
BART	**91.8**	91.7	91.5	91.3

Table 16: Spearman's rank correlation for Baseline+DefinitionEMB with various α on the STS validation set.

Baseline \ α	3	7	10	30
RoBERTa	**92.9**	**92.9**	92.8	92.5
BART	**94.8**	94.7	94.5	94.4

Table 17: Accuracy for Baseline+DefinitionEMB with various α on the QNLI validation set.

Baseline \ α	3	5	10	30
RoBERTa	**91.8**	91.7	**91.8**	**91.8**
BART	92.4	**92.6**	92.5	92.2

Table 18: Accuracy for Baseline+DefinitionEMB with various α on the QQP validation set.

Baseline \ α	3	5	10	30
RoBERTa	**87.6**	87.5	87.4	87.3
BART	89.7	89.6	**89.8**	89.6

Table 19: Accuracy for Baseline+DefinitionEMB with various α on the MNLI validation set.

α	ROUGE (F1) 1	2	L
10	44.45	**21.62**	41.17
30	44.30	21.44	41.04
50	44.23	21.37	40.99
100	**44.62**	21.54	**41.40**

Table 20: ROUGE scores for BART+DefinitionEMB with various α on the CNNDM validation set.

α	ROUGE (F1) 1	2	L
5	44.02	20.66	34.95
10	**44.22**	**20.95**	**35.24**
20	43.81	20.55	34.78
100	42.46	19.18	33.62

Table 21: ROUGE scores for BART+DefinitionEMB with various α on the XSUM validation set.

α	ROUGE (F1) 1	2	L
5	50.63	32.19	38.81
7	50.85	**32.44**	**39.10**
10	**51.08**	32.17	38.97
100	50.04	31.43	38.27

Table 22: ROUGE scores for BART+DefinitionEMB with various α on the Billsum validation set.

α	ROUGE (F1) 1	2	L
10	43.62	18.53	37.43
20	43.93	18.84	37.74
30	**44.22**	**19.12**	**38.03**
100	42.96	17.76	36.73

Table 23: ROUGE scores for BART+DefinitionEMB with various α on the Y-BIGPATENT validation set.

Model	**MRPC**	**SST**	**RTE**	**STS**	**QNLI**	**QQP**	**MNLI**
RoBERTa	5	1	5	3	3	3	3
BART	5	1	3	3	3	5	10

Table 24: Tuned α for GLUE datasets.

Model	**CNNDM**	**Y-BIGPATENT**	**XSUM**	**Billsum**
BART	100	30	10	7

Table 25: Tuned α for text summarization datasets.

Model	Spearman Score ↑				
	RG65	RW	SimLex	SimVerb	Ave
RoBERTa	25.55	22.33	**18.04**	**10.78**	19.18
+ DefinitionEMB	**29.56**	**22.82**	17.48	10.58	**20.11**
BART	**24.35**	23.20	**22.00**	**12.73**	**20.57**
+ DefinitionEMB	22.93	**23.40**	21.04	12.09	19.87

Table 26: Experimental results on the word similarity task with cosine similarity. DefinitionEMB replaces $\mathbf{E}$ completely.

I Projected Initial Token Embeddings

Figure 17 shows the projected token embeddings of models before and after replacing $\mathbf{E}$ completely.

J Word Similarity Task Using Cosine Similarity

Table 26 shows the results for word similarity tasks, where the similarity between word embeddings is calculated using cosine similarity. Comparing Table 26 with Table 2, we observe that when using dot product, DefinitionEMB achieves a higher Spearman score than BART. However, when using cosine similarity, the opposite result is observed, which indicates the constructed embeddings for BART prioritize distance over angle aspects from the pre-trained embeddings.

K Isotropy on GLUE Task

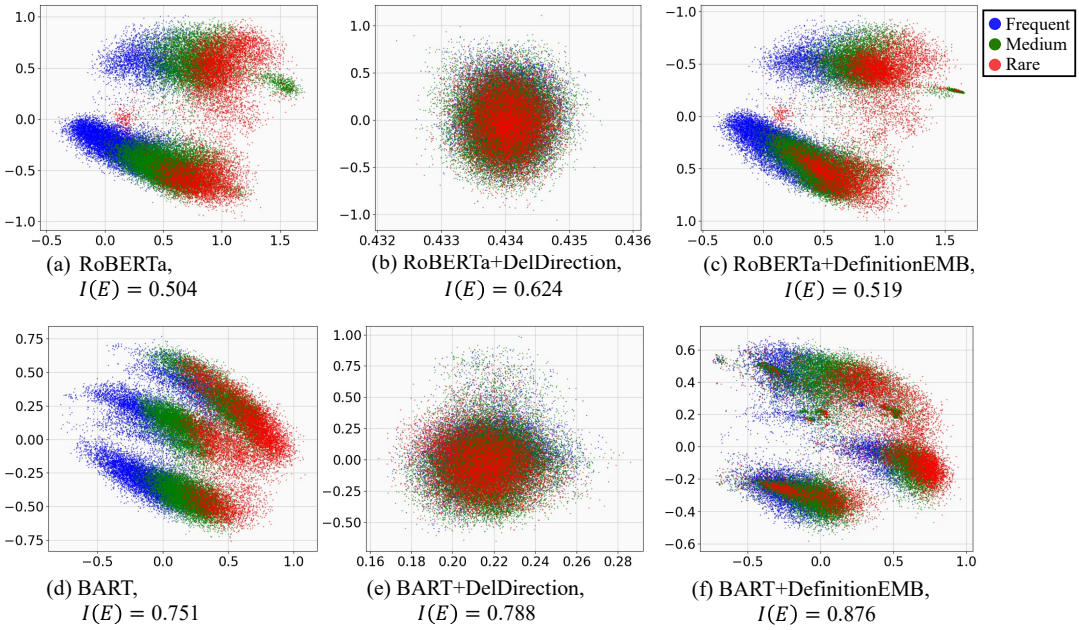
Table 27 presents $I(\mathbf{E})$ for models before and after fine-tuning on the GLUE task. Because $I(\mathbf{E})$ of PLMs and the DelDirection model before fine-tuning does not depend on $\mathcal{V}_{[task]}$, it is reported only once in the table. For the MRPC, STS, and RTE datasets, $I(\mathbf{E})$ shows a minimal difference between before and after fine-tuning models, likely due to the limited number of fine-tuning steps on these datasets. The token embedding distribution in BART appears to be more stable than RoBERTa on the SST, QQP, and MNLI datasets. Using DelDirection for RoBERTa and BART achieves the highest $I(\mathbf{E})$ across all datasets; however, it also results in the lowest accuracy and Pearson/Spearman's rank correlation in most cases, as shown in Table 3. This supports our assumption that DelDirection model focuses on the distribution of embeddings at the expense of semantic information.

L Ablation Study on MRPC

Table 28 shows the ablation study with respect to the replacing strategies for Baseline+DefinitionEMB on the MRPC test set. When replacing only appeared tokens, both RoBERTa and BART yield higher accuracy scores.

M Sample of Summarization

Tables 29 and 30 show sample summarizations of CNNDM test set.



(a) RoBERTa, $I(E) = 0.504$
(b) RoBERTa+DelDirection, $I(E) = 0.624$
(c) RoBERTa+DefinitionEMB, $I(E) = 0.519$
(d) BART, $I(E) = 0.751$
(e) BART+DelDirection, $I(E) = 0.788$
(f) BART+DefinitionEMB, $I(E) = 0.876$

Figure 17: Projected token embeddings of models before and after replacing $\mathbf{E}$ completely. The frequent (30%), medium (50%), and rare (20%) groups are determined based on the token index in $\mathcal{V}$.

Model	**SST**		**MRPC**		**STS**		**QQP**		**MNLI**		**QNLI**		**RTE**	
	Before	After	Before	After	Before	After	Before	After	Before	After	Before	After	Before	After
RoBERTa	0.504	0.533	-	0.505	-	0.506	-	0.542	-	0.544	-	0.509	-	0.505
+DelDirection	0.624	0.627	-	0.624	-	0.625	-	0.642	-	0.639	-	0.629	-	0.625
+DefinitionEMB	0.528	0.536	0.529	0.529	0.529	0.530	0.530	0.550	0.530	0.554	0.539	0.541	0.529	0.533
BART	0.751	0.751	-	0.751	-	0.751	-	0.752	-	0.751	-	0.751	-	0.751
+DelDirection	0.788	0.788	-	0.788	-	0.788	-	0.800	-	0.805	-	0.794	-	0.788
+DefinitionEMB	0.766	0.766	0.767	0.767	0.769	0.769	0.769	0.777	0.770	0.771	0.753	0.753	0.753	0.753

Table 27: $I(\mathbf{E})$ for models before and after fine-tuning on the GLUE task.

Model	**Replaced tokens**	**MRPC**
RoBERTa	Appearing	**87.7**
	Both	87.3
BART	Appearing	**88.3**
	Both	88.1

Table 28: Replacing appearing tokens only vs. replacing both appearing and non appearing tokens for RoBERTa and BART on the MRPC test set with $X = 24,900$.

Source	(CNN)For superhero fans, the cup runneth over. Most of us know the members of the Avengers by now: Iron Man, Captain America, Hulk and the rest, and the fact that a few more like Quicksilver are joining the cast **in the "Avengers: Age of Ultron" sequel**. But there was one character who remained a mystery: **the Vision, to be played by Paul Bettany.** Thus far, we've only seen his eyes in a trailer. With less than a month to go before the movie hits theaters, **Marvel Studios put all the speculation to rest with a poster featuring Bettany as the heroic android**, who was a member of the superhero group for many years in the comics. Meanwhile, as many Marvel fans know, **Thursday was the eve of the new Netflix series "Daredevil," and after a photoshopped first look at Charlie Cox's iconic red Daredevil suit went out, Marvel put out a video of the real one.** Not to be outdone, **director Bryan Singer announced a new character for next year's sequel "X-Men: Apocalypse," by telling Empire magazine that Ben Hardy would be playing the role of the winged mutant Angel.** He even had a photo to share. And Thursday's new super images weren't quite done, because the questions over how **Jamie Bell's rocky character The Thing in the rebooted "Fantastic Four" movie (out August 7) might look were also finally answered**. And he looks ... pretty much like The Thing we already knew (but reportedly, CGI this time). Within 24 hours, we got yet another indication that the superhero trend isn't going anywhere anytime soon (and we didn't even talk about the new photo of Ryan Reynolds' "Deadpool").
Reference	Marvel Studios releases first looks at Paul Bettany as the Vision in "Avengers: Age of Ultron" and Charlie Cox in full "Daredevil" costume . Jamie Bell's character of The Thing was also unveiled for 20th Century Fox's Marvel-based reboot of "Fantastic Four" Bryan Singer unveiled the first look at "X-Men: Apocalypse" Angel played by Ben Hardy .
BART	Paul Bettany will play the Vision in the "Avengers: Age of Ultron" sequel . The actor has been playing the android for many years in the comics . The "Fantastic For" reboot's" The Thing" looks pretty much like The Thing we already knew .
+DelDirection	Paul Bettany's character in "Avengers: Age of Ultron" is finally revealed . The actor has been playing the Vision in the comics for many years . The "Fantastic Fou" reboot's" The Thing" looks pretty much like The Thing we already knew (but CGI)
+DefinitionEMB	Paul Bettany will play the Vision in the "Avengers: Age of Ultron" sequel . Marvel Studios also announced a new character for "X-Men: Apocalypse" Ben Hardy will play the winged mutant Angel in "X-Men: Apocalypse," director Bryan Singer said .

Table 29: Sample summarization of CNNDM test set. **Bold** in source indicates the reference-related text. Underline in reference and model outputs indicates the rare token with index larger than 40,000 in $\mathcal{V}$.

Source	(CNN)It would have made Thomas Jefferson proud. Established on the birthday of the American founding father, Liberland – the world's newest micronation – is founded on a firm belief in liberty and noninterference from the powers-that-be. A tiny, 7 square-kilometer parcel of land, marked on maps as Gornja Siga, its territory abuts the Danube **on the border between Serbia and Croatia.** The victim of a border dispute between Serbia and Croatia, it is claimed by neither side – effectively a no-man's land. No one lives on this patch of land, which is heavily forested and contains only badly-maintained access roads and a run-down house, abandoned for decades. **This is where Euroskeptic Czech politician Vit Jedlicka stepped in. On April 13 he planted his newly-designed yellow and black flag in the territory, declaring the area the Free Republic of Liberland** – a tiny sliver of a country, bigger only than the Vatican and Monaco. **He tells CNN that the country will be formally founded on May 1** and is inviting, through the media, the world's heads of state to attend a formal ceremony marking the presumptive nation's birth. He says that he will also invite 7,500 of the 300,000 applicants that applied to become citizens of Liberland to the ceremony, where he will grant them citizenship. "I will grant citizenship if they can make it to the party," he told CNN by phone. "It's short notice but a good challenge, and also for the presidents (and other heads of state) if they can make it to the founding of our country." Jedlicka, an active member of the Czech Republic's Party of Free Citizens, opposes excessive government interference. He says his attempts to enact change in his home country led him to the political experiment that is Liberland. "I would describe it as a global revolution. It's just the beginning," he tells CNN via Skype. Founded on staunchly libertarian principles – its motto is "To live and let live" – its website describes its system of governance as being a "constitutional republic with elements of direct democracy." It will use a form of cryptocurrency – similar to Bitcoin – as its national currency, bypassing the need for a central bank and will, according to its constitution, keep government's noses out of everything possible, from the banks to prostitution. "Liberland prides itself on personal and economic freedom of its people, which is guaranteed by the Constitution, which significantly limits the power of politicians so they could not interfere too much in the freedoms of the Liberland nation," the world's newest constitutional document states. Financial regulation will be minimal, if at all present. **Jedlicka says almost 300,000 applications for citizenship have been received**, about 20 of which have been accepted. "Thousands of Americans, Swiss people. Also a lot of Arabic peoples who feel oppressed by the regimes there." He envisions, ultimately, a community of around 35,000 Liberlanders, not all of whom will be full-time residents. He says he expects trouble from his neighbors, whose land he has effectively annexed. "From Serbia, Croatia, we expect some trouble but we expect international laws will applied and any movement against us would be an attack on a sovereign nation, and we will offer nothing but passive resistance. For now, (though) we will make roads, docks." For its part, the Serbian Ministry of Foreign Affairs released a statement to CNN that stated that Liberland would not theoretically impinge upon its border, which is delineated by the Danube, but "the Ministry also considers this a frivolous act which needs no further comment." Croatia's counterpart was similarly dismissive. "Virtual quips, however interesting they occasionally sound, remain what they are – virtual quips, and for them we have no official comment."
Reference	Vit Jedlicka, the first president of Liberland, tells CNN that the country will be formally founded on May 1 . On April 13, Jedlicka declared an area between Croatia and Serbia "the Free Republic of Liberland" Jedlicka says that almost 300,000 applications for citizenship have so far been received .
BART	Liberland is a tiny, 7 square-kilometer parcel of land, marked on maps as Gornja Siga, which abuts the Danube on the border between Serbia and Croatia . The land is claimed by neither Serbia or Croatia, and is effectively a no-man's land . Czech politician Vit Jedlicka planted his newly-designed yellow and black flag in the territory, declaring the area the Free Republic of Liberland . He says he expects trouble from his neighbors, whose land he has effectively annexed .
+DelDirection	Czech politician Vit Jedlicka has declared the Free Republic of Liberland, a tiny sliver of land on the border of Serbia and Croatia . He says he expects trouble from his neighbors, whose land he has effectively annexed . "I would describe it as a global revolution. It's just the beginning," he tells CNN .
+DefinitionEMB	Czech politician Vit Jedlicka has declared the Free Republic of Liberland, a tiny sliver of land on the Serbian-Croatian border . He says he will grant citizenship to 7,500 of the 300,000 applicants who applied to become citizens of Liberland . "I would describe it as a global revolution. It's just the beginning," says Jedlicka via Skype .

Table 30: Sample summarization of CNNDM test set. **Bold** in source indicates the reference-related text. Underline in reference and model outputs indicates the numeric in $\mathcal{V}$.

Token	Index in $\mathcal{V}$	MSE
sys	43103	23.3829
ĠNASL	47179	23.2779
resso	27989	23.2549
ĠFAQ	39313	22.8146
ĠpH	39228	22.1278
ĠB	163	0.0061
ER	2076	0.0059
s	29	0.0055
ING	1862	0.0051
-	12	0.0040

Table 31: Top and bottom 5 tokens based on the degree of MSE estimated by DefinitionEMB based on RoBERTa model, listed in descending order of MSE.

Token	Index in $\mathcal{V}$	MSE
ourke	18338	18.7777
esson	24711	17.6324
aeus	39174	17.3797
wagen	42099	16.8839
auga	24491	16.8624
ER	2076	0.0020
ES	1723	0.0020
ING	1862	0.0020
-	12	0.0018
S	104	0.0018

Table 32: Top and bottom 5 tokens based on the degree of MSE estimated by DefinitionEMB based on BART model, listed in descending order of MSE.

N MSE between Pre-trained and Definition Embeddings

For tokens in $\mathcal{V}$, we analyze the MSE between their pre-trained and definition embeddings. Figure 18 presents the results for DefinitionEMB on the RoBERTa model. The left subfigure illustrates that around 20% of tokens have an MSE of less than 1, while less than 20% tokens have an MSE larger than 8. The right subfigure shows that the distribution of token index is almost uniform across the MSE, indicating that the pre-trained embedding of high-frequency tokens may contain semantically unrelated information, while the pre-trained embedding of low-frequency tokens may contain semantically related information even with limited pre-training steps. In addition, more tokens falling in the MSE range of [5, 8) than in the range of [0, 1), which indicates a significant difference between pre-trained embeddings and definition embeddings.

Figure 19 presents the results for DefinitionEMB on the BART model. The left subfigure illustrates that around 40% of tokens have an MSE of less than 1, while less than 10% tokens have an MSE larger than 8. The right subfigure also shows that the distribution of token index is almost uniform across the MSE. However, most of the tokens fall in the MSE range [0, 1), indicating less difference between pre-trained embeddings and definition embeddings than in the case of RoBERTa.

Tables 31 and 32 lists examples of tokens with corresponding MSE. The top 5 tokens with the highest MSE can be used as named entities.

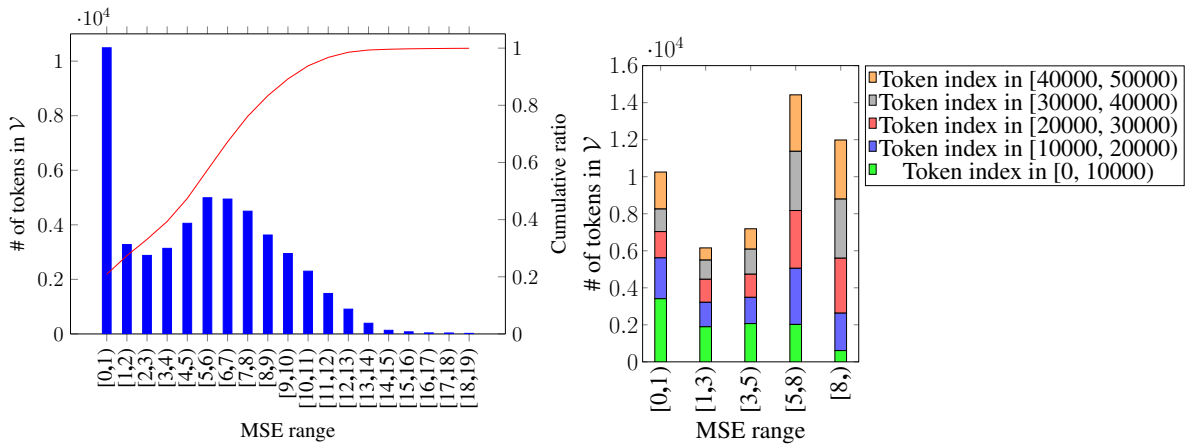


Figure 18: Number of tokens in $\mathcal{V}$ versus MSE estimated by DefinitionEMB based on RoBERTA model.

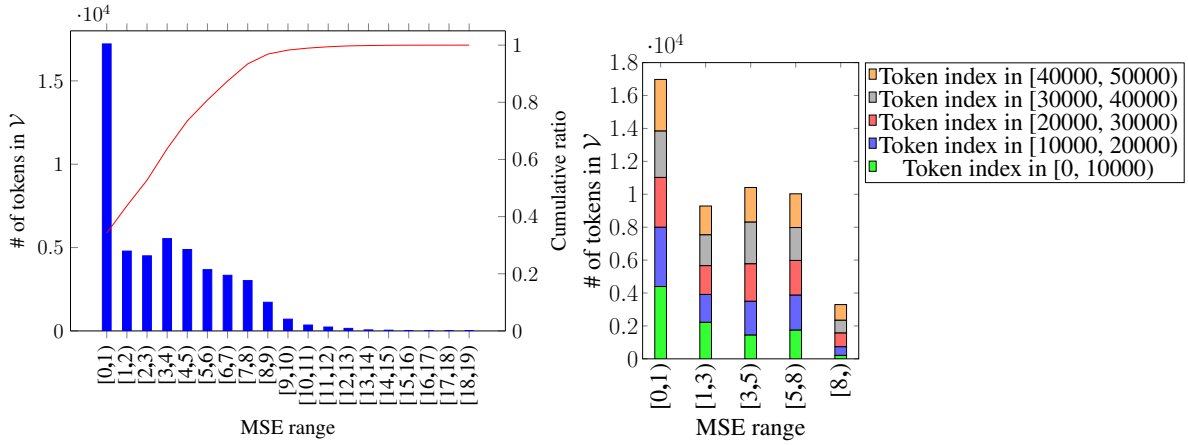


Figure 19: Number of tokens in $\mathcal{V}$ versus MSE estimated by DefinitionEMB based on BART model.