Low Anisotropy Sense Retrofitting (LASeR) : Towards Isotropic and Sense Enriched Representations

Geetanjali Bihani and **Julia Taylor Rayz**
Department of Computer and Information Technology
Purdue University
{gbihani,jtaylor1}@purdue.edu

Abstract

Contextual word representation models have shown massive improvements on a multitude of NLP tasks, yet their word sense disambiguation capabilities remain poorly explained. To address this gap, we assess whether contextual word representations extracted from deep pretrained language models create distinguishable representations for different senses of a given word. We analyze the representation geometry and find that most layers of deep pretrained language models create highly anisotropic representations, pointing towards the existence of representation degeneration problem in contextual word representations. After accounting for anisotropy, our study further reveals that there is variability in sense learning capabilities across different language models. Finally, we propose **LASeR**, a 'Low Anisotropy Sense Retrofitting' approach that renders off-the-shelf representations isotropic and semantically more meaningful, resolving the representation degeneration problem as a post-processing step, and conducting sense-enrichment of contextualized representations extracted from deep neural language models.

1 Introduction

Distributional word representations, developed using large-scale training corpora, form an integral part of the modern NLP methodological paradigm. The advent of deep pre-trained neural language models such as BERT (Devlin et al., 2018) and GPT-2 (Radford et al., 2019) has led the shift towards the development of contextualized word representations. Unlike static word representation models, such as *word2vec* (Mikolov et al., 2013) and *fastText* (Bojanowski et al., 2017), which conflate multiple senses of a word within a single representation, contextual word representation models assign as many representations to a word as the number of contexts it appears in. The preference for contextual word representations can be attributed to the significant improvements they have achieved in a wide variety of NLP tasks including question answering, textual entailment, sentiment analysis (Peters et al., 2018; Devlin et al., 2018) and commonsense reasoning (Da and Kasai, 2019; Sap et al., 2020), to name a few.

To utilize contextual word representations as knowledge resources, it is necessary to determine their ability to mirror the linguistic relations employed in language (Schnabel et al., 2015). There is a growing body of literature that assesses whether contextual representations encode information about word-senses, where each word-sense portrays an aspect of the meaning of a given word in a given context (Jurafsky and Martin, 2019). A recent analysis by Nair et al. (2020) reported that contextual word representations can learn human-like word sense knowledge, where they compared cosine relatedness between homonyms and polysemous word senses against human sense-related judgements. When calculating cosine relatedness, such studies assume the encoded vector space to be isotropic in nature. Geometrically, isotropy in a vector space is defined as vectors being uniformly distributed across all directions, instead of occupying a narrow cone (Ethayarajh, 2019; Mu and Viswanath, 2018). Recent studies point towards anisotropy (lack of isotropy) in contextual word representations (Ethayarajh, 2019; Zhang et al., 2020), which affects prior conclusions regarding word-sense information encoded in vector spaces. For example, in an isotropic vector space, if cosine relatedness between word representations A and B is 0.9, we conclude them to be highly similar. But, if the vector space is anisotropic, where cosine relatedness between randomly sampled words is 0.95, then the representations A and B are deemed less similar than randomly sampled words. This shows that the existence and the extent of anisotropy in the vector space affects conclusions regarding whether representations are actually similar or merely a

Proceedings of Deep Learning Inside Out (DeeLIO):
The 2nd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures, pages 81–95
Online, June 10, 2021. ©2021 Association for Computational Linguistics

product of representation degeneration. Hence, when evaluating the sense learning capabilities of deep pretrained language models through vector relatedness measures, accounting and adjusting for vector space anisotropy becomes necessary.

In this regard, our work presents three key contributions. First, we analyze and adjust for anisotropy across contextual representations extracted from all layers of four language models (BERT, GPT-2, XLNet and ELECTRA). The representation space for each model encodes anisotropy, varying in terms of number and strength of common directions in model representations. We find that models learning unidirectional context create more anisotropic representations than models learning bidirectional context. Second, we observe that sense information is not equally encoded in all models, where (pseudo) bidirectional models learn to disambiguate word senses better than others. Moreover, sense information is better retained in the lower layers and significantly reduces in the upper model layers due to the representations getting more contextualized. Third, to address these preliminary findings and to contribute towards the creation of sense-coherent representations, we propose **LASeR**, a 'Low Anisotropy Sense Retrofitting' approach, bringing word representations closer to the goal of mirroring lexical semantic relations present in natural language while removing artifacts of representation degeneration from learned representations. Thus, we combine vector space transformation and knowledge-based vector specialization methods to create more isotropic and sense enriched representations, ensuring that we retain the distributional properties learnt during pretraining, while aligning and grounding the representation geometry towards better sense learning.

2 Related Work

Prior works which modify off-the-shelf embeddings to improve their lexical-semantic representation can be divided into two primary categories: (1) Anisotropy treatment methods and (2) Retrofitting methods. Anisotropy treatment methods focus on improving the isotropy of word vectors, promoting uniform distribution of information across all directions (Mu and Viswanath, 2018; Raunak et al., 2019; Wang et al., 2019). Isotropy in contextual vector spaces is regarded valuable, especially when utilizing vector geometry and relatedness measures in downstream analyses (Ethayarajh, 2019). Prior methods that focus on creating more isotropic vector spaces have suggested principle component manipulation (removal, extension) of vector spaces (Mu and Viswanath, 2018; Jo and Choi, 2018). To our knowledge, these methods have been proposed for static word representations, but are yet to be extended to contextual word representations extracted from a wide variety of language models.

On the other hand, retrofitting methods are focused on enhancing the representation geometry, by encoding lexical semantic relations through semantic specialization, a post-processing approach that enforces linguistic constraints on vector spaces by relying on external linguistic knowledge databases (Vulić and Mrkšić, 2018; Faruqui et al., 2015; Jo and Choi, 2018; Vulić, 2018). Semantic specialization as a post processing step (retrofitting) is currently limited to static word representations (Mu and Viswanath, 2018; Vulić and Mrkšić, 2018) where they have yielded impressive performance improvements over raw embeddings (Lauscher et al., 2020a). Existing methods towards semantic specialization of contextual representations primarily focus on retraining the model from scratch (Lauscher et al., 2020b) or post-hoc fine-tuning the model (Zhang et al., 2019; Peters et al., 2019; Wang et al., 2020). These methods are (1) resource-intensive (retraining or fine-tuning) and (2) do not address the representation degeneration problem in vector representations (Gao et al., 2018).

3 Methodology

3.1 Contextual Word Representation Models

In this work, we focus on contextual word representations generated from four transformer-based model architectures, i.e., BERT (Devlin et al., 2018), GPT-2 (Radford et al., 2019), XLNet (Yang et al., 2019) and ELECTRA (Clark et al., 2019). These models have been selected to assess the impact of variation in context learning and pretraining over the quality of generated representations, while keeping the number of hidden layers and dimensionality identical (*layers* = 13 $(0+12)$*; dimensions* = 768). BERT and ELECTRA are both bidirectional learners, but they differ in terms of the pre-training objectives used to train the models: BERT uses a masked language modeling objective, limiting its learning to a small subset of word tokens; ELECTRA uses replaced token detection and is able to learn across a wider range of words tokens. On the other hand, GPT-2 and XLNet are both unidirectional learners, where GPT-2 learns only left-to-right context, while XLNet learns over all possible permutations of the given input. A comparison over these models in a uniform setting allows us to relate the behavior of representations to the context learning and pre-training choices of the respective models.

3.2 Data

Contextual word representations for individual words are generated by feeding sentences into the language model. In order to generate representations, we use sense annotated corpora from various SemEval and SenseEval tasks, including SensEval 3 task 1 (`S3-T1`) (Snyder and Palmer, 2004), SensEval 2 all-words task (`S2-TA`) (Edmonds and Cotton, 2001), SemEval 2013 task 12 (`S13-T12`) (Navigli et al., 2013), SemEval 2007 task 7 (`S7-T7`) (Navigli et al., 2007) and SemEval 2015 task 13 (`S15-T13`) (Moro and Navigli, 2015). To ensure that the Wordnet sense keys are unified across corpora, we utilize the Wordnet 3.0 sense annotated data (Vial et al., 2018) and summarized in Table 1. Since we want to evaluate sense-learning, we limit our analyses to multi-sense words, retaining nouns, adjectives and verbs that appear within the corpora as more than one sense.

3.3 Sense Learning Measures

In order to compute how sense information is encoded with the word representations, we define two word-sense specific cosine relatedness measures.

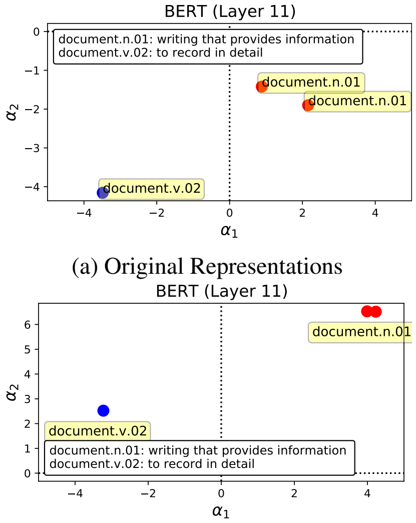
Definition 1 (Sense Similarity)**.** Let w_s be a sense of the word w, appearing in m different contexts. Let v_l be the vector that maps the each word sense occurrence w_{s_i} to the vector space. Then, the average sense similarity between all m instances of the word sense w_s for layer ℓ is

$$SenSim_\ell(w_s) = \frac{1}{m} \sum_j \sum_{k \neq j} \cos(v_\ell(w_{s_j}), v_\ell(w_{s_k})) \quad (1)$$

This metric calculates the average cosine similarity between contextual representations of the same sense of a word.

Corpus	Nouns	Verbs	Adjectives
`S3-T1`	428	635	166
`S2-TA`	292	307	344
`S13-T12`	338	164	46
`S7-T7`	512	814	380
`S15-T13`	110	156	127
Total	1680	2076	1063

Table 1: Data Summary.



(a) Original Representations

(b) LASeR Representations

Figure 1: Representations of different senses of the word ‘document’ (BERT Layer 11).

Definition 2 (Inter Sense Similarity)**.** Let the word w have S different word senses, where w_a and w_b are a pair of different senses of w, appearing in m and n different contexts respectively and $a, b \in S$. Let v_l be the vector that maps each word sense occurrence w_{s_i} to the vector space. Then, the average inter sense similarity between the representations of all instances of the word w for layer ℓ is

$$InterSim_\ell(w) = \mathbb{E}_{a,b \in S}\left[\frac{1}{mn} \sum_{j=1}^{m} \sum_{i=1}^{n} \cos(v_\ell(w_{a_i}), v_\ell(w_{b_j}))\right] \quad (2)$$

This metric calculates the average cosine similarity between contextual representations of different senses of a word.

Thus, if a word w has $SenSim_l(w_s) > InterSim_l(w)$, it suggests that the representations for the same sense of a given word lie much closer together within the vector space, as compared to the representations of different senses of the same word. For example, a given word *‘document’* can refer to multiple senses. According to WordNet 3.0, two senses of the word *‘document’* are: (1) *document.n.01* - writing that provides information and (2) *document.v.02* - to record in detail.

As an example, we have visualized the representations of these two senses as encoded within the vector space of BERT (Layer 11), shown in Figure 1. The ‘original’ representations, shown in Figure 1(a), of the word sense *document.n.01* lie slightly close to each other, and farther away from the *document.v.02* representation. Thus, if a model is able to encode similar representations for same sense of a word, and distinguishable representations for different senses of a word, we claim that the model encodes sense information.

3.4 Anisotropy Adjusted Sense Similarity

In order to assess whether contextual word representations encode sense information, we measure the sense similarity and inter sense similarity for multi-sense words (polysemes and homonyms) in our datasets, across model layers. Given that contextual word representations encode anisotropy, we calculate anisotropy adjusted sense relatedness measures as follows.

$$B(v_\ell){=}\mathbb{E}_{a,b\sim U}[\cos(v_\ell(a){,}v_\ell(b))] \tag{3}$$

$$SenSim_\ell(w_s)^*{=}SenSim_\ell(w_s){-}B(v_\ell) \tag{4}$$

$$InterSim_\ell(w)^*{=}InterSim_\ell(w){-}B(v_\ell) \tag{5}$$

This baseline calculation utilizes the theory from prior works examining contextualization in word representations (Ethayarajh, 2019). Here, $B(v_\ell)$ is the average cosine similarlity between n randomly sampled words, U is the set of all word occurrences, and $v_\ell(.)$ maps a word occurrence to the respective word representation in layer ℓ.

3.5 Low Anisotropy Sense Retrofitting

In this subsection, we describe LASeR, a post-processing approach to render off-the-shelf representations more isotropic and sense-enriched. Our approach builds upon the work on anisotropy reduction Mu and Viswanath (2018) and retrofitting Faruqui et al. (2015). Mu and Viswanath (2018) suggests that anisotropy can be reduced by removing primary components to make the representations more distinct and uniformly distributed within the vector space. We extend this to contextual word representations, evaluating the efficacy of removing primary components on anisotropy reduction in contextual representations. Turning towards retrofitting methods, we extend the retrofitting approach proposed by Faruqui et al. (2015), which targets static word representations and brings synonyms closer together in the vector space. Our work extends this retrofitting goal to contextual representations, where we aim to bring representations of same word-senses closer in the vector space, ensuring better sense disambiguation capabilities for representations.

Algorithm 1: LASeR (Low Anisotropy Sense Retrofitting).

```
Input: Raw word representation
       ${v(w_i), w_i \in V}$
1 Perform mean centering of vector:
    $\mu \leftarrow \frac{1}{|v|}\Sigma_{w_i \in V} v(w_i); \tilde{v}(w_i) \leftarrow v(w_i) - \mu$
2 Compute the PCA components:
    $u_{i1}, \ldots, u_{iD} \leftarrow PCA(\{\tilde{v}(w_i), w_i \in \mathcal{V}\})$
3 Remove top $d$ principal components:
    $v'(w_i) \leftarrow \tilde{v}(w_i) - \Sigma_{j=1}^{d} \left(u_{ij}^{\top} v(w_i)\right) u_{ij}$
4 Apply retrofitting update:
    $\hat{v}(w_i) = \frac{\sum_{j:(i,j)\in E} \beta_{ij} v(w_j) + \alpha_i v(w_i)}{\sum_{j:(i,j)\in E} \beta_{ij} + \alpha_i}$
Output: Processed word representation
        $\hat{v}(w_i)$
```

Let $v(w_i)$ be the original contextual representation, $v'(w_i)$ be the *low anisotropy* contextual representation and $\hat{v}(w_i)$ be the *sense enriched* contextual representation of i^{th} occurrence of a word sense w. We simulate an undirected knowledge graph $\Omega(V, E)$, where V represents the vocabulary of word tokens, each word token representing a vertex, and E represents all the edges connecting respective vertices. Finally, Q represent the matrix of post-processed representations $[\hat{v}(w_1), \hat{v}(w_2), \ldots, \hat{v}(w_n)]$. The approach works on achieving dual objectives, described as follows: **Objective 1 (Lower Anisotropy) :** Remove top d common directions across all $v(w_i)$, to create $v'(w_i)$, creating more uniformly distributed word vectors and lowering anisotropy in representations. **Objective 2 (Sense Retrofitting) :** Learn $\hat{v}(w_i)$ such that same sense representations lie closer together in vector space as well as close to the original embedding

The algorithm takes the original representations as input. These representations undergo mean centering and removal of dominant primary components (1,2,3) to reduce the anisotropy in the vector space. This is followed by a sense-retrofitting update (4) . Here, for each word token representation $v(w_i)$, we define its neighbours as $v(w_j), \forall j$ where $sense(w_i) = sense(w_j)$, and hyper-parameters β_{ij} and $\alpha_i = 1$ represent the reciprocal of the node degree of the word token w_i and edge weights respectively.

4 Results

We first show anisotropy analysis results (§4.1), further evaluating sense learning in contextual representations (§4.2). Finally, we present improvements in isotropy and lexical-semantic capabilities of the post-processed representations (§4.3).

4.1 Anisotropy Analysis

4.1.1 Similarity between Random Words

We first assess the amount of anisotropy encoded within contextual word vector spaces. We plot the average cosine similarity between 1K randomly sampled words, across different layers of language models, as seen in Figure 2. If a vector space is isotropic, the average cosine similarity between uniformly randomly sampled words would be 0 (Ethayarajh, 2019). Thus, the closer this measure is to 1, the more anisotropic the vector space. It can be seen that anisotropy evolves very differently across different models. Unidirectional language models (XLNet, GPT-2) portray far more anisotropy in word representations as compared to bidirectional language models (BERT, ELECTRA). Thus, language models learning one-directional context (L-to-R or R-to-L) encode more common directions in the representations as compared to those learnt from bidirectional context. Moreover, anisotropy monotonically increases across layers for BERT and XLNet, where both models have been trained on masked language modeling tasks. This shows that anisotropy accumulates in the upper layers of masked language models. The rate of increase in anisotropy in XLNet is higher than BERT representations, showing that permutation language modeling propagates higher amounts of anisotropy than traditional MLM. These results are consistent with the results obtained for all multi-sense words in the corpora (Appendix A).

4.1.2 Analysis of Principal Components

High anisotropy leads to word vectors being distributed within a very narrow cone in the vector space (Mimno and Thompson, 2017), further signifying that the word representations encode common directions (Mu and Viswanath, 2018). We plot the top two dominating directions for word representations, across each model's layers, as shown in Figure 3. These plots reveal that contextual word representations extracted from different language models are encoded extremely differently within the vector space. It can be seen that BERT and XLNet embeddings are more spread across the vector space, as compared to GPT-2 and ELECTRA embeddings. Moreover, ELECTRA embeddings form highly concentrated, yet separated regions of anisotropy, thus leading to an overall low score on the average similarity between randomly sampled words. Moreover, GPT-2 embeddings reveal extreme anisotropy, where most of the embeddings encode a singular common direction. The plots in Figure 3 also reveal that word frequency is significantly encoded in the top two principal components of BERT and XLNet embeddings. We cannot claim the same for GPT-2 and ELECTRA embeddings, where all embeddings cluster within highly dense regions of anisotropy.

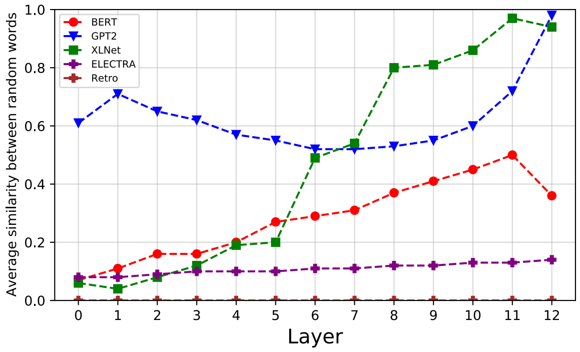


Figure 2: Average similarity between representations of randomly sampled words (1K) across model layers.

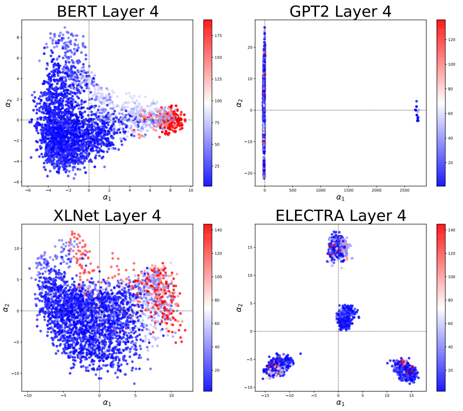
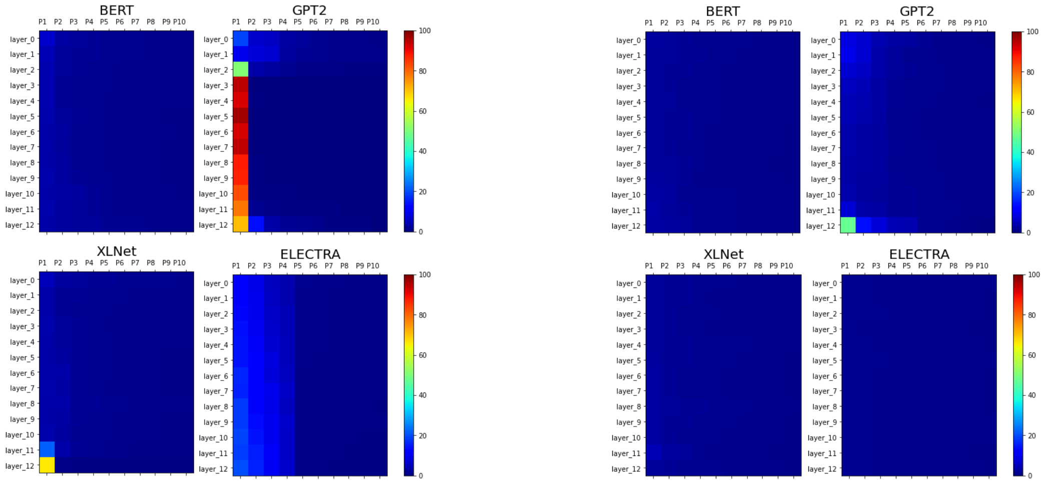


Figure 3: PCA plots of original word representations across top 2 primary components; Blue:Low frequency word tokens, Red:High frequency word tokens.

We also evaluate anisotropy across model layers by assessing the explained variance across common directions encoded across all word representations. We plot the proportion of variance encoded within the top $d = 10$ dominant principal components of the original contextual representations across model layers, shown in Figure 4(a). While bidirectional models such as BERT and ELECTRA encode multiple common directions, unidirectional models like GPT-2 and XLNet embeddings primarily encode a singular common direction. For BERT embeddings, the top 10 primary components only contribute to 17-24% of the explained variance, showing that the embeddings are more uniformly distributed across the vector space, as compared to other models. GPT-2 provides a stark contrast,



(a) Vanilla Representations.

(b) Retrofitted Representations.

Figure 4: Plots of proportion of variance encoded within the top $d = 10$ dominant principal components of the contextual representations across model layers. The horizontal labels (P1-P10) represent each of the ten principal components, and the vertical labels (layer_0 - layer_12) represent each of the 12 model layers.

where the top 10 principal components contribute to up to 97% of the explained variance, highly concentrated within the first principal component, especially for the middle layers (Layer 3-8). XLNet embeddings capture comparatively lower common directions across model layers, apart from the final model layer (Layer 12), where 66.1% of the explained variance is concentrated within the first principal component. Thus, representations learnt through the goal of predicting the next word yields all representations extremely similar.

4.2 Sense Learning in Original Representations

A model differentiates between different word senses if it encodes representations of the same sense of a word to be more similar than the representations of other senses of the same word. We utilize the sense learning measures, defined in (§3.3) to assess whether original representations encode word-sense information. To examine overall learning across model layers, we calculate average sense similarity ($\overline{SenSim_\ell(w)}$) and mean difference between average sense similarity and inter sense similarity for a word token w (Δ).

$$\Delta = \overline{SenSim_\ell(w)} - InterSim_\ell(w);\ \ \Delta \in [-1,1] \tag{6}$$

Ideally, a language model being able to capture distinction between all word senses should have $\overline{SenSim_\ell(w)} = 1$ and $\Delta >> 0$. Here, higher sense similarities correspond to similar senses being encoded closer in the vector space and $\Delta > 0$ shows that on an average, same sense representations are more cohesive and well separated from the representations of other senses.

The evolution of sense learning over different models and their layers is portrayed using sense similarity measures, aggregated in Table 2. The reported vanilla sense similarity scores have been adjusted for anisotropy. Prior to retrofitting, BERT and XLNet embeddings for the same word senses show increasing dissimilarity across model layers, signifying a loss of sense information as the model gets more contextualized. The similarity between same sense word representations from the GPT-2 model is close to 0, showing that GPT-2 captures almost no sense information within the embedding

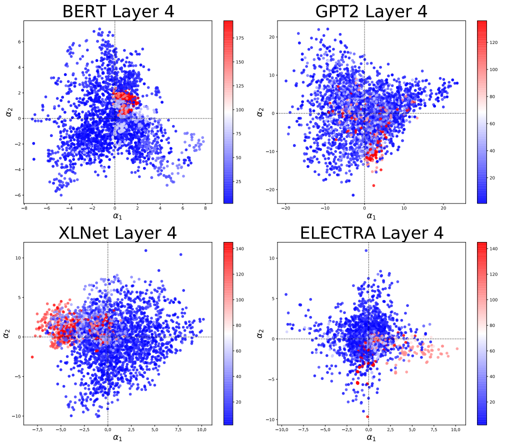


Figure 5: PCA plots of post-processed word representations across top two primary components, for each model.

space. ELECTRA embeddings remain consistent in terms of sense learning, not varying significantly across model layers. Furthermore, $\Delta \sim 0$ across all models shows that the original representations do not significantly distinguish between different senses of a given word. We visualize an example in Figure 1(a), where representations of the word *document* lie close together, regardless of the different senses associated with each occurrence. This finding signifies that the sole reliance on word form to learn representations does not suffice in helping the model distinguish between multiple senses of a given word.

4.3 Low Anisotropy Sense Retrofitting

We evaluate the efficacy of the proposed **LASeR** approach by comparing improvements in vector space isotropy and improved disambiguation of different word senses, as captured by retrofitted word representations.

4.3.1 Improvements in Isotropy

We conduct experiments by removing the most dominant common direction ($d = 1$) across generated embeddings across each model layer. This step yields significantly better isotropy in the resulting representations, where average similarity between randomly sampled words ($k = 1000$) is 0, across all models and model layers. Improvements can also be observed from the reduced proportions of explained variance in Figure 4(b). Overall, most of the anisotropy in the vector space is treated by removing one dominating direction. The retrofitted GPT-2 embeddings still show high anisotropy in the 12^{th} layer, showing that more common directions remain to be addressed and possibly removed. These results show that high anisotropy effects can be reduced by removing the primary common directions across representations. The effect of this step is also visualized in Figure 5, where the representations are significantly less anisotropic and more uniformly spread across the vector space, encoding fewer artifacts of word frequency in the vector space, as compared to the original representations. For visualizations across all model layers, refer to Appendix B. In most cases, removal of the most dominant common direction can yield significant improvements in isotropy, as seen for BERT, XLNet and ELECTRA. In other cases, where representations share more than one significant common directions, such as for GPT-2, we can remove $d > 1$ common directions to treat anisotropy.

4.3.2 Improvements in Sense Representation

The retrofitting update applied to model representations enforces lexical-semantic constraints, bringing same sense representations closer together (increase same-sense cohesion) and pushing different sense representations farther apart (increase inter-sense separation).

Results from Table 2 show the efficacy of our retrofitting update ($\alpha_i = 1$), where average sense similarity between word vectors increases significantly, and similarity between same sense representations is significantly higher than similarity between representations of different senses. This portrays that the retrofitted representations encode same sense representations closer together and different sense representations farther apart. An example of how retrofitting changes the distribution of representations in the vector space is given in Figure 1(b), where inter-sense separation between two different senses of the word *document* increases and same-sense cohesion between representations of the same word sense increases.

Across the model layers, the retrofitting significantly increases sense similarity and Δ. The improved similarity scores can be seen in Figure 6, which show that retrofitting moves same sense representations to be more similar than different sense representations. For BERT embeddings, the improvements are more visible in the upper model layers, as they create more separated different sense representations, and more cohesive same sense representations. The slight drop in cohesion ($SenseSim$) is due to the model's upper layers being more contextualized than the lower layers, also suggested in prior works on contextualization (Ethayarajh, 2019). Retrofitting is extremely effective for GPT-2 embeddings. This can been from the drastic increase in sense similarity ($SenseSim$) and Δ, showing that same sense representations lie closer and different sense representations lie farther apart in the retrofitted vector space.

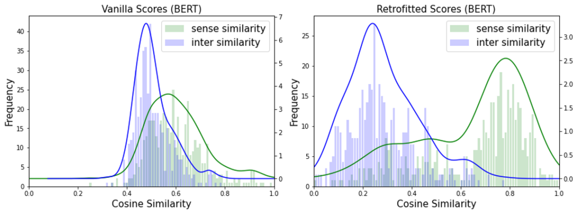


Figure 6: Effect of retrofitting on sense relatedness in contextual embeddings. Here, retrofitted embeddings portray higher same-sense similarity and lower inter-sense similarity.

Layer	BERT		GPT-2		XLNet		ELECTRA	
	vanilla (Δ)	*retro* (Δ)	*vanilla*(Δ)	*retro* (Δ)	*vanilla* (Δ)	*retro* (Δ)	*vanilla* (Δ)	*retro* (Δ)
0	0.82 (0.00)	0.93 (0.04)	0.08 (0.05)	0.49 (0.38)	0.08 (-0.27)	0.85 (-0.02)	0.49 (0.11)	0.64 (0.25)
1	0.74 (0.02)	0.90 (0.08)	0.06 (0.03)	0.51 (0.38)	0.80 (0.02)	0.90 (0.08)	0.50 (0.11)	0.65 (0.26)
2	0.66 (0.05)	0.88 (0.13)	0.06 (0.02)	0.51 (0.38)	0.65 (0.07)	0.83 (0.18)	0.50 (0.12)	0.65 (0.26)
3	0.62 (0.07)	0.85 (0.18)	0.08 (0.04)	0.52 (0.39)	0.58 (0.10)	0.80 (0.23)	0.50 (0.12)	0.65 (0.26)
4	0.54 (0.09)	0.82 (0.23)	0.09 (0.05)	0.53 (0.39)	0.46 (0.09)	0.76 (0.27)	0.51 (0.12)	0.65 (0.26)
5	0.43 (0.10)	0.77 (0.28)	0.09 (0.05)	0.53 (0.39)	0.40 (0.09)	0.72 (0.30)	0.52 (0.13)	0.65 (0.27)
6	0.36 (0.12)	0.72 (0.33)	0.11 (0.06)	0.54 (0.40)	0.23 (0.06)	0.68 (0.31)	0.51 (0.12)	0.65 (0.27)
7	0.31 (0.11)	0.68 (0.35)	0.12 (0.06)	0.55 (0.41)	0.20 (0.06)	0.66 (0.33)	0.52 (0.13)	0.65 (0.27)
8	0.25 (0.10)	0.65 (0.37)	0.11 (0.07)	0.55 (0.41)	0.08 (0.03)	0.64 (0.33)	0.53 (0.13)	0.64 (0.26)
9	0.22 (0.09)	0.63 (0.39)	0.11 (0.07)	0.56 (0.42)	0.07 (0.02)	0.63 (0.34)	0.53 (0.13)	0.64 (0.26)
10	0.22 (0.09)	0.63 (0.38)	0.09 (0.06)	0.56 (0.43)	0.05 (0.01)	0.65 (0.34)	0.53 (0.13)	0.64 (0.27)
11	0.20 (0.07)	0.63 (0.36)	0.08 (0.04)	0.55 (0.43)	0.01 (0.00)	0.65 (0.32)	0.53 (0.13)	0.64 (0.27)
12	0.24 (0.09)	0.62 (0.37)	0.00 (0.01)	0.51 (0.40)	0.02 (0.00)	0.63 (0.32)	0.53 (0.13)	0.64 (0.27)

Table 2: Average sense similarity scores across model layers; $\Delta = SenSim_\ell(w) - InterSim_\ell(w)$.

While originally, the representations were highly anisotropic and held no sense learning, the retrofitted embeddings capture better sense distinction. XLNet embeddings, much like BERT, encode representations of the same word form closer together, especially in lower model layers, regardless of the respective word-sense distinction. Post-retrofitting, XLNet embeddings show higher similarity between same word senses and lower similarity between different word senses, revealing better sense disambiguation. Compared to the other three models, original ELECTRA embeddings are able to capture more distinction between different sense representations and more similarity between same sense representations. Our retrofitting update further improves these lexical-semantic relations in the representation space.

5 Discussion

Recent works have discussed whether contextualized word representations extracted from deep pretrained language models encode word sense knowledge within the representation space. Studies suggest that while lower layer BERT embeddings encode more semantic information (Reif et al., 2019), the upper layer embeddings become increasingly contextual (Ethayarajh, 2019). Works exploring semantic capabilities of representations have also used nearest neighbour classifier probes to assess whether same-sense representations are classified together (Reif et al., 2019; Nair et al., 2020). Since these classifiers show slightly better accuracy than classifying as the most frequent sense, they claim that the representation space encodes sense information. Although our work supports this conclusion, we additionally argue that after accounting for anisotropy, the cohesion between same sense representations and separation between different sense representations is not significant. Here, the principal premise of the removal of anisotropy prior to injecting sense information is based on creating an embedding space geometry where the effects of representation degeneration are reduced. The representation degeneration of embeddings reduces their representational power (Gao et al., 2018). Thus, to improve the representation ability of embeddings, we deem it important to create methods that promote representations that are not only lexico-semantic relation enriched but also isotropic. Our method reveals that the additional step of lowering anisotropy renders improved representation geometry, where word vectors are not constricted within a narrow cone, and are uniformly distributed within the vector space. Further, sense-retrofitting on contextualized word representations render same sense representations more similar and different sense representations more different, increasing the word sense disambiguation capabilities of the encoded representations.

Our work presents a novel intrinsic evaluation of sense information in word embeddings, required to understand the sense geometry encoded by various models. In the future, we will focus on integrating sense information in contextual word representations by extending this approach to words that are unseen to the LASeR model, and further perform extrinsic analyses of the embeddings.

6 Conclusion

In this work, we investigated the geometry of contextual word representations for isotropy and sense disambiguation capabilities. We further proposed a post-processing approach for anisotropy treatment and semantic enrichment of contextual word representations, by transforming the vector space using principal component manipulation and lexical semantic knowledge-based sense-retrofitting. Our method significantly reduced the impact of representation degeneration problem, improving isotropy within the vector space and rendered off-the-shelf contextual word vectors semantically more meaningful. In the future work, we will study the impact of changes in retrofitting hyperparameters and variable removal of primary components on representation quality. Further, we will focus on extrinsic evaluation of the impact of anisotropy removal and sense retrofitting on downstream word-sense disambiguation tasks.

Acknowledgements

This work is partially supported by National Science Foundation grant number 1737591. We would like to thank the three anonymous reviewers for their helpful comments and suggestions. We would also like to thank the members of the AKRaNLU lab at Purdue University for their feedback.

References

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.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Jeff Da and Jungo Kasai. 2019. Cracking the contextual commonsense code: Understanding commonsense reasoning aptitude of deep contextual representations. *EMNLP 2019*, page 1.

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

Philip Edmonds and Scott Cotton. 2001. Senseval-2: overview. In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 1–5.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. *arXiv preprint arXiv:1909.00512*.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tieyan Liu. 2018. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations*.

Hwiyeol Jo and Stanley Jungkyu Choi. 2018. Extrofitting: Enriching word representation and its vector space with semantic lexicons. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 24–29.

Dan Jurafsky and James H Martin. 2019. Speech and language processing (3rd draft ed.).

Anne Lauscher, Ivan Vulić, Edoardo Maria Ponti, Anna Korhonen, and Goran Glavaš. 2020a. Specializing unsupervised pretraining models for word-level semantic similarity. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1371–1383.

Anne Lauscher, Ivan Vulić, Edoardo Maria Ponti, Anna Korhonen, and Goran Glavaš. 2020b. Specializing unsupervised pretraining models for word-level semantic similarity. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1371–1383, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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

David Mimno and Laure Thompson. 2017. The strange geometry of skip-gram with negative sampling. In *Empirical Methods in Natural Language Processing*.

Andrea Moro and Roberto Navigli. 2015. Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 288–297.

Jiaqi Mu and Pramod Viswanath. 2018. All-but-the-top: Simple and effective post-processing for word representations. In *6th International Conference on Learning Representations, ICLR 2018*.

Sathvik Nair, Mahesh Srinivasan, and Stephan Meylan. 2020. Contextualized word embeddings encode aspects of human-like word sense knowledge. In *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, pages 129–141.

Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. Semeval-2013 task 12: Multilingual word sense disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (* SEM),* *Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 222–231.

Roberto Navigli, Kenneth C Litkowski, and Orin Hargraves. 2007. Semeval-2007 task 07: Coarse-grained english all-words task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 30–35.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 43–54, Hong Kong, China. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Vikas Raunak, Vivek Gupta, and Florian Metze. 2019. Effective dimensionality reduction for word embeddings. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 235–243.

Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B Viegas, Andy Coenen, Adam Pearce, and Been Kim. 2019. Visualizing and measuring the geometry of bert. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Maarten Sap, Vered Shwartz, Antoine Bosselut, Yejin Choi, and Dan Roth. 2020. Commonsense reasoning for natural language processing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 27–33, Online. Association for Computational Linguistics.

Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 298–307.

Benjamin Snyder and Martha Palmer. 2004. The english all-words task. In *Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43.

Loïc Vial, Benjamin Lecouteux, and Didier Schwab. 2018. Ufsac: Unification of sense annotated corpora and tools. In *Language Resources and Evaluation Conference (LREC)*.

Ivan Vulić. 2018. Injecting lexical contrast into word vectors by guiding vector space specialisation. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 137–143.

Ivan Vulić and Nikola Mrkšić. 2018. Specialising word vectors for lexical entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1134–1145.

Bin Wang, Angela Wang, Fenxiao Chen, Yuncheng Wang, and C-C Jay Kuo. 2019. Evaluating word embedding models: Methods and experimental results. *APSIPA transactions on signal and information processing*, 8.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Cuihong Cao, Daxin Jiang, Ming Zhou, et al. 2020. K-adapter: Infusing knowledge into pre-trained models with adapters. *arXiv preprint arXiv:2002.01808*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5753–5763.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Zhong Zhang, Chongming Gao, Cong Xu, Rui Miao, Qinli Yang, and Junming Shao. 2020. Revisiting representation degeneration problem in language modeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 518–527.

A Anisotropy Across All Words

We plot the average similarity between all words (multi-sense nouns, verbs and adjectives) extracted from the annotated corpora, across model layers, as shown in Figure 7.

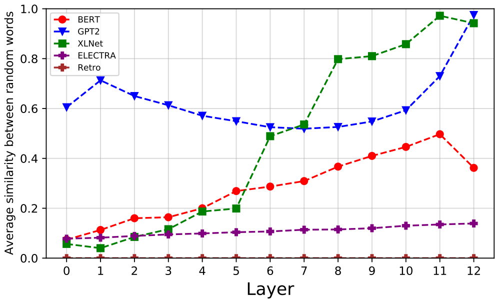
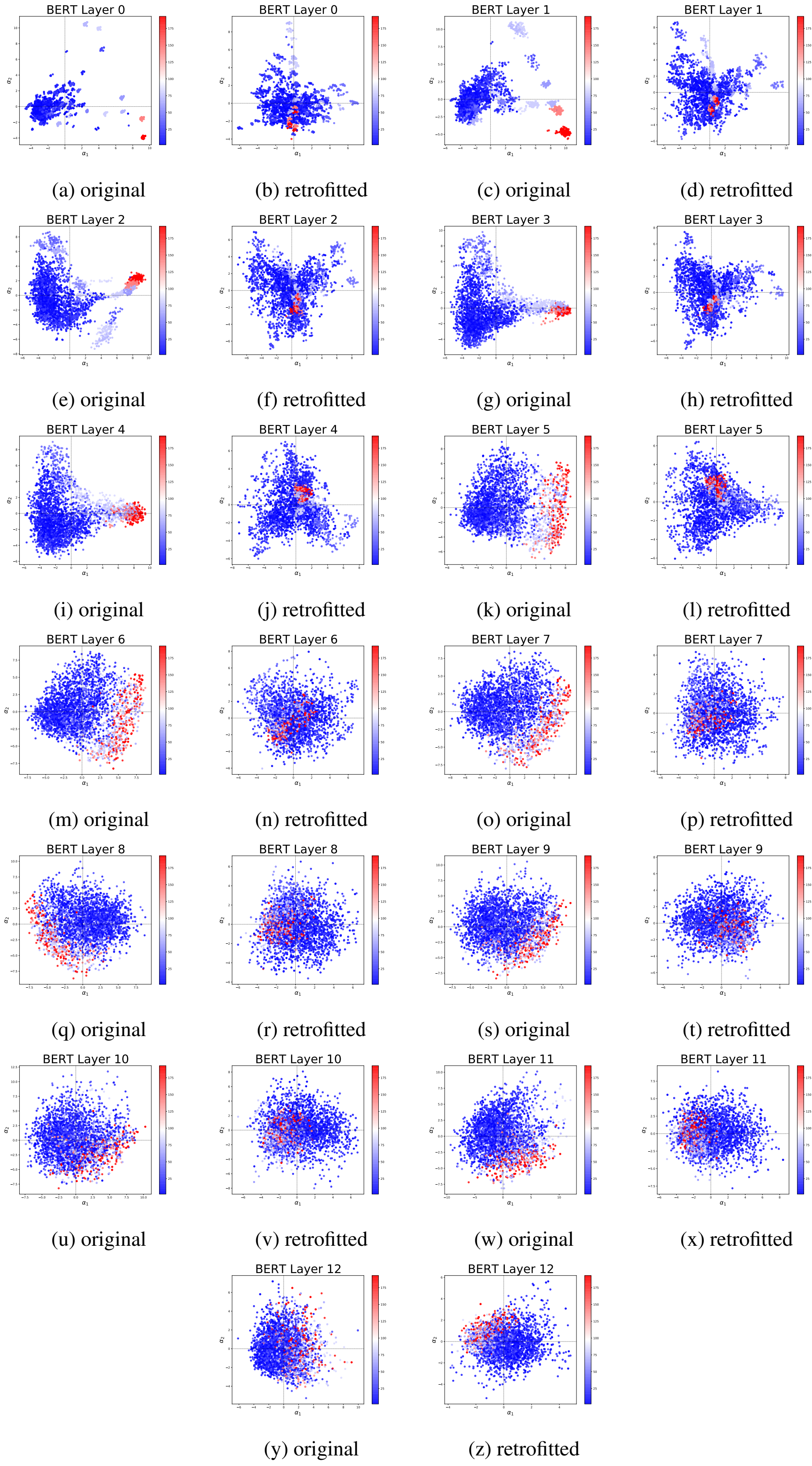


Figure 7: Average similarity between representations of randomly sampled words across model layers.

B PCA Plots of Word Representations

We plot distribution of word representations across the vector space, for all models across their layers. To assess whether word frequency is encoded within vector dimensions, we color code representations ranging from low frequency words (Blue) to high frequency words (Red). The plots are given in Figure 8 (BERT), Figure 9 (GPT-2), Figure 10 (XLNet) and Figure 11 (ELECTRA). We see that using LASeR post-processing ($d = 1$ and hyperparameters mentioned in the main text), anisotropy in vector space is significantly treated. For extremely anisotropic models such as GPT2 and ELECTRA, remove of the first primary component yields more uniformly spread word representations.



(a) original (b) retrofitted (c) original (d) retrofitted

(e) original (f) retrofitted (g) original (h) retrofitted

(i) original (j) retrofitted (k) original (l) retrofitted

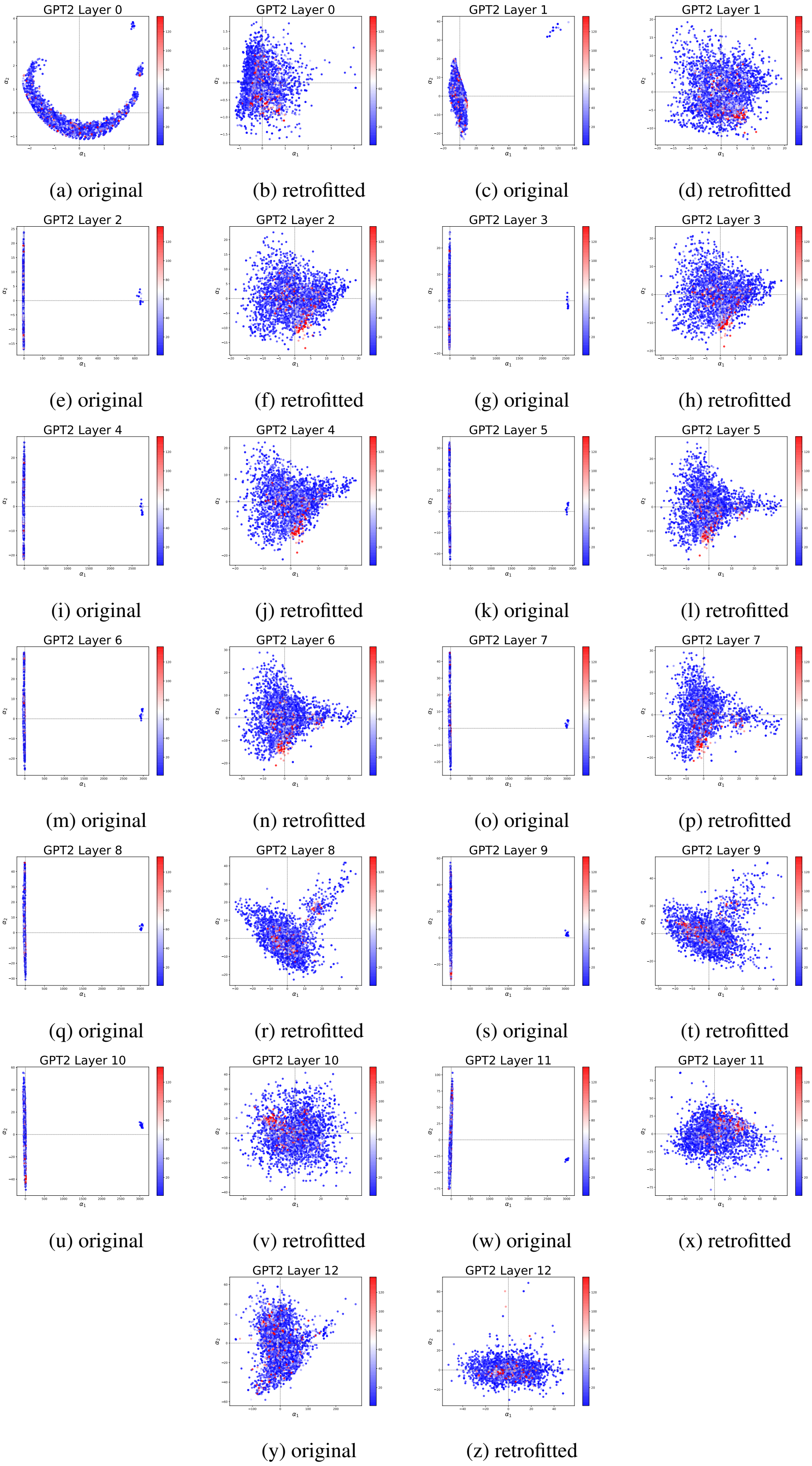
(m) original (n) retrofitted (o) original (p) retrofitted

(q) original (r) retrofitted (s) original (t) retrofitted

(u) original (v) retrofitted (w) original (x) retrofitted

(y) original (z) retrofitted

Figure 8: PCA Plots of BERT Word Representations.



GPT2 Layer 0

(a) original

GPT2 Layer 0

(b) retrofitted

GPT2 Layer 1

(c) original

GPT2 Layer 1

(d) retrofitted

GPT2 Layer 2

(e) original

GPT2 Layer 2

(f) retrofitted

GPT2 Layer 3

(g) original

GPT2 Layer 3

(h) retrofitted

GPT2 Layer 4

(i) original

GPT2 Layer 4

(j) retrofitted

GPT2 Layer 5

(k) original

GPT2 Layer 5

(l) retrofitted

GPT2 Layer 6

(m) original

GPT2 Layer 6

(n) retrofitted

GPT2 Layer 7

(o) original

GPT2 Layer 7

(p) retrofitted

GPT2 Layer 8

(q) original

GPT2 Layer 8

(r) retrofitted

GPT2 Layer 9

(s) original

GPT2 Layer 9

(t) retrofitted

GPT2 Layer 10

(u) original

GPT2 Layer 10

(v) retrofitted

GPT2 Layer 11

(w) original

GPT2 Layer 11

(x) retrofitted

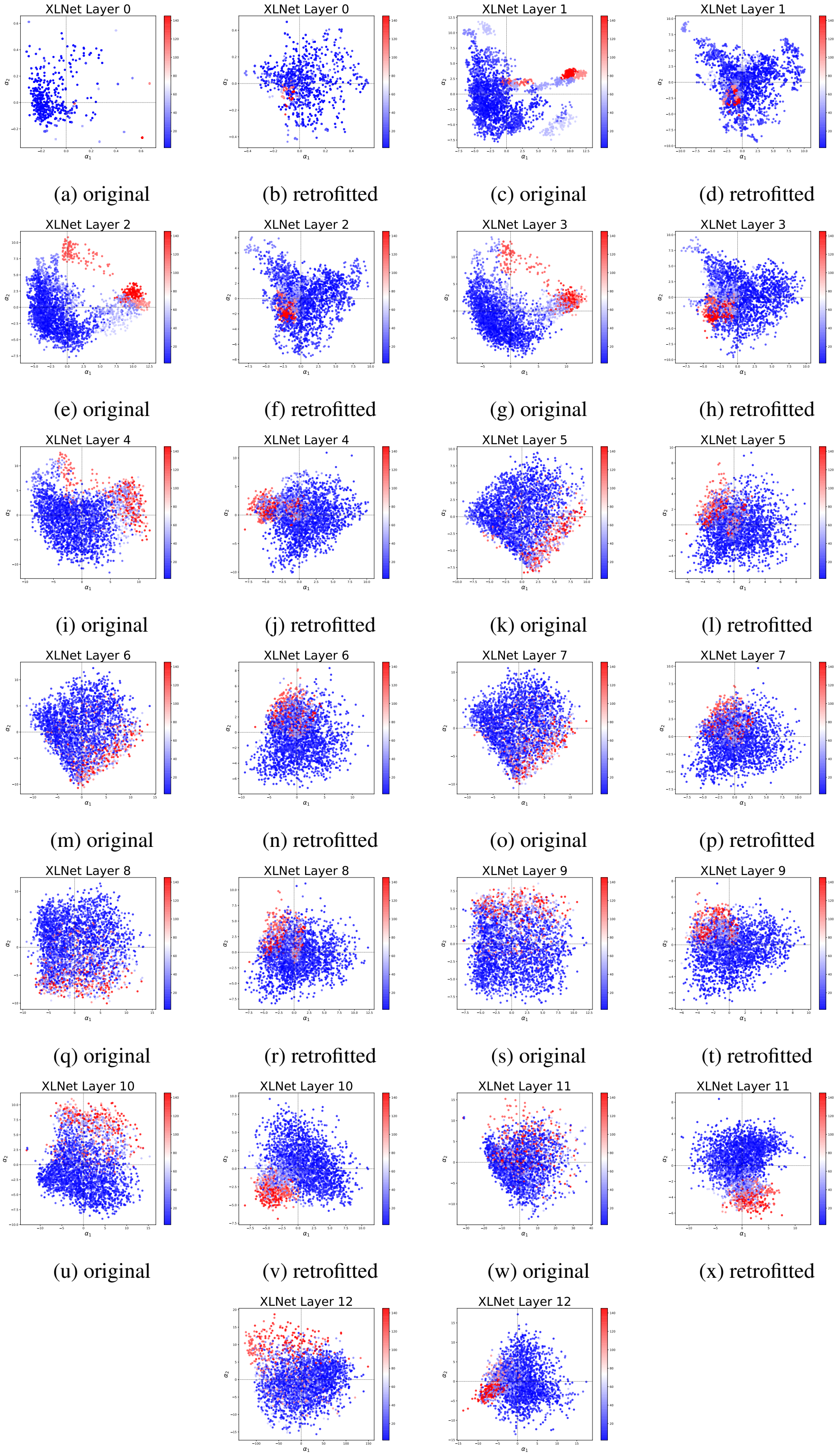
GPT2 Layer 12

(y) original

GPT2 Layer 12

(z) retrofitted

Figure 9: PCA Plots of GPT2 Word Representations.



(a) original (b) retrofitted (c) original (d) retrofitted

(e) original (f) retrofitted (g) original (h) retrofitted

(i) original (j) retrofitted (k) original (l) retrofitted

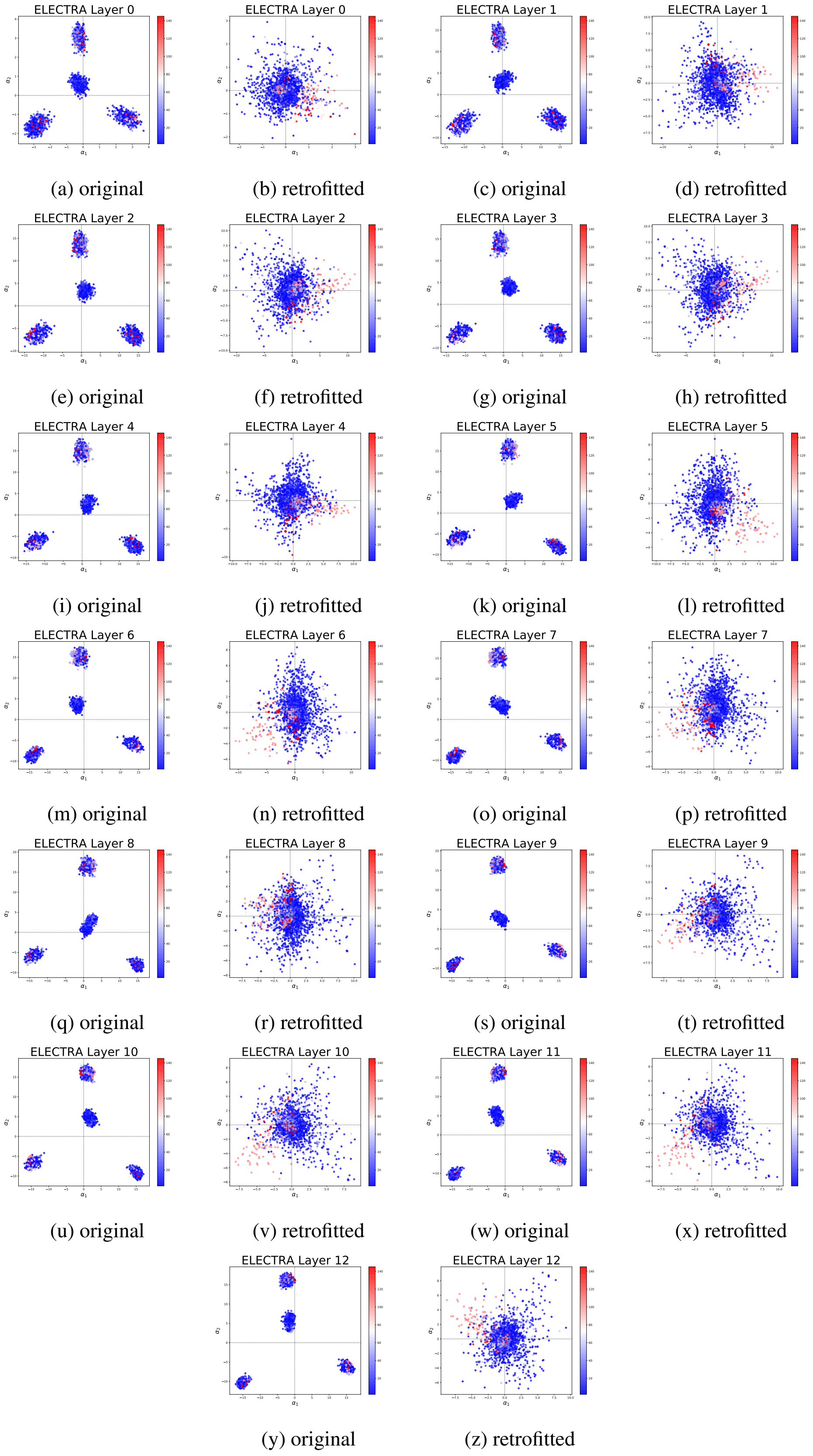
(m) original (n) retrofitted (o) original (p) retrofitted

(q) original (r) retrofitted (s) original (t) retrofitted

(u) original (v) retrofitted (w) original (x) retrofitted

(y) original (z) retrofitted

Figure 10: PCA Plots of XLNet Word Representations.



ELECTRA Layer 0

(a) original

ELECTRA Layer 0

(b) retrofitted

ELECTRA Layer 1

(c) original

ELECTRA Layer 1

(d) retrofitted

ELECTRA Layer 2

(e) original

ELECTRA Layer 2

(f) retrofitted

ELECTRA Layer 3

(g) original

ELECTRA Layer 3

(h) retrofitted

ELECTRA Layer 4

(i) original

ELECTRA Layer 4

(j) retrofitted

ELECTRA Layer 5

(k) original

ELECTRA Layer 5

(l) retrofitted

ELECTRA Layer 6

(m) original

ELECTRA Layer 6

(n) retrofitted

ELECTRA Layer 7

(o) original

ELECTRA Layer 7

(p) retrofitted

ELECTRA Layer 8

(q) original

ELECTRA Layer 8

(r) retrofitted

ELECTRA Layer 9

(s) original

ELECTRA Layer 9

(t) retrofitted

ELECTRA Layer 10

(u) original

ELECTRA Layer 10

(v) retrofitted

ELECTRA Layer 11

(w) original

ELECTRA Layer 11

(x) retrofitted

ELECTRA Layer 12

(y) original

ELECTRA Layer 12

(z) retrofitted

Figure 11: PCA Plots of ELECTRA Word Representations.