Tracing the Development of Word Meaning During Training

Stanford CS224N Custom Project

Shenghua Liu
Department of Physics
Stanford University
sliu24@stanford.edu

Yiheng Ye
Department of Physics
Stanford University
yihengy@stanford.edu

Abstract

Analyzing the training dynamics of transformer-based language models is crucial for understanding how such models acquire impressive capabilities and knowledge from training. Here, we study the development of word meaning over training by utilizing the *Pythia* Biderman et al. (2023) suite to track the dynamics of word embeddings over training steps. We study the structure of the embedding space by comparing word-pair similarity scores in the WordSim353 Finkelstein et al. (2001) dataset and GloVe Pennington et al. (2014b) embeddings as benchmarks to the corresponding embeddings in model checkpoints. We provide detailed analysis of the training dynamics using various baseline as comparison, and different metrics. Our findings suggest that a metric composed of two Kendall-Tau distance may capture the fundamental structural differences between the human-scored WordSim353 dataset and the GloVe, while a more standard metric based on cosine similarities failed to capture such characteristics. Furthermore, we study the evolution in 2D of the embedding space, and we find that the learning of word meaning occues early in training and does not proceed much even if it is not well learned.

1 Key Information to include

- TA Mentor: Kaylee Burns
- External mentor: Isabel Victoria Papadimitriou
- External Collaborators: No
- Sharing project: No
- Team contributions: Both members contribute significantly to the brainstorming, coding, and writing of this report.

2 Introduction

Large language models (LLMs) have demonstrated significant success in various downstream tasks, especially in few-shot and zero-shot scenarios. Consequently, researchers have been exploring the nature of information these networks acquire and the methods by which this information is encoded within the model's parameters. Analyzing the training dynamics of transformer-based language models is crucial for understanding how such models acquire impressive capabilities and knowledge from training.

The gateway into higher-level reasoning in LLMs is the very first layer, or the embedding layer. Intuitively, the embedding layer should encode the meanings of individual words, which are further processed down the line. Even though the embedding space has been extensively studied for trained models, an important yet underexplored question is: how are word meaning formed throughout the

training? Various related question can be asked: to what extent does the word embedding capture the meaning of the word, or equivalently, to what extent is the meaning of the word are captured in the later layers, and are inferred from the context? Do LLMs encode word meanings in the same way that humans do? Answers to these questions can significantly contribute to the interpretability of LLMs, thereby providing a strong foundation for the construction of trustworthy AI system.

In this study, we focus on the evolution of word embeddings throughout the training process using the *Pythia*Biderman et al. (2023) suite, which contains model checkpoints over different training steps. We assess the learning of word meanings by comparing word-pair similarity scores from the WordSim353Finkelstein et al. (2001) dataset and GloVe Pennington et al. (2014b) embeddings to the corresponding embeddings in various model checkpoints. Beyond these summary-score comparisons, we also visualize how the embedding space structure forms by projecting it down to two dimensions using singular value decomposition (SVD). Furthermore, we study the effect of model size on the dynamics of the embedding space to see in what form does larger models learn better.

A very important observation that emerges from these analyses is the importance of metric chosen to compare different high-dimensional word embeddings and similarity scores. The metric is much more than a trick to obtain desirable, intuitive results, but a fundamental choice to capture the prominant characteristics of the data. The importance of the metric chosen is particularly explored in Timkey and van Schijndel (2021), where the authors call into question the informativity of standard representational similarity measures such as cosine similarity and Euclidean distance for contextualized language models. In our work, we explore the difference between cosine similarities and ranking-based metric when applied to the analysis of word-embeddings. Our findings suggest that the metric does play an important role in drawing conclusions and therefore should be considered carefully in future work dealing with high-dimensional space comparisons.

Overall, our analyses provide insight into the underexplored mechanisms through which transformer-based language models acquire word meanings, contributing to a scientific understanding of the impressive capabilities of transformers.

3 Related Work

Our paper addresses problems lying in the emergent field of training dynamics. For more information regarding this extensive body of work, please refer to this well-written review article on emergent structures and training dynamics in LLMs Teehan et al. (2022).

Most studies of Transformers and LSTMs agree that models acquire linguistic knowledge quite early in the learning process Zhuang et al. (2021); Hochreiter and Schmidhuber (1997). There have also been studies showing that local syntactic information, e.g. parts of speech, is learned by LLMs earlier than information encoding long-distance dependencies, such as main theme, etc Liu et al. (2021); Saphra (2021). Studies indicate that ALBERT and LSTM-based networks exhibit distinct learning patterns for function and content words. These differences extend to more specific classifications such as parts of speech and verb forms, highlighting finer distinctions within these word categories Lan et al. (2019); Saphra (2021); Chiang et al. (2020). Variations in learning trajectories were also noted across layers. In LSTMs, recurrent layers become more task-independent during training, whereas embeddings grow more task-specific Saphra (2021). For Transformer-based models like ALBERT and ELECTRA, Chiang et al. (2020) identified performance pattern differences between the top and bottom layers. These works also motivate us to explore the structure formation in different layers, especially in the word embedding layer.

A particularly motivating work is a somewhat older paper Zhang et al. (2021). This paper investigates the impact of pretraining data volume on Transformer-based language models like RoBERTa. It challenges the notion that larger datasets always enhance performance, showing substantial linguistic knowledge can be gained from smaller datasets. However, mastering downstream NLU tasks and commonsense knowledge still requires larger datasets. The study also discusses the ethical and environmental implications of large-scale model training and suggests ways to make training more data-efficient and accessible, potentially democratizing NLP technology and addressing ethical concerns. Our project aims to trace the development of word meaning in LLMs during training. The paper by Zhang et al. (2021) offers an inspiring parallel by examining the formation of common knowledge in models, showing how training data size impacts this and downstream task performance. While their focus is on common knowledge, our study investigates how models develop meaningful

word representations and transition to effective meaning spaces. The methods and insights from their study could significantly guide our research on linguistic feature development in LLMs.

4 Approach

We utilize *Pythia* Biderman et al. (2023), a public suite of 16 transformer-based LLMs all trained on public data seen in the exact same order and ranging in size from 70M to 12B parameters with 154 checkpoints each. We write code to extract word embeddings (which is the first layer of the model) given a checkpoint of a model. In the first part of the study, we use WordSim353 and GloVe as our two main baselines and define metrics which we use to compare model word embeddings at different checkpoints to these baselines. This allows us to benchmark how well the model has learned word meanings over training. We perform these comparisons for different model sizes to study their effect on embedding formation.

4.1 Comparison Metrics

The core challenge is to sensibly compare these high-dimensional spaces (many high-dimensional word vectors) and draw conclusions. The WordSim353 dataset contains 353 pairs of words rated on their similarity on a scale of 1 to 10 by human survey participants. Therefore, to analyze the degree to which a word embedding space encodes word meanings, we need two metrics: one to measure the distance, or similarity, between words, and the other (a meta-metric) to quantify how close these similarity scores are to the baseline.

To get the similarity between a pair of words, an intuitive starting point is the cosine similarity metric. However, once we have these similarity scores, we need to them compare them to those from the baseline. Since the baseline scores are not on the same scale as the cosine similarities from the model embeddings, we need a metric for fair comparison across these spaces. Therefore, as a coarse-grained but more general approach, we compute the *rankings* of the word pair similarities, which intuitively should be the same in both word embeddings and WordSim353 if word meanings are perfectly learned.

We quantify the distance between two sets of similarity scores using the Kendall Tau (KT) distance on their rankings. Recall that the Kendall Tau distance for two lists τ_1 and τ_2 is:

$$K_d(\tau_1, \tau_2) = |\{(i, j) : i < j, [\tau_1(i) < \tau_1(j) \wedge \tau_2(i) > \tau_2(j)] \vee [\tau_1(i) > \tau_1(j) \wedge \tau_2(i) < \tau_2(j)]\}|,$$

where $\tau_1(i)$ and $\tau_2(i)$ are the rankings of the element i in τ_1 and τ_2 respectively. We normalize it by defining $K_n = \frac{K_d}{\frac{1}{2}n(n-1)}$ such that it lies in the interval $[0, 1]$. A KT score of 1 indicates perfect ranking match, 0 a perfect antimatch, and 0.5 no relation.

4.2 Comparisons to Baselines

Concretely, to study how well the model has learned word meanings over training, we perform the following comparisons:

- For each model (with a different parameter size), we use its respective word embeddings at the last checkpoint as the baseline and compare it to the embeddings at every checkpoint. This gives us an idea of how the embeddings converge to its final state.
- We compare the embedding at each checkpoint of each model to the last checkpoint of the biggest (410M) model. Assuming that bigger models lead to "better" embeddings, this gives us an idea of how the well the embeddings are formed when compared to better versions of the same learning system (allowing us to exclude the complication that humans, GloVe, and LLMs may not learn word meanings in the same way).
- We compare the embedding at each checkpoint of each model to the same checkpoint of the biggest (410M) model. This gives us an idea of how synchronized the training dynamics of differently-sized models are.
- We compare the embedding at each checkpoint of each model to GloVePennington et al. (2014a). This gives us an idea of how well the embeddings are formed compared to a well-established, quantitative, and relatively interpretable baseline.

- We compare the embedding at each checkpoint of each model to WordSim353Finkelstein et al. (2001). This allows us to evaluate how the embeddings are formed compared to human-rated similarity scores. Even though humans are largely considered the gold standard of language understanding, the similarity scores are obtained simply by survey, which may be quite imprecise and subjective.

4.3 Visualizing Embedding Structure Evolution

For the second part of the analysis, we go beyond summary-scores calculated from the metrics above and visualize embedding structure evolution in 2D by SVD. We randomly select 5 word pairs. In the 2D SVD space, we plot the evolution of the 10 words in both GloVe and the 5 models we consider. We also mark the WordSim353 scores between each pair. This analysis allows to visualize the formation of structure in the embedding space, and results are shown in Fig 7.

5 Experiments

5.1 Data

We use the pretrained LLM models available in *Pythia* Biderman et al. (2023) as described above. In particular, due to time and space constraints, our analysis are based on models with 14M, 31M, 70M, 160M, 410M parameters, respectively. We use all 154 checkpoints for each model. We use WordSim353 Finkelstein et al. (2001) and GLoVe Pennington et al. (2014b) as baseline for comparison. Of the 353 word pairs in WordSim353, only 96 are within the vcabulary of both GloVe and the model tokenizers. Therefore, we only use these 96 pairs of words for our analyses for consistency.

5.2 Experimental details

We load models with the above five different parameter sizes and save the word embeddings at each checkpoint for the 96 words. We compute the similarities of word-pairs, and compare these similarity scores across spaces using different comparison metrics. We use two different metrics:

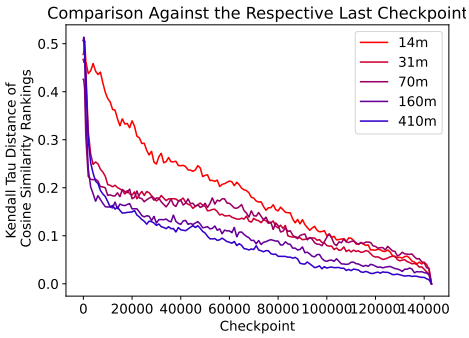
- For a given embedding, we compute the cosine similarity between words in a pair. To compare these 96-dimensional scores, we first rank them, and compute the KT distance with the similarity scores from the baseline. Intuitively, more similar words should have higher cosine similarities, so the rankings of the cosine similarities should be close if the two embedding spaces are similar. We call it the "KTcos" metric. Results using this metric is shown in figure 6.
- We use the KT distance in both steps. For each word pair, instead of cosine similarity, we simply rank each word vector's entries and compute the KT distance between them. Then, we compute the KT distance with the similarity scores from the baseline. We call it the "KTKT" metric. This metric is partially motivated by Timkey and van Schijndel (2021), which suggests that cosine similarity may not be a good metric to compare high-dimensional vectors. Results using this metric is shown in figure 2.

6 Results and Analysis

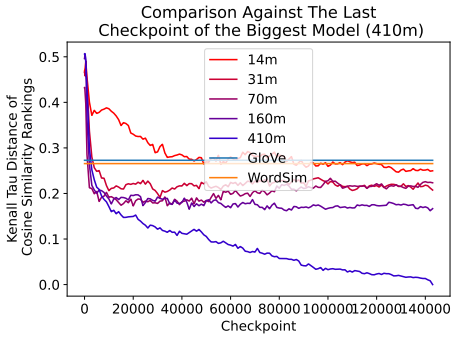
6.1 Comparisons to Baselines

With both metrics, we can see a relatively smooth behavior as each model converge to their respective last checkpoints, as shown in figure 1(a) and 2(a). We observe that there is a rapid descent of the lost curve, indicating that most word meaning learned by the model up till the last checkpoint is learned at the very beginning of the training. This feature is more significant as we increase the model size, and can also be seen in the following comparisons, notably shown in figure 1(b), 2(b), 1(c), and 2(c).

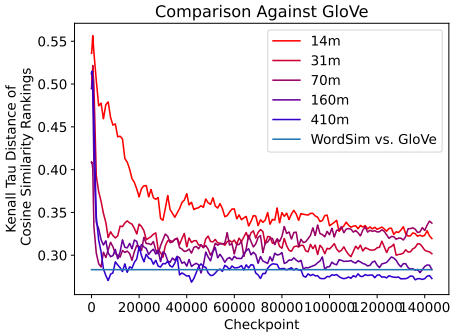
When we make the last checkpoint of the biggest model our baseline, we observe that models with other sizes converge and stabilize at a $\mathcal{O}(1)$ distance computed in both metrics, as shown in figure 1(b) and 2(b). Bigger model converges to a closer distance to the biggest model. This is within our



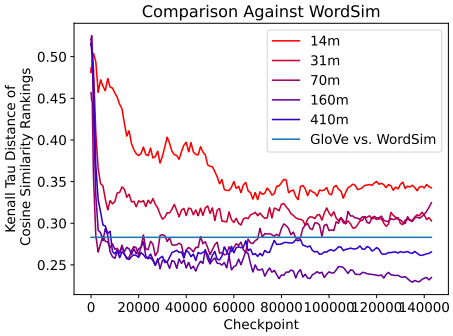
(a) KTcos Comparisons of each checkpoint against the last checkpoint of the same model.



(b) KTcos Comparisons of each checkpoint against the last checkpoint of the biggest model (410M).



(c) KTcos Comparisons of each checkpoint against GloVe embeddings.



(d) KTcos Comparisons of each checkpoint against WordSim353 scores.

Figure 1: Training dynamics captured by the Kendall-Tau distance of cosine similarity rankings.

expectation, since neural scaling law tells us that under the same condition bigger model generically produces better results (we assume that the last checkpoint of the biggest model has, among all, the "best" embedding.) Moreover, this stabilization of distance at a value somewhat bigger than when compared to GloVe or WordSim353 indicates that the model embeddings end up at some distance "around" the GloVe or WordSim353 baseline in the high-dimensional space.

Note that in figure 1(b) and 2(b), we also plot the distance between the last checkpoint of the biggest model and the GloVe/WordSim embedding. These results are somewhat unexpected. When the distance is computed using the KT distance of the cosine similarity ranking, the distance with respect to the WordSim and GloVe are respectively similar (roughly 0.27). If the distance is computed using the KT distance of the KT similarity ranking, GloVe still has a relatively small distance with respect to the embedding at the last checkpoint of the biggest model, while the WordSim, on the contrary, has a huge distance (roughly 0.72) with respect to the same baseline. This indicates that different metrics may capture different features of the word embedding. One possibility is that the human-rated scores in WordSim353 may have other considerations factored in, such as whether the words tend to appear in the same context even though they are not similar per se. GloVe, on the other hand, is a machine-learned representation, which may have closer resemblance to the embedding layer of LLMs.

We further use GloVe as the baseline, and compare the embedding at each checkpoint to it. The result is demonstrated in figure 1(c) and 2(c). As expected, under both metrics, the embedding of each model converges to a small distance to GloVe. Assuming the GloVe as the optimal embedding, bigger model has better performance (converges more quickly to a smaller distance to GloVe). The exotic difference between GloVe and WordSim appears again: the distance between GloVe and WordSim is small with respect to the KTcos metric, and big with respect to the KTKT metric. Our question is still unresolved.

Our fourth comparison gives us some hint. We use WordSim as the baseline, and compute, under both metrics, the distance from each checkpoint of each model to it. The result is shown in figure 1(d) and 2(d). We can see that when we use the KTcos metric, we have dynamics satisfy our expectation: as the training proceed, the KTcos distance with respect to WordSim descents and converges to a small $\mathcal{O}(1)$ value, with bigger model leading to better convergence. But if we repeat the computation using KTKT metric, the KTKT distance with respect to WordSim actually increases as the training proceeds!

To further illustrate this phenomenon, we examine the dynamic of the checkpoint-wise distance, using each checkpoint of the biggest model as the baseline. We can see that the dynamic is captured equally well by both metrics under most circumstances, and satisfy the general expectation from the neural scaling law. However, under the KTKT metric, they behave drastically differently. In particular, the KTKT distance between each model at each checkpoint and the WordSim actually increases drastically as the training proceeds.

We come up with two (not mutually exclusive explanations):

- The KTKT metric captures different features of the word embedding.
- Word meaning is represented or conveyed in a way fundamentally different from human beings (as represented by WordSim).

The second explanation seems more reasonable to us. In most scenarios we considered in this paper, KTKT metric and KTcos metric captures very similar results. It is thus most reasonable to conjecture that KTKT metric captures some fundamental difference between WordSim and GloVe.

WordSim is of course a very different data set from GloVe: it is based on human being's subjective similarity scores, while GloVe is obtained from training neural networks. It is very reasonable to assume that word meaning is represented in a way very different from human being in LLMs, and in this case the KTKT metric is a better metric that captures this fundamental difference.

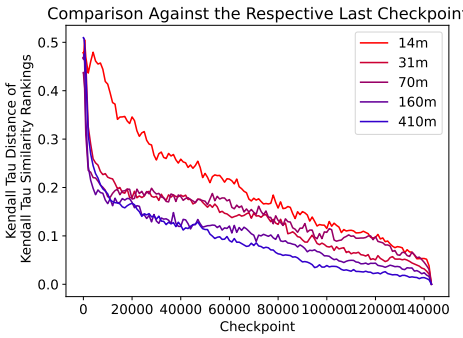
6.2 2D Evolution of the Embedding Space

Figure. 7 illustrates the 2D evolution of the embedding spaces of the 5 models in addition to the static representation of GloVe. One of the most striking features we see is that bigger models have much "cleaner" evolution: The word embeddings start near the origin and shoot off almost in straight lines in different directions. This is clearly seen in the 410M plot, while in contrast, the 14M plot shows that the evolution of each word embedding is much more chaotic. This supports the common knowledge that bigger models capture better representations of word meaning.

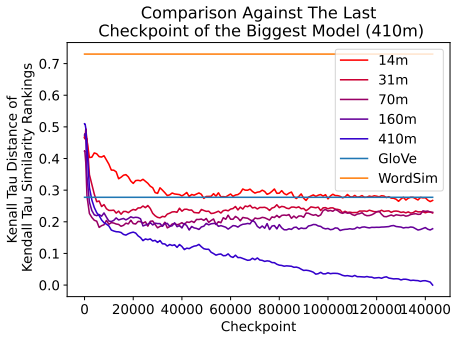
In terms of dynamics, these plots corroborate our findings earlier that word meaning is learned very early in training. This is seen in every subplot as most of the displacement from the origin is covered early on, and eventually the embedding just settles down. An interesting point is that in the smaller models, even though they have not learned the word meanings well near the end of training, the word embeddings have settled down nonetheless. This suggests that the training dynsmics of LLMs is indeed hierarchical—later stages of learning happens mostly in later layers despite the embedding layer not being learned perfectly.

7 Conclusion

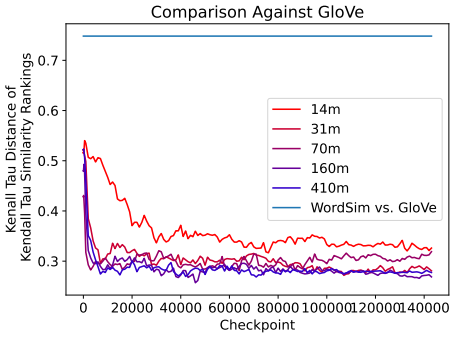
In this paper, we provide detailed analysis of the training dynamics using various baseline as comparison, and different metrics. We found that the KTKT metric (defined in our main text) captures the fundamental structural difference between the human-scored WordSim dataset and the GloVe, while a more standard metric based on cosine similarities failed to capture such characteristic. We explain such phenomena by concluding that word meaning is represented in an exotic way in LLMs that is fundamentally different from human being's intuitive understanding. We also visualize embedding structure evolution in 2D by ploting the evolution of the 10 words in both GloVe and the 5 models we consider, as well as the WordSim353 scores between each pair. This enables us to visualize the formation of structures, including word meaning.



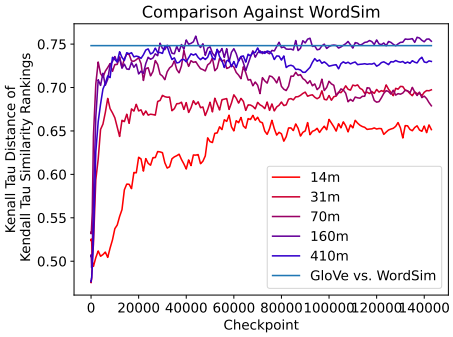
(a) KTKT comparisons of each checkpoint against the last checkpoint of the same model.



(b) KTKT comparisons of each checkpoint against the last checkpoint of the biggest model (410M).

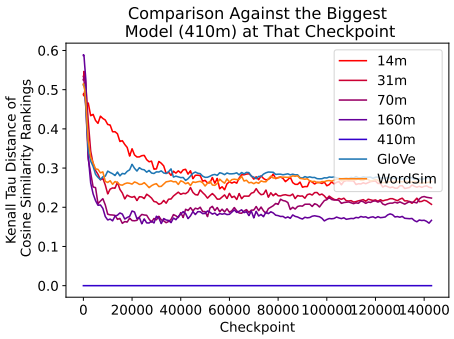


(c) KTKT comparisons of each checkpoint against GloVe embeddings.

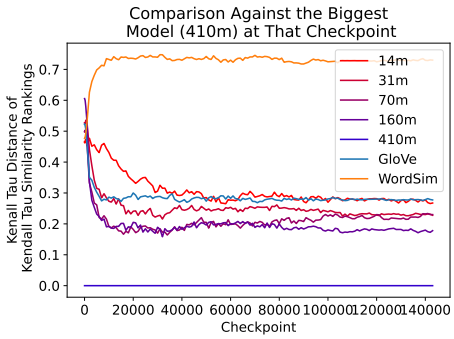


(d) KTKT comparisons of each checkpoint against WordSim353 scores.

Figure 2: Training dynamics captured by the Kendall-Tau distance of Kendall-Tau distance similarity rankings.

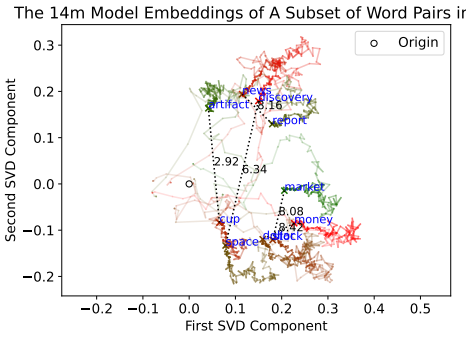


(a) KTcos comparisons of each checkpoint against the same checkpoint of the biggest model (410M).

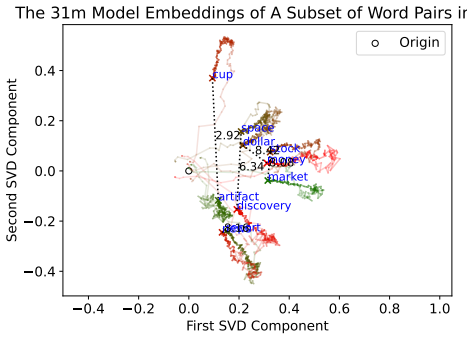


(b) KTKT comparisons of each checkpoint against the same checkpoint of the biggest model (410M).

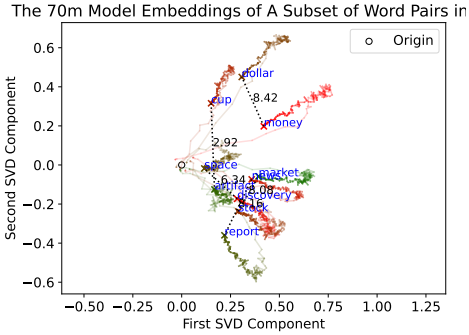
Figure 3: Comparisons against the biggest model (410m) at respective checkpoints using two different metrics.



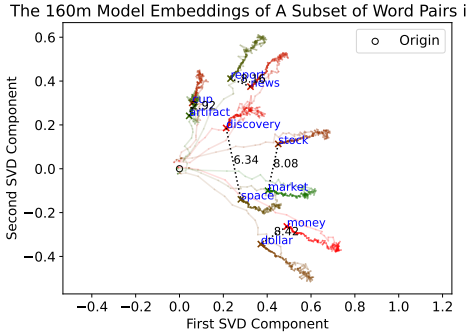
(a) 2D SVD for 14m model



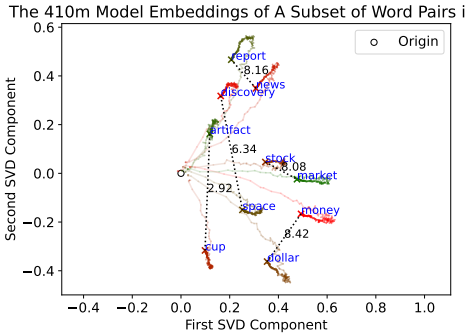
(b) 2D SVD for 31m model



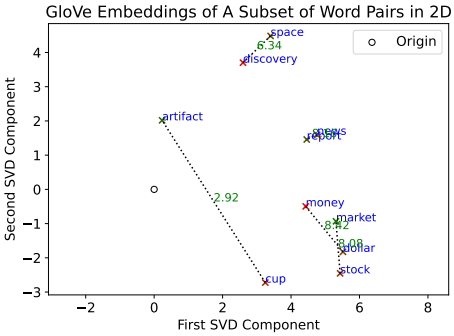
(c) 2D SVD for 70m model



(d) 2D SVD for 160m model



(e) 2D SVD for 410m model



(f) 2D SVD for GloVe

Figure 4: 2D visualization of the dynamic of the first and the second SVD component of the embedding of a subset of word pairs.

8 Ethics Statement

One ethical issue related to this project is the potential reinforcement of biases in language models. By analyzing the development of word meanings during training, we risk highlighting and propagating existing biases in the training data, which could perpetuate stereotypes and unfair treatment in real-world applications. In particular, WordSim is a human-scored word similarities dataset, which is obviously inherently biased. Models on which our analysis is based are also trained on biased data. Thus, all of our conclusions are based on an inherently biased foundation. To mitigate this, it is crucial to incorporate bias detection and correction strategies, such as using diverse datasets and implementing fairness constraints during the training process.

Another societal risk involves the misuse and misinterpretation of our research findings, leading to overconfidence in AI capabilities. This could result in AI systems being deployed in critical decision-making processes without adequate oversight, especially in legal, medical, or financial contexts. To address this, we need to clearly communicate the limitations and uncertainties of our research, emphasizing the need for careful and contextualized interpretation of findings before practical application. Comprehensive documentation and engaging with interdisciplinary experts can help ensure a balanced understanding of our work's implications.

References

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. 2023. Pythia: a suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Cheng-Han Chiang, Sung-Feng Huang, and Hung-yi Lee. 2020. Pretrained language model embryology: The birth of albert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6813–6828. Association for Computational Linguistics.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *ACM Transactions on Information Systems - TOIS*, volume 20, pages 406–414.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Zeyu Liu, Yizhong Wang, Jungo Kasai, Hannaneh Hajishirzi, and Noah A. Smith. 2021. Probing across time: What does RoBERTa know and when? In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 820–842, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jeffrey Pennington, R Socher, and Ch D Manning. 2014a. Glove: Global vectors for word representation. empirical methods in natural language processing (emnlp), 1532–1543.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014b. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Naomi Saphra. 2021. *Training dynamics of neural language models*. Ph.D. thesis, The University of Edinburgh.

Ryan Teehan, Miruna Clinciu, Oleg Serikov, Eliza Szczechla, Natasha Seelam, Shachar Mirkin, and Aaron Gokaslan. 2022. Emergent structures and training dynamics in large language models. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 146–159, virtual+Dublin. Association for Computational Linguistics.

William Timkey and Marten van Schijndel. 2021. All bark and no bite: Rogue dimensions in transformer language models obscure representational quality. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4527–4546, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yian Zhang, Alex Warstadt, Xiaocheng Li, and Samuel R. Bowman. 2021. When do you need billions of words of pretraining data? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1112–1125, Online. Association for Computational Linguistics.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China.