Interpretable Steering of Dense Embeddings for Controllable Retrieval

Anonymous ACL submission

Abstract

Embedding-based information retrieval models can suffer from false retrievals due to issues such as training data bias and polysemy. To address this problem, we propose a novel method for controlling embedding models through an interpretable steering technique based on Sparse Autoencoders (SAEs). SAEs decompose embeddings into semantically disentangled features, and a steering vector selectively enhances or suppresses features that contribute to false retrievals, thereby correcting the search results. Experimental results demonstrate that the proposed method effectively rectifies false retrievals within a limited range while maintaining the generalization performance of the model. However, limitations of the SAE, including potential performance degradation, possible side effects from polysemantic features, and the difficulty in determining optimal correction values, indicate the need for further research. Future work should focus on overcoming these limitations and expanding the scope of the interpretlalbr steering technique to build a more sophisticated search result correction system.

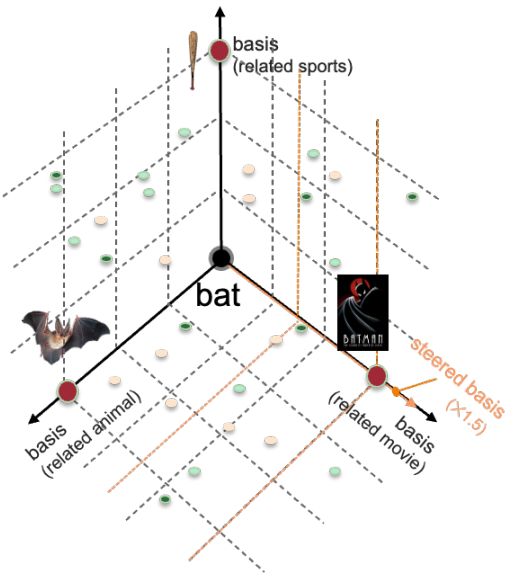

Figure 1: **Steering in Embedding space.** An illustration of the semantic space of the word 'bat' across different contextual bases (sports, animals, and movies). Steering the embedding 1.5× toward a movie-related context reduces the unit distance from three units (three-unit distance) to two units (two-unit distance), decreasing the semantic gap between 'bat' and its meaning in a movie-related context, making 'bat' more closely associated with 'Batman'.

1 Introduction

The rapid advancements in Large Language Models (LLMs) (Ouyang et al., 2022; Achiam et al., 2023; Touvron et al., 2023) emphasize the significance of efficient and precise information retrieval (Zhao et al., 2023). Embedding models (Gao et al., 2022; Karpukhin et al., 2020), which convert text into vector representations for similarity-based retrieval, have become essential for knowledge discovery and information search (Izacard and Grave, 2020; Shi et al., 2023).

However, embedding models exhibit biases categorized into two types: Intrinsic bias (Sun et al., 2019), which originate from the training data, as well as the architecture and underlying assumptions made during model design,and Extrinsic bias (Kiritchenko and Mohammad, 2018), which emerge during the application of LLMs in real-world tasks (Guo et al., 2024). These biases can skew search results, prioritizing on certain interpretations over others and misaligning with user intent. For instance, a query for "bat" may predominantly return results related to baseball rather than the animal, or "apple" may emphasize Apple Inc. over the fruit (Figure 1).

The black box nature of neural network-based models makes it difficult to understand the factors driving specific retrieval results. This lack of transparency undermines trust and limits the ability

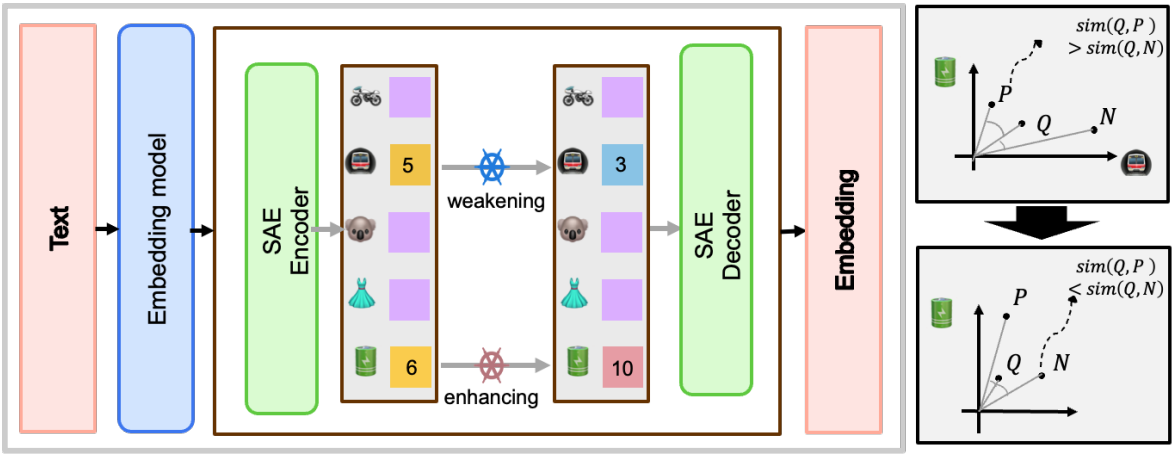

Figure 2: **Interpretable Steering in Embedding. (left) Steering the representation of an embedding model using a SAE.** The input text is transformed into an embedding vector, which is then decomposed into sparse features by SAE. By manipulating specific feature activations related to "train" and "battery", we can directly modify the embedding, thereby steering its semantic meaning. **(right) Changes in the Corrected Embedding Space and Retrieval Adjustment.** In the original embedding space (top), the positive instance (P) is retrieved due to its higher similarity to the query (Q). In the reconstructed embedding space by steering (bottom), the negative instance (N) becomes more similar to the query and is thus retrieved instead. This demonstrates that the steering process can dynamically adjust the actual retrieval ranking.

to identify or correct unintended biases. O'Neill et al. (2024) introduced the Sparse AutoEncoder to improve interpretability, providing insights into internal representations and aiding in potential bias mitigation.

Interpretable steering is gaining attention as a technique to actively adjust black-box models by modifying specific semantic attributes. Feature decomposition based on Sparse AutoEncoder facilitates the selective enhancement or suppression of dimensions, providing dynamic control over retrieval outputs. In contrast to traditional methods that require model retraining, steering offers a flexible and efficient approach to refine search results while preserving model integrity.

We propose a steering-based method for correcting false retrievals through direct user intervention. Unlike static pretrained embeddings, our approach enables the dynamic adaptation of retrieval results to better align with evolving search needs. The steering method based on Sparse AutoEncoder provides a powerful tool for interactive control, enhancing accuracy and personalization in search systems.

This approach improves transparency and controllability, making embedding models more adaptable for applications such as search engines, retrieval-augmented generation (RAG) systems, and medical information retrieval.

Key Contributions

- We propose a dynamic method for embedding correction using steering vectors.
- We establish the theoretical foundation for steering-based false retrieval correction.
- We empirically validate the effectiveness of SAE-based steering for retrieval correction.

2 Theoretical Analysis

"Can a steered embedding still function as a valid embedding?"

Before proposing that embedding models can be adjusted via steering, it is essential to address this core question. In this section, we provide a theoretical foundation for separating specific semantic axes within the embedding space using SAEs, and demonstrate how linear steering can be employed to effectively manipulate these semantic axes. Detailed proofs are provided in the Appendix [1].

This paper is based on three fundamental theoretical components:

1. Assumption of Linear Superposition in Embedding Space. (Section 2.1)
2. Proof of Semantic Disentanglement via Sparsity. (Section 2.2)
3. Linearity of Steering Operation. (Section 2.3)

[1]Each proof is discussed in detail in the Appendix.

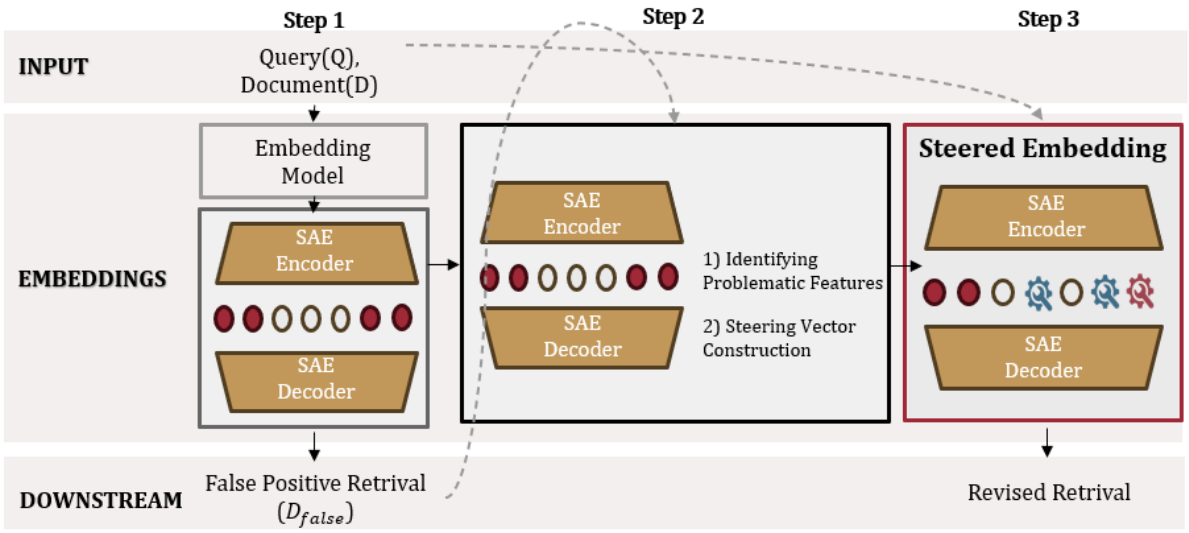

Figure 3: **Overall framework of the proposed correction method** step 1. After generating the embedding vector, it is transformed into interpretable representations using a Sparse AutoEncoder (SAE), and false positive retrieval cases are identified based on the results. step 2. 1) Identify the problematic features that cause false positive retrieval 2) Based on these features, steering vector(s) are constructed. step 3. the final embedding is reconstructed by adjusting steering vector(s).

2.1 Superposition of Semantic Features

Word embedding techniques such as Word2Vec (Mikolov, 2013), GloVe (Pennington et al., 2014), BERT (Devlin, 2018), and Nomic Embed (Nussbaum et al., 2024) are grounded in the distributional hypothesis, which asserts that "words or documents in similar contexts exhibit similar meanings.". These models map a word (or phrase, document) x to a d-dimensional vector $\mathbf{v} = f(x) \in \mathbb{R}^d$.

Definition 1 (Linear Superposition in Embedding Space) The embedding space $\mathbb{R}^d$ formed by the embedding function $f : \mathcal{X} \to \mathbb{R}^d$ is hypothesized to be represented as a linear combination of m latent semantic axes $\{\mathbf{u}_1, \ldots, \mathbf{u}_m\} \subset \mathbb{R}^d$. This means that any embedding vector $\mathbf{v} = f(x)$ can be expressed as:

$$\mathbf{v} \approx \sum_{j=1}^{m} \alpha_j \mathbf{u}_j, \tag{1}$$

where α_j denotes the weight or contribution associated with each corresponding semantic axis.

Lemma 1 (Existence of Semantic Axes) A sufficiently trained embedding model can account for the similarity between word (or phrase, document) vectors as a linear combination of multiple latent semantic axes.

As stated in Lemma 1, the structure of the embedding space, which is conceived as a superposition of multiple semantic representations, serves as the theoretical basis for the sparsity-driven disentanglement methodology outlined in Section 2.2. The formal proof of this lemma is provided in the Appendix.

2.2 Semantic Disentanglement Using Sparse AutoEncoder (SAEs)

As the embedding space consists of multiple interwoven (superposed) semantic axes, it is challenging to isolate the meaning represented by each individual axis. To mitigate this issue, we employ a SAEs, which autonomously disentangles these semantic axes.

Proposition 1 (Semantic Disentanglement via SAE) SAEs with sparsity constraints and an overcomplete structure effectively decompose overlapping semantic representations within the embedding space into components that interfere minimally.

This approach facilitates the learning of latent representations $\mathbf{z}$, which correspond to distinct semantic axes. The proof of Proposition 1 is provided in the Appendix.

2.3 Linear Steering

Assuming SAEs have learned a representation in which each hidden unit corresponds to a distinct semantic axis, we now focus on manipulation (steer-

ing) these semantic axes within the embedding vector.

Theorem 1 (Linearity of Steering under Linear SAE) Given that both the encoder E and decoder D are linear transformations ($E(v) = W_E v$, $D(z) = W_D z$), the steering operation results in a linear transformation of the embedding vector $f(x)$.

The proof (provided in the Appendix) demonstrates that the steering operation, element-wise multiplication in the latent space ($\mathbf{z} \odot \mathbf{s}$), followed by a linear decoding, is equivalent to a linear transformation applied to the original embedding. This linearity ensures that the steered embedding remains within the valid embedding space and that applying the steering vector results in predictable and proportional changes to the embedding. This is integral for maintaining the semantic coherence of the embedding.

Conclusion of Theoretical Analysis This analysis validates the feasibility of embedding steering using SAEs grounded in three primary principles. First, the linear superposition assumption demonstrates that embeddings can be represented as a linear combination of independent semantic axes, providing the foundation for semantic disentanglement. Next, we prove that overcomplete and sparse SAE can effectively separate these axes into interpretable units. Finally, we demonstrate that the steering operation maintains linearity within a linear SAE, ensuring the steered embedding remains valid and semantically coherent. In conclusion, SAEs facilitate intuitive and interpretable embedding adjustments by manipulating the steering vector, which represents semantic axes in the latent space.

3 Interpretable Steering for Controllable Retrieval

We introduce a novel steering method that uses interpretable features from SAEs to enhance interpretability of embedding model and correct retrieval errors, overcoming the limitations of current retrieval systems. The proposed methodology comprises three main steps: (1) diagnosing retrieval failures to analyze problematic features, (2) adjusting embeddings via interpretable feature steering, and (3) performing search tasks using the adjusted embeddings. The overall workflow of the proposed method is illustrated in Figure 3.

3.1 Diagnosing Retrieval Failures for Targeted Correction

First, we encode queries using an existing embedding model (e.g., Nomic Embedding (Nussbaum et al., 2024)) to detect false retrievals and analyze problematic features. We employ a triplet dataset $\mathcal{T} = \{(\mathbf{q}_i, \mathbf{p}_i, \mathbf{n}_i)\}_{i=1}^{N}$, where $\mathbf{q}_i$ represents a query, $\mathbf{p}_i$ denotes the correct (positive) document, and $\mathbf{n}_i$ refers to the incorrect (negative) document, with N denoting the total number of triplets.

Each query-document pair is embedded into a vector space via the embedding model f: $\mathbf{v}_{q_i} = f(\mathbf{q}_i)$, $\mathbf{v}_{p_i} = f(\mathbf{p}_i)$, $\mathbf{v}_{n_i} = f(\mathbf{n}_i)$. The similarity between the query and document embeddings is then computed using cosine similarity:

$$score(\mathbf{q}_i, \mathbf{d}) = \cos(\mathbf{v}_{q_i}, \mathbf{v}_d) = \frac{\mathbf{v}_{q_i} \cdot \mathbf{v}_d}{\|\mathbf{v}_{q_i}\| \|\mathbf{v}_d\|}, \quad (2)$$

where $\mathbf{d}$ can be either $\mathbf{p}_i$ or $\mathbf{n}_i$. A false retrieval occurs if:

$$score(\mathbf{q}_i, \mathbf{n}_i) > score(\mathbf{q}_i, \mathbf{p}_i) \quad (3)$$

We collect instances of false retrieval set $\mathcal{D}_{false}$, which serves as the primary target for embedding adjustments in subsequent steps.

3.2 Embedding Adjustment via Interpretable Feature Steering

In this step, we analyze the causes of false retrieval instances within $\mathcal{D}_{false}$ in the interpretable feature space of SAEs, enabling precise and targeted adjustments to the embeddings. Since each hidden unit of SAEs corresponds to a distinct semantic feature (Corollary 1), we identify and isolate the problematic features contributing to retrieval errors.

3.2.1 Identifying Problematic Features

To identify problematic features, we analyze the activation values of SAEs' hidden units for each query, positive document, and negative document. For each triplet $(\mathbf{q}, \mathbf{p}, \mathbf{n}) \in \mathcal{D}_{false}$, embeddings are generated through embedding model f and encoded via SAE encoder E: $\mathbf{z}_q = E(f(\mathbf{q}))$, $\mathbf{z}_p = E(f(\mathbf{p}))$, $\mathbf{z}_n = E(f(\mathbf{n}))$.

For each hidden unit j, if $z_{qj} \neq 0$ and $z_{nj} > z_{pj}$, the feature is considered to contribute to the false retrieval. Consequently, the set of problematic features $\mathcal{J}_{q,p,n}$ is identified for each triplet.

	Example Passage
Query	the most important factor that influences k+ secretion is _____.
Positive	Internal K+ balance regulates K+ distribution between intracellular and extracellular spaces, mainly controlled by insulin and catecholamines.
Negative	Cholecystokinin and secretin regulate bile secretion and flow, responding to gallbladder movement and acid in the duodenum.

Table 1: The example of MS MARCO triplet dataset

# of revision	retrieval accuracy (%)	Consistency (%)
0	95.4	-
1	93.26	96.02
2	92.87	94.45
3	92.16	93.21
4	91.59*	91.68

Table 2: The performance of MS MARCO triple dataset by the number of revisions.

Steering rate	retrieval accuracy (%)	Consistency (%)
1/3	90.49	90.78
1/2	91.19	92.31
1	93.26	96.02
2	91.42	92.24
3	90.20	91.28

Table 3: The performance of MS MARCO triple dataset based on the rate of steering.

3.2.2 Construction of the Steering Vector

Based on the identified problematic features, a steering vector $\mathbf{s}$ is constructed to adjust embeddings while preserving useful features. For a given triplet $(\mathbf{q}, \mathbf{p}, \mathbf{n})$, the elements s_j of $\mathbf{s}$ are determined as follows:

- If $j \in \mathcal{J}_{q,p,n}$ and $z_{pj} > z_{nj}$, reinforcing relevant features: $s_j \geq 1$ such that $\cos(D(\mathbf{z}_q \odot \mathbf{s}), D(\mathbf{z}_n \odot \mathbf{s})) < \cos(D(\mathbf{z}_q \odot \mathbf{s}), D(\mathbf{z}_p \odot \mathbf{s}))$.
- If $j \in \mathcal{J}_{q,p,n}$ and $z_{nj} > z_{pj}$, weakening misleading features: $0 \leq s_j \leq 1$ such that $\cos(D(\mathbf{z}_q \odot \mathbf{s}), D(\mathbf{z}_p \odot \mathbf{s})) > \cos(D(\mathbf{z}_q \odot \mathbf{s}), D(\mathbf{z}_n \odot \mathbf{s}))$.
- Otherwise, if $s_j = 1$, no adjustment is needed.

To derive a practical steering vector, a steering rate λ is introduced, where $s_j = \lambda$ for reinforcing relevant features and $s_j = 1/\lambda$ for suppressing misleading features. The optimal λ is selected to minimize embedding distortion while effectively addressing false retrievals.

3.3 Downstream Task Using Adjusted Embeddings

The final adjusted embeddings are obtained by applying the steering vector to the latent representation:

$$\mathbf{z}' = \mathbf{z} \odot \mathbf{s}. \tag{4}$$

The adjusted latent representation $\mathbf{z}'$ is then passed through the SAE's decoder D to produce the steered embedding $\mathbf{v}_{steered}$:

$$\mathbf{v}_{steered} = D(\mathbf{z}') = D(\mathbf{z} \odot \mathbf{s}). \tag{5}$$

By applying this method, we ensure that the cosine similarity condition $\cos(D(\mathbf{z}_q \odot \mathbf{s}), D(\mathbf{z}_p \odot \mathbf{s})) > \cos(D(\mathbf{z}_q \odot \mathbf{s}), D(\mathbf{z}_n \odot \mathbf{s}))$ is satisfied. This ensures that false retrievals are corrected effectively without the need for additional training or other burdensome tasks.

4 Experiments and Evaluation

In this section, we identify False retrieval cases in the MS-MARCO triplet dataset (Bajaj et al., 2016) with the steering technique based on SAEs, and analyze the effect of correcting these cases on retrieval performance. Additionally, We evaluate whether the model's generalization capability is preserved by the Multi-Task Embedding Benchmark (MTEB) (Muennighoff et al., 2022).

4.1 Experiment Settings

Model We use the pretrained Nomic Embed model (Nussbaum et al., 2024) for initial text embeddings. A SAE also pretrained on over 420,000 Nomic Embed outputs from scientific paper abstracts in the computer science and astronomy domains (O'Neill et al., 2024).

Datasets We utilize a 10 million instance triplet corpus (query, positive, negative) from the MS MARCO dataset, which is Bing search query logs (Bajaj et al., 2016). An example is presented in

Category →	Cls.	Clust.	PairCls.	Rerank	Retr.	STS	Avg
nomic-embed	74.1	43.9	85.2	55.7	52.8	82.1	65.63
+sae	55.78	**23.75**	76.16	44.02	**17.64**	**67.63**	**47.50**
+steering	52.21	23.34	**78.21**	**44.38**	10.65	64.74	45.59
+over revised	**56.47**	21.93	59.92	40.08	8.40	59.67	41.08

Table 4: The performance comparison on general task (MTEB). The highest performance across the different steering levels (SAE, steering, and over steering), excluding the Nomic-Embed model, is highlighted in bold.

Table 1. Our experiments identify and correct false retrievals, where a negative instance n is retrieved instead of the correct positive instance p for a query q.

4.2 The effects of Steering-based Correction

As shown in Table 2, applying a single revision (steering one problematic feature) results in a retrieval accuracy of 93.26% and a consistency of 96.02%. This demonstrates that a single, targeted steering operation can effectively correct false retrievals while maintaining, and even slightly improving, the consistency of the embedding space. Increasing the number of revisions leads to a gradual decrease in retrieval accuracy, suggesting that excessive steering can begin to distort the original embedding, although consistency mostly remains above 90%. This indicates a trade-off between the number of corrections and overall embedding quality.

Table 3 explores the effect of the steering rate, λ. A steering rate of 1 (equivalent to one revision, as shown in Table 2) achieves the highest retrieval accuracy (93.28%) and consistency (96.02%). Both weaker ($1/3$, $1/2$) and stronger ($\lambda > 1$) steering rates lead to a decrease in performance. This suggests that a moderate steering strength, accurately reflecting the identified problematic feature's influence, is optimal. Overly aggressive steering ($\lambda = 2$ or 3) can negatively impact the embedding, while overly weak steering may not sufficiently correct the retrieval error.

From these results, we conclude that interpretable steering, applied judiciously with an appropriate steering rate ($\lambda = 1$ in our experiments) and a limited number of revisions (optimally 1, and up to 2 while maintaining acceptable performance), can effectively correct false retrievals while preserving the overall performance and consistency of the embedding model. This highlights the potential of our method for controllable retrieval.

4.3 Overall Performance on Benchmark

Multi-Task Embedding Benchmark (MTEB) The experimental results show that the *sae(unrevised)* model achieved an average score of 47.50, while the *revised* model scored 45.59, demonstrating that the steering technique, when applied within a limited range, does not significantly degrade generalization performance.

However, when the number of corrections exceeded four (over-revised state), the average score dropped to 41.08, indicating that excessive corrections can negatively impact model performance.

These findings highlight the importance of predefining an acceptable number or range of corrections, ensuring that false retrievals can be effectively corrected while maintaining the model's overall performance.

5 Related Work

5.1 Embedding based Retrieval

Word representations have evolved from the initial one-hot encoding to more advanced, context-based models such as Word2Vec (Mikolov, 2013), GloVe (Pennington et al., 2014), BERT (Devlin, 2018), and Nomic embed (Nussbaum et al., 2024). These models capture semantic relationships by mapping them into high-dimensional vector spaces, forming the foundation for embedding-based retrieval techniques (Chang et al., 2020; Gao et al., 2022; Lee et al., 2024). However, these models often face significant challenges, including low interpretability and the potential for distorted search results caused by factors such as domain-specific biases or negative sample bias (Subramanian et al., 2018). To address these issues, various approaches have been proposed, including the analysis of internal model structures (Mikolov, 2013; Liu et al., 2021) and the enhancement of semantic feature extraction using domain-specific datasets (Reimers, 2019; Lee et al., 2020). Nevertheless, the inherent structural limitations of the embedding models themselves remain a significant challenge. This paper proposes a novel

Category	#	Feature Explanation
Positive	1842	Chemical, physiological, and industrial applications of Potassium (K) and Potassium Hydroxide (KOH)
	17469	General information, technology, science, research, space, democracy, research institutions, industrial trends in various fields
	3596	Physics (especially mechanics, kinematics, and the law of motion) & logical and scientific approaches
Negative	20887	Physiology, metabolism, blood & immune system, endocrinology, nutrition, brain & fluid secretion
	3468	Physiology, metabolism, blood and immune system, endocrinology, brain & fluid secretion
	598	Military, aerospace, rocket launches, satellites, engineering data, military history, scientific research

Table 5: **Qualitative analysis on generated explanations.** Regarding the dataset used as an example in Table 1, the identified features and interpretation through auto-interp (Paulo et al., 2024)

technique to mitigate biased retrieval results caused by these limitations while preserving the baseline model's performance by dynamically adjusting its behavior during the inference phase.

5.2 Mechanical Interpretability and Steering

According to the Superposition hypothesis (Park et al., 2023; Nelson Elhage†, 2021), Neural network embeddings often intertwine multiple semantic meanings, leading to polysemanticity, which complicates their interpretability (Bolukbasi et al., 2021). By employing Sparse Autoencoder (SAEs) (Ng, 2011), latent semantic structures within embeddings can be effectively disentangled, transforming them into more interpretable components. This approach has been extensively applied across various domains, from language models to multimodal frameworks such as CLIP (Cunningham et al., 2023; Adly Templeton*, 2024; Daujotas, 2024). In particular, research have demonstrated that it can significantly enhance the interpretability of embedding models to decompose the multifaceted meanings embedded in dense representations into sparse spaces using SAEs can significantly enhance the interpretability of embedding models(O'Neill et al., 2024; Han et al., 2024). Furthermore, SAEs have proven effective in enabling interpretable steering functions, offering the ability to modulate model behavior. We apply SAE-based interpretability techniques to text embeddings, leveraging interpretable steering to mitigate search bias in retrieval models.

6 Discussion

This study applies an SAEs-based steering technique to mitigate false retrievals using the MS-MARCO Triplet dataset (Bajaj et al., 2016). We analyzed the impact of our proposed SAEs-based steering technique on model performance and generalization, and evaluated whether modifying common features between the Positive and Query improved retrieval accuracy.

6.1 The Effectiveness of the Steering Technique

The experimental results demonstrate that applying corrections within a defined range effectively mitigates retrieval errors in the existing retrieval model (Table 2, Table 3). Furthermore, these corrections do not significantly affect overall retrieval performance or domain generalization (Table 4).

Visualization of embedding changes (Figure 4) shows that it enhances a specific feature through steering by increasing the distance between the Query and Positive, which improves their cosine similarity and increases the likelihood of retrieving the correct Positive. On the other hand, weakening a feature decreases the distance between the Query and Negative, reducing their similarity and preventing the retrieval of the incorrect Negative.

6.2 Feature Analysis

The analysis revealed that feature number 1842 in Table 5 had the most significant impact on false retrieval, as it was closely related to the positive information associated with the query. Therefore,

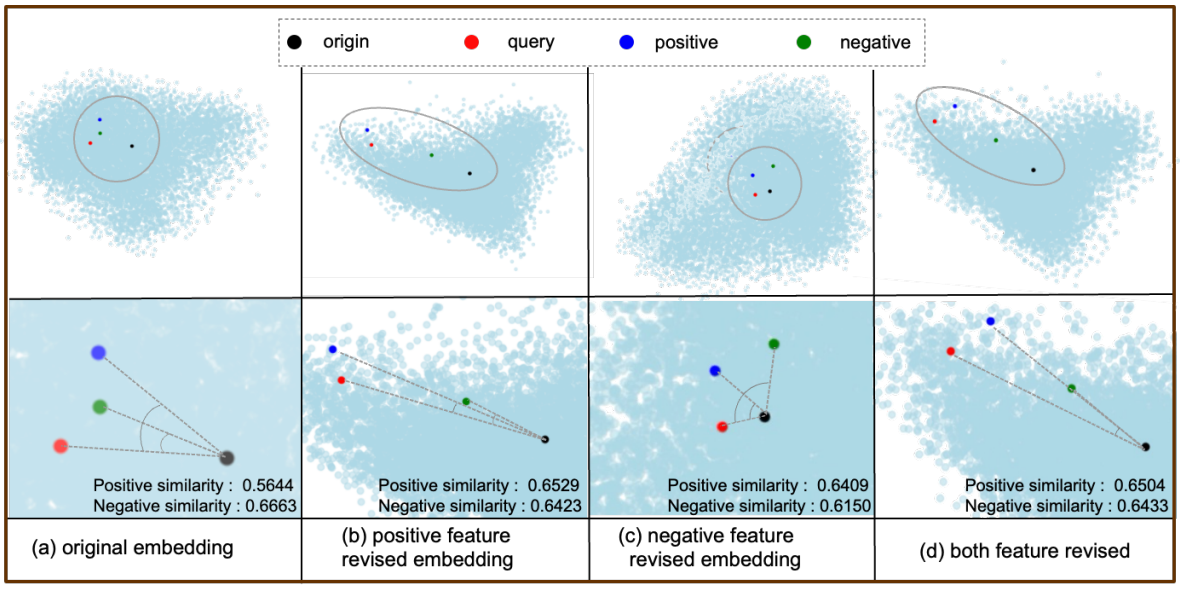

Figure 4: **Visualization of embedding space adjustments with interpretable steering techniques using PCA** (a) In the original embedding prior to steering, the negative correlation outweighs the positive correlation. (b) After strengthening the positive feature for correction, both the positive and query move farther from the origin, resulting in an increase in positive similarity. (c) When the negative feature is weakened, the negative feature and query become closer, resulting in a reduction of negative similarity. (d) Both features are adjusted through steering.

we conclude that further research is needed to refine feature selection and optimize the steering vector for more precise corrections beyond the method proposed in this study.

6.3 Practical Implications

The proposed steering technique demonstrates the potential to correct erroneous retrieval results caused by model bias without further training. By analyzing the features, we can identify those that directly contribute to false retrievals and apply steering with the appropriate intensity, allowing for more accurate corrections.

However, a key limitation of this study is the lack of an established objective criterion for determining the intensity and selection of features for steering. Additionally, further research is required to analyze the long-term effects of feature manipulation on overall retrieval performance. Future studies should focus on expanding the application scope of the steering technique and systematically establishing optimal feature manipulation methods to enhance retrieval correction capabilities.

7 Conclusion

In this work, we introduces a new approach to controlling embedding models using an interpretable steering technique based on SAEs. Experimental results demonstrate that this approach effectively mitigates false retrieval problems while maintaining the generalization performance of the model. Notably, the SAE-based correction method offers a practical advantage by enabling localized corrections of erroneous retrieval results without requiring a complete retraining of the existing model. This approach allows for error correction with relatively low computational cost and minimal data, even in situations where retraining large-scale models is not feasible, thereby contributing to the development of more efficient retrieval systems.

8 Limitation

We demonstrate the effectiveness of our steering approach, based on SAEs, on an embedding model using the MS MARCO dataset. However, we observed that performance decreased when the corrections were excessive or too frequent. This can be attributed to the following limitations:

Limitations of SAE The Sparse Autoencoder (SAE) model, due to its autoencoder architecture, is inherently limited in its ability to fully reconstruct the original input, which may lead to performance degradation, as shown in Table 4.

Ideally, the features should be mono-semantic, where only the features corresponding to the desired semantic axis should be activated and rectified. However, due to polysemanticity, some fea-

tures may be entangled with unrelated information. This suggests that the proposed correction method could introduce potential side effects.

As observed in Section 4, excessive corrections resulted in side effects. To mitigate this issue, the SAE needs to better decompose the embedding representation into more semantically meaningful components. However, there are currently technical limitations in achieving this.

Problematic Features and Steering Rate In this work, we propose the minimal correction value required to address erroneous retrievals, and we correct both the associated positive and negative features. However, there exists an optimal correction value that aligns with the user's intended criteria (which is subjective and qualitative), and as mentioned in 6.2, the feature that the user perceives as causing the error may represent only a subset of the overall positive and negative features.

As done in this study, steering all related features may be inefficient. However, identifying and correcting only the semantically relevant features requires an analytical approach, and since the model's semantic perception often differs from that of humans, this remains a key limitation.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jonathan Marcus Jack Lindsey Trenton Bricken Brian Chen Adam Pearce Craig Citro Emmanuel Ameisen Andy Jones Hoagy Cunningham Nicholas L Turner Callum McDougall Monte MacDiarmid Alex Tamkin Esin Durmus Tristan Hume Francesco Mosconi C. Daniel Freeman Theodore R. Sumers Edward Rees Joshua Batson Adam Jermyn Shan Carter Chris Olah Tom Henighan Adly Templeton*, Tom Conerly*. 2024. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. Anthropic Blog.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Tolga Bolukbasi, Adam Pearce, Ann Yuan, Andy Coenen, Emily Reif, Fernanda Viégas, and Martin Wattenberg. 2021. An interpretability illusion for bert. *arXiv preprint arXiv:2104.07143*.

Wei-Cheng Chang, Felix X Yu, Yin-Wen Chang, Yiming Yang, and Sanjiv Kumar. 2020. Pre-training tasks for embedding-based large-scale retrieval. *arXiv preprint arXiv:2002.03932*.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2023. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.

Gytis Daujotas. 2024. Case study: Interpreting, manipulating, and controlling clip with sparse autoencoders.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022. Precise zero-shot dense retrieval without relevance labels. *arXiv preprint arXiv:2212.10496*.

Yufei Guo, Muzhe Guo, Juntao Su, Zhou Yang, Mengqiu Zhu, Hongfei Li, Mengyang Qiu, and Shuo Shuo Liu. 2024. Bias in large language models: Origin, evaluation, and mitigation. *arXiv preprint arXiv:2411.10915*.

Chi Han, Jialiang Xu, Manling Li, Yi Fung, Chenkai Sun, Nan Jiang, Tarek Abdelzaher, and Heng Ji. 2024. Word embeddings are steers for language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16410–16430.

Gautier Izacard and Edouard Grave. 2020. Distilling knowledge from reader to retriever for question answering. *arXiv preprint arXiv:2012.04584*.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Svetlana Kiritchenko and Saif M Mohammad. 2018. Examining gender and race bias in two hundred sentiment analysis systems. *arXiv preprint arXiv:1805.04508*.

Jinhyuk Lee, Zhuyun Dai, Xiaoqi Ren, Blair Chen, Daniel Cer, Jeremy R Cole, Kai Hui, Michael Boratko, Rajvi Kapadia, Wen Ding, et al. 2024. Gecko: Versatile text embeddings distilled from large language models. *arXiv preprint arXiv:2403.20327*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. 2021. Dexperts: Decoding-time controlled text generation with experts and anti-experts. *arXiv preprint arXiv:2105.03023*.

Tomas Mikolov. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 3781.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2022. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*.

Catherine Olsson Tom Henighan† Nicholas Joseph† Ben Mann† Amanda Askell Yuntao Bai Anna Chen Tom Conerly Nova DasSarma Dawn Drain Deep Ganguli Zac Hatfield-Dodds Danny Hernandez Andy Jones Jackson Kernion Liane Lovitt Kamal Ndousse Dario Amodei Tom Brown Jack Clark Jared Kaplan Sam McCandlish Chris Olah‡ Nelson Elhage†, Neel Nanda. 2021. A mathematical framework for transformer circuits.

A. Ng. 2011. Sparse autoencoder. `http://web.stanford.edu/class/cs294a/sparseAutoencoder.pdf`. CS294A Lecture notes.

Zach Nussbaum, John X Morris, Brandon Duderstadt, and Andriy Mulyar. 2024. Nomic embed: Training a reproducible long context text embedder. *arXiv preprint arXiv:2402.01613*.

Charles O'Neill, Christine Ye, Kartheik Iyer, and John F. Wu. 2024. Disentangling dense embeddings with sparse autoencoders. *Preprint*, arXiv:2408.00657.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2023. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*.

Gonçalo Paulo, Alex Mallen, Caden Juang, and Nora Belrose. 2024. Automatically interpreting millions of features in large language models. *Preprint*, arXiv:2410.13928.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*.

Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. 2018. Spine: Sparse interpretable neural embeddings. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. 2019. Mitigating gender bias in natural language processing: Literature review. *arXiv preprint arXiv:1906.08976*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.