Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Investigating the Effect and Mechanism of Semantic Alignment in Fixed-Backbone Protein Sequence Design

Ke Zhang

kzhang23@tju.edu.cn

Tianjin University

Research Article

Keywords:

Posted Date: July 12th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10307155/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Abstract

AGSDD proposed semantic alignment (SA) as a way to incorporate amino-acid-level semantic information into diffusion-based inverse folding, based on the idea that amino-acid types are not merely discrete class indices but may also encode biochemical, functional, and evolutionary regularities. However, it remains unclear whether this idea can be transferred beyond the original AGSDD framework, and the output-level effect of SA remains underexplored. In this study, we use MapDiff as a case study to examine the transferability of AGSDD-inspired SA and analyze its effect on the final amino-acid probability distribution. AGSDD-inspired SA can be integrated into MapDiff and modestly improves perplexity while leaving median recovery largely unchanged. The AGSDD-style increase in true-amino-acid semantic attention is reproducible, but output-level analyses suggest that SA mainly behaves as a distributional regularizer or smoother. Current NSSR and BLOSUM-based probability-mass analyses do not provide clear evidence that SA enhances BLOSUM-defined biochemical similarity.

1 Introduction

Proteins play a crucial role in biological systems, with functions including chemical catalysis, immune responses, regulation of metabolism, regulation of gene expression, structural support for tissues and bones. Functions vary across different protein structures, which makes protein structure design and sequence design indispensable and promising in drug discovery, materials science, and other related fields[1]. Protein structure prediction from amino-acid sequences, often associated with the protein folding problem, has achieved considerable progress in recent years. Models such as AlphaFold2 (AF2) [2], RoseTTAFold[3], ESMFold[4] can achieve near-experimental accuracy in protein structure prediction from amino-acid sequences—with or without multiple sequence alignment (MSA) information. However, fixed-backbone protein sequence design, commonly formulated as inverse protein folding (IPF) in recent studies, aims to predict amino-acid sequences conditioned on known protein structures or backbone geometries and remains an active area for further development.

Traditional physics-based methods, such as EvoEF2[5] and Rosetta[6], treat IPF as an energy optimization problem, but often incur high computational costs and show limited predictive performance. In recent years, owing to the outstanding capability to learn high dimensional and nonlinear patterns, deep learning has emerged as an alternative with better accuracy and shorter computation time[7]. Among deep learning methods for IPF, graph-based methods are commonly used, with graph neural networks (GNNs) or GNN-like architectures serving as core components[8]. For example, ProteinMPNN adopted an encoder–autoregressive-decoder architecture based on message passing neural networks and achieved outstanding performance in both in silico and experimental tests[7]. Later, PiFold used an enhanced residue featurizer and replaced autoregressive decoder with more stacked PiGNN layers, achieving state-of-the-art recovery rates and largely improved efficiency[9]. GRADE-IF then employed an equivariant GNN (EGNN) as the denoising network in a diffusion model, which further elevated the prediction accuracy[10]. MapDiff adopted a mask-prior-guided denoising diffusion framework[8]. In its diffusion network, an EGNN first denoises the noisy sequence and provides

a basic predicted sequence, and then a pretrained masked sequence designer (an invariant-point-attention-based network, named IPAN below) refined the basic predicted sequence on entropy-masked positions, which substantially outperformed state-of-the-art methods.

Despite the continuously developing architectures, most backbone-conditioned inverse folding models primarily condition on backbone geometry and do not explicitly model surface, ligand, or other biochemical contexts, which are also important for designing a biologically functional protein[11,12]. For instance, two proteins with complementary shapes may still not bind due to incompatible polarity or hydrophobicity at their binding interface[13]. To bridge this gap, several studies have tried to incorporate surface biochemistry and reported improvements, such as SurfPro and SurfFold[11,14]. From another perspective, AGSDD suggested that amino-acid types are not merely discrete class indices: they also encode biochemical, functional, and evolutionary regularities and modeled this idea by introducing a semantic alignment (SA) module for diffusion-based inverse folding[15]. SA uses a learnable amino-acid type dictionary and attention-based alignment between residue hidden states and the 20 amino-acid type embeddings. AGSDD reports that, as denoising layers become deeper, residues tend to assign increasing attention to their true amino-acid types, which also brings a moderately improved performance.

SA is intended to encourage the hidden states to align with amino-acid-type embeddings, which may act as an amino-acid-level proxy for biochemical similarity or evolutionary substitution regularities[15]. This idea is motivated by the possibility that, even without explicit biochemical or evolutionary input features, models may learn implicit residue-type correlations from protein context, because amino acids with similar biochemical properties or evolutionary substitutability may appear in similar structural environments[16]. Since this module does not require additional structural or biochemical input features, it can be conveniently integrated into other inverse-folding architectures. Even a moderate but stable improvement from such an auxiliary module could be useful for architecture development. However, it remains unclear whether the semantic alignment idea can be directly transferred beyond the original AGSDD framework. In addition, AGSDD reported increasing attention weights assigned to the true amino-acid types across denoising layers, but attention weights alone do not necessarily provide a faithful explanation of model outputs[17]. Therefore, the output-level effect of SA, especially its influence on the final amino-acid probability distribution, remains underexplored.

To bridge this gap, this study examines the transferability of AGSDD-inspired SA to MapDiff as a case study and further explores its mechanism related to output probability distribution. MapDiff shares a partially similar diffusion framework with AGSDD, making it a suitable pilot case for evaluating the transferability of SA—if SA does not work well in AGSDD-inspired-MapDiff, it is less likely that SA can be transferred to other more differentiated architectures. AGSDD-inspired-MapDiff is built on top of a dihedral-standardized MapDiff baseline, where the backbone dihedral feature calculation was standardized before adding semantic modules. In summary, this study's goals include:

1. Evaluate whether SA improves AGSDD-inspired-MapDiff on CATH 4.2 and PDB2022.

2. Test whether the attention alignment phenomenon in AGSDD can be reproduced in AGSDD-inspired-MapDiff.

3. Analyze how SA changes the output probability distribution.

2 Method

2.1 Overall Framework

The framework of AGSDD-inspired-MapDiff is shown in Fig. 1. Consistent with MapDiff[8], AGSDD-inspired-MapDiff uses a discrete diffusion process over amino-acid sequences (Fig. 1.a). During training, the clean sequence is denoted as X_0^{aa}, and X_t^{aa} is the corrupted sequence at diffusion step t. The diffusion process progressively adds random discrete noise to the native sequence X_0^{aa} according to the cumulative transition matrix $\bar{Q}_t$. When denoising, given the protein backbone geometry and X_t^{aa}, the denoising network ϕ predicts the original sequence X_0^{aa}. Given the predicted original sequence $\widehat{X}_0^{aa}$ and X_t^{aa}, a less noisy sequence X_{t-1}^{aa} can be sampled from the computed posterior distribution $q\left(X_{t-1}^{aa}|X_t^{aa},\widehat{X}_0^{aa}\right)$. Through iterative denoising steps, X_T^{aa}, which is initially sampled from a prior distribution, is expected to be denoised toward X_0^{aa} after a predefined number of reverse steps.

The denoising network contains two main branches (Fig. 1.b). The EGNN branch uses backbone geometry and the noisy sequence to produce base prediction over the 20 amino-acid types. EGNN contains multiple blocks, where each block includes one message passing layer and one semantic alignment layer. The IPAN branch is pretrained as a mask-prior model. IPAN also includes multiple blocks, where each block includes one IPA layer and one semantic alignment layer. During diffusion training and inference, EGNN first predicts the full sequence. Positions with high EGNN uncertainty are marked as masked positions, and IPAN refines these positions using backbone geometry and the EGNN prediction as context. In each block of both branches, the updated node feature and residue representation are aligned to amino acid type-specific semantics by a SA layer. The SA layer is a cross-attention-based network (Fig. 1.c), which is to be explained later.

2.2 Backbone Dihedral Feature Standardization

Before adding semantic alignment, we first constructed a dihedral-standardized MapDiff baseline by revising several details in the backbone dihedral feature calculation of our reproduced implementation. Specifically, dihedral angles were converted from degrees to radians before applying NumPy's function $np.sin$ and $np.cos$ [18]; the backbone phi-angle was computed using the standard torsion definition $\phi_i = \mathrm{dihedral}\,(C_{i-1}, N_i, CA_i, C_i)$; and peptide-bond torsion features across likely chain breaks were avoided by checking the distance between $C_i$and N_{i+1}, with a conservative threshold of 1.8 Å using in Biopython[19]. Undefined torsions at termini or chain breaks are encoded as ($sin\theta$, $cos\theta$) = (0, 0). AGSDD-inspired-MapDiff was then built on top of this dihedral-standardized baseline.

2.3 Semantic Alignment Module

The semantic alignment module follows the core idea of AGSDD[15] but is adapted to MapDiff's EGNN and IPAN branches. For each SA layer, the input hidden representation (node feature or residue representation) is denoted as $h \in \mathbb{R}^{N\times C}$, where N is the number of residues and C is the hidden dimension. Batch dimensions are omitted for simplicity. All amino acid types are initialized as a learnable amino-acid dictionary: $H_s \in \mathbb{R}^{20\times C}$, which is shared across layers.

The hidden representation is projected to the query space:

$$Q = hW_q^T + b_q,$$

where $W_q \in \mathbb{R}^{C\times C}$ and $Q \in \mathbb{R}^{N\times C}$. The amino-acid dictionary is projected to key and value spaces:

$$K = H_s W_k^T + b_k, \;\; V = H_s W_v^T + b_v,$$

where $W_k, W_v \in \mathbb{R}^{C\times C}$ and $K, V \in \mathbb{R}^{20\times C}$.The semantic attention logits A and probabilities P then can be computed as:

$$A = \frac{QK^T}{\sqrt{C}}, \;\; P = \mathrm{softmax}\left(A\right).$$

Here $A \in \mathbb{R}^{N\times 20}$ and $P \in \mathbb{R}^{N\times 20}$. The semantic attention logits are supervised with a cross-entropy loss against the true amino-acid type, encouraging semantic alignment with amino-acid labels.

$$\mathcal{L} = CE\left(A, y_{true}\right)$$

Then, taking the attention probabilities as weights, the semantic value update is:

$$u = PV.$$

where $u \in \mathbb{R}^{N\times C}$. AGSDD-inspired-MapDiff adopts a residual update rather than directly replacing the hidden representation with the attention-weighted semantic value, following a similar principle of AGSDD:

$$g = sigmoid\left(\left[h; u\right] W_g^T + b_g\right), \;\; \Delta = \mathrm{MLP}\left(\left[h; u\right]\right),$$

$$\text{where } \left[h; u\right] \in \mathbb{R}^{N\times 2C},\; W_g \in \mathbb{R}^{C\times 2C},\; \text{and } g, \Delta \in \mathbb{R}^{N\times C}.$$

$$h_{out} = h + g \odot \Delta\,.$$

The output h_{out} has shape $\mathbb{R}^{N\times C}$.

2.4 Embed SA in MapDiff

AGSDD-inspired-MapDiff inserts SA into each layer in EGNN and IPAN, even though AGSDD does not contain a unit like IPAN. In EGNN, SA is applied to the node feature after each message-passing layer, while coordinate features and edge features are not directly modified by SA. In IPAN, semantic alignment is applied after each IPA layer.

For EGNN, the semantic loss is computed over all valid residues and averaged across layers:

$$\mathcal{L}_{sem}^{EGNN} = \frac{1}{L_E} \sum_{l=1}^{L_E} \mathrm{CE}\left(A_l^{EGNN}, y_{true}\right),$$

where L_E is the number of EGNN layers, A_l^{EGNN} is the semantic attention logits at layer l, and y_{true} is the true amino-acid label.

For IPAN, the semantic loss is computed on masked positions, consistent with the IPA prediction loss in MapDiff:

$$\mathcal{L}_{sem}^{IPA} = \frac{1}{L_I} \sum_{l=1}^{L_I} \mathrm{CE}\left(A_l^{IPA}[M], y[M]\right),$$

where M denotes the EGNN uncertainty mask. The total training loss is:

$$\mathcal{L}_{total} = \mathcal{L}_{EGNN} + \mathcal{L}_{IPA} + \lambda \left(\mathcal{L}_{sem}^{EGNN} + \mathcal{L}_{sem}^{IPA}\right),$$

where λ denotes the semantic loss weight. The final prediction and inference pipeline remain the same as MapDiff: EGNN and IPAN logits are fused by the entropy-based fusion rule. The semantic attention logits are used for auxiliary supervision and hidden-state updates, not as an additional third prediction branch.

3 Experiments

3.1 Dataset and Evaluation Metrics

Experiments are conducted on CATH 4.2 [20] using the same split and evaluation protocol as MapDiff: this dataset is split based on their CATH topology classification codes, consisting of 18,024, 608 and 1,120 structures in train, validation and test set, respectively. The main metrics are median recovery rate and perplexity. Recovery rate quantifies the accuracy of model predictions compared to the ground-truth amino-acid sequence, and perplexity measures the uncertainty of model's predictions.

In addition, zero-shot transferability is evaluated in PDB2022, an external test set composed of single-chain protein structures released in the Protein Data Bank in 2022[21, 22]. Because these structures were released after the construction of the CATH 4.2 benchmark, PDB2022 provides a time-split external evaluation setting.

For mechanism analysis of SA, additional metrics include:

- Semantic attention weights assigned to the true amino-acid type,
- Rank of the true amino-acid type among the 20 semantic attention weights,
- Mean output probability assigned to the true amino-acid type,
- Fraction of residues where the true amino-acid type appears in the top-k predicted classes,
- Probability difference between the probability of the true amino acid and the highest probability among incorrect amino acids,
- Native sequence similarity recovery (NSSR) 62/90[23], measured by the blocks substitution matrix (BLOSUM) 62/90[24].
- Probability mass assigned to non-native amino-acid types with positive BLOSUM substitution scores relative to the native amino acid.

3.2 Experimental Setting

AGSDD-inspired-MapDiff mainly follows the original MapDiff setting. Before training of the denoising diffusion model, IPAN model was pretrained on the same but masked train set. Experiments were performed on 2 × NVIDIA A100 GPUs using DDP, and batch size on each GPU was set to four. IPAN was pretrained for 200 epochs and the diffusion model was trained for 100 epochs. An Adam optimizer with a one-cycle scheduler was used for parameter optimization, with a peak learning rate of 5×10^{-4}. EGNN and IPAN both consisted of six blocks. The number of attention heads was set to four in IPAN and one in SA. Hidden dimension was 128 in both branches. The dropout rate was set to 0.2, and the sampling setting was the same as MapDiff. A marginal prior distribution was adopted in the denoising process. The semantic loss weight λ was set to 0.5. This follows the spirit of AGSDD, where semantic supervision is trained at a scale comparable to the prediction objective.

3.3 Main Results

To explore the effect of SA, published results of MapDiff[8], local reproduction of MapDiff, dihedral-standardized MapDiff and AGSDD-inspired-MapDiff were compared. As shown in Table 1, dihedral-standardized MapDiff produced generally comparable results on median recovery rate (slight improvement) and perplexity. Compared with dihedral-standardized MapDiff, AGSDD-inspired-MapDiff produced a comparable median recovery rate, with a small relative increase of 0.29%, while reducing perplexity from 3.54 to 3.41, corresponding to a moderate relative reduction of 3.67%. The effect of SA on recovery rate was more conservative than that reported in AGSDD (relative increase 1.49%) [15], and its relative effect on perplexity was numerically larger than that reported in AGSDD (relative reduction 2.33%).

Table 1
Experiment results on the CATH 4.2 test dataset.

Version	Parameters	Median recovery rate (%)	Perplexity
Published results of MapDiff[8]	14.7M	60.93	3.43
Local reproduction of MapDiff	14.7M	60.97	3.54
Dihedral-standardized MapDiff	14.7M	61.23	3.54
AGSDD-inspired-MapDiff (SA & Dihedral-standardized)	16.3M	61.41	3.41

For further comparison, validation performances on the CATH 4.2 validation dataset are shown in Table 2. At epoch 50, AGSDD-inspired-MapDiff is slightly worse than the dihedral-standardized MapDiff on recovery and perplexity. However, at epoch 100, this comparison reverses, and the semantic version becomes better on both metrics. This suggests that the semantic module may require longer optimization before its benefit becomes visible.

Table 2
Validation on the CATH 4.2 validation dataset.

Version	Epoch	Median recovery rate (%)	Perplexity
Local reproduction of MapDiff	50	57.25	3.91
Local reproduction of MapDiff	100	59.69	3.7
Dihedral-standardized MapDiff	50	57.35	3.9
Dihedral-standardized MapDiff	100	59.75	3.73
AGSDD-inspired-MapDiff (SA & Dihedral-standardized)	50	56.78	3.93
AGSDD-inspired-MapDiff (SA & Dihedral-standardized)	100	60.11	3.57

3.4 Ablation Study

This experiment investigated the effect when SA was only embedded in the final layer on EGNN and IPAN. As shown in Table 3, the SA final-only version is weaker than AGSDD-inspired-MapDiff on both recovery and perplexity. This suggests that distributing SA across multiple blocks may be more beneficial than using it only as a final adapter.

Table 3
Ablation study

Version	Median recovery rate (%)	Perplexity
SA Final-only	61.03	3.53
AGSDD-inspired-MapDiff	61.41	3.41

3.5 Mechanism Analysis for SA

The mechanism analysis has two goals. First, it checks whether the AGSDD-style semantic alignment phenomenon can be reproduced in AGSDD-inspired-MapDiff. Second, it investigates the effect of SA on the output probability distribution.

3.5.1 Attention weights on true AAs

AGSDD reports that semantic attention to the true amino-acid type increases as the denoising network becomes deeper[15], which also appears in AGSDD-inspired-MapDiff as shown in Fig. 2. Both EGNN and IPAN show increasing mean attention weights and decreasing mean ranks of attention weights to the true AAs from shallow to deep layers, although the trend is relatively flat in EGNN.

However, this trend may arise from multiple factors, including direct supervision from the semantic loss and richer structure-conditioned information encoded in deeper hidden states. Therefore, attention-level alignment should be interpreted as a diagnostic signal rather than direct evidence of SA's effect on final predictions.

3.5.2 Effects on output probability distribution

Although AGSDD-inspired-MapDiff produced a slight improvement in median recovery compared with Dihedral-standardized MapDiff, the mean predicted probability of the true AAs decreased. Meanwhile, the proportions of true AAs in the top 1 and top 5 predictions and the mean probability rank of the true AAs remained relatively stable as shown in Table 4. In addition, the mean probability difference between true AAs and highest incorrect AAs slightly decreased. It suggests that SA did not simply sharpen the final output distribution on true AAs. Instead, it appears to make the distribution flatter.

In addition, given that SA is intended to guide the model to learn semantic correlations regarding biochemical and evolutionary information, one might expect the model to learn the biological correlation between similar amino acids in high dimensional space and redistribute probability mass from the true AAs to biologically similar alternatives. However, the median NSSR62/90 recoveries and the total probability mass assigned to the true AAs and BLOSUM-similar alternatives were not improved in AGSDD-inspired-MapDiff (Table 4). Thus, there is not enough evidence that SA enhances BLOSUM-defined biochemical similarity or residue-type substitutability. Together with the decreased perplexity in AGSDD-inspired-MapDiff, these results suggest that SA behaves more like a distributional regularizer or

smoother rather than a simple sharpening mechanism. A plausible explanation is that SA redistributes probability mass, potentially reducing overconfident predictions for some residues while improving the log-likelihood of more difficult residues.

Table 4
Changes in output probability distribution

Metrics **(summarized in residue-level)**	**Dihedral-standardized MapDiff** **(ref.)**	**AGSDD-inspired-MapDiff**	**Difference**
Mean predicted probability of true AAs (%)	54.92%	54.23%	**-0.69%**
Proportion of true AAs at top 1 (%)	60.66%	60.82%	0.16%
Proportion of true AAs at top 5 (%)	92.39%	92.39%	0.00%
Mean rank of probability of true AAs	2.15	2.14	-0.01
Mean probability difference between true AAs and highest incorrect AAs (%)	27.39%	27.33%	-0.06%
Median NSSR 62 recovery (%)	80.14%	80.12%	-0.02%
Median NSSR 90 recovery (%)	77.25%	77.33%	0.08%
Mean probability masses of similar AAs (%, BLOSUM 62)	20.32%	20.44%	0.12%
Mean probability masses of similar AAs (%, BLOSUM 90)	17.37%	17.43%	0.06%
Mean probability masses of true and similar AAs (%, BLOSUM 62)	75.26%	74.68%	**-0.58%**
Mean probability masses of true and similar AAs (%, BLOSUM 90)	72.31%	71.67%	**-0.64%**

3.6 Zero-shot generalization

To assess the generalization ability of the effect of SA, AGSDD-inspired-MapDiff was tested on PDB2022. The results are shown in Table 5. Consistent with the results on CATH 4.2, AGSDD-inspired-MapDiff produced a comparable median recovery rate (though slightly decreased) and moderately improved perplexity compared with Dihedral-standardized MapDiff. This suggests that the perplexity-reducing effect of SA may generalize beyond CATH 4.2, although the effect on recovery remains negligible.

Table 5
Experiment results on the PDB2022 dataset.

Version	Median recovery rate (%)	Perplexity
Published results of MapDiff[8]	68.03	Not reported
Local reproduction of MapDiff	67.82	2.84
Dihedral-standardized MapDiff	67.78	2.84
AGSDD-inspired-MapDiff (SA & Dihedral-standardized)	67.74	2.78

4 Discussion

The effect of semantic alignment in AGSDD-inspired-MapDiff can be summarized by four related observations. First, AGSDD-inspired-MapDiff moderately improves perplexity over the dihedral-standardized MapDiff. Second, the semantic attention assigned to the true amino-acid type increases across layers, reproducing the main SA phenomenon reported in AGSDD. Third, the final output distribution is not simply sharpened; instead, it appears to become relatively flatter. Fourth, the median NSSR62/90 recoveries and the total probability mass assigned to the true AAs and BLOSUM-similar alternatives were not improved in AGSDD-inspired-MapDiff. These results suggest that SA acts more like a distributional regularizer or smoother, while there is not enough evidence that SA could enhance BLOSUM-defined biochemical similarity or residue-type substitutability. To facilitate the biological understanding of models, future SA variants may need to incorporate explicit biochemical embeddings or representations.

The more limited recovery gain observed in AGSDD-inspired-MapDiff may be explained by three factors. First, SA is not the dominant module even in AGSDD. In the AGSDD ablation study, semantic alignment contributes as one additional component rather than as a standalone source of large improvement. The effect observed here is more conservative, but remains consistent with SA providing a modest auxiliary contribution. Second, the current best checkpoint appears at the final training epoch, and the validation comparison between the dihedral-standardized MapDiff and AGSDD-inspired-MapDiff changes from unfavorable for SA at epoch 50 to favorable at epoch 100. This suggests that SA may need longer optimization to show its benefit. Furthermore, AGSDD reports 70,000 training steps and a batch size of 32, which corresponds to about 124–125 effective epochs. The current MapDiff experiment is trained for 100 epochs, so the semantic module may not yet be fully optimized under the current computational budget. Third, AGSDD-inspired-MapDiff does not include AGSDD's contextual aggregation (CA) module. In AGSDD, semantic denoising is paired with additional contextual information from the geometric denoising pipeline. In AGSDD-inspired-MapDiff, especially inside EGNN, SA has less global context available when aligning residue hidden states to amino-acid type embeddings. This architectural difference may limit how much benefit SA can provide.

This study has several limitations. First, only a limited set of semantic designs and lambda values were tested. Future experiments could include EGNN-only semantic alignment, IPA-only semantic alignment, additional lambda values such as 0.1, and different adapter initializations. These experiments were not fully explored because of limited cloud GPU resources. Second, the input information of SA in AGSDD-inspired-MapDiff is not completely the same as that in AGSDD because original MapDiff does not include AGSDD's CA module, which may introduce a difference in SA's behavior. Third, because SA increases the parameter count from 14.7M to 16.3M, the current experiments cannot fully separate the effect of semantic alignment from the effect of increased model capacity. Fourth, the experiments were conducted without repeated random seeds because of computational constraints; therefore, small differences in recovery should be interpreted cautiously. Finally, this study evaluates sequence-level compatibility rather than experimental functionality or folding self-consistency; therefore, the observed perplexity improvement should not be interpreted as direct evidence of improved foldability or biological function.

5 Conclusion

AGSDD-inspired semantic alignment can be integrated into MapDiff and modestly improves perplexity while leaving median recovery largely unchanged. The extent of its benefit appears to be architecture-dependent. The AGSDD-style increase in true-amino-acid semantic attention is reproducible, but output-level analyses suggest that SA mainly behaves as a distributional regularizer or smoother. Moreover, the current NSSR and BLOSUM-based probability-mass analyses do not provide clear evidence that SA enhances BLOSUM-defined biochemical similarity.

Declarations

Declaration of Generative AI and AI-assisted technologies in the writing process

Declaration of Generative AI and AI-assisted technologies in the writing process
During the preparation of this work the author used ChatGPT in order to improve clarity and succinctness. After using this tool/service, the author reviewed and edited the content as needed and takes full responsibility for the content of the publication.

Code availability

The source code, checkpoints and implementation details of AGSDD-inspired-MapDiff are available via GitHub at https://github.com/Zhangke123jimu/AGSDD-inspired-MapDiff.

References

1. Huang PS, Boyken SE, Baker D (2016) The coming of age of de novo protein design. Nature 537(7620):320–327. 10.1038/nature19946
2. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O et al (2021) Highly accurate protein structure prediction with AlphaFold. Nature 596(7873):583–589. 10.1038/s41586-021-03819-2
3. Baek M, DiMaio F, Anishchenko I, Dauparas J, Ovchinnikov S, Lee GR et al (2021) Accurate prediction of protein structures and interactions using a three-track neural network. Science 373(6557):871–876. 10.1126/science.abj8754
4. Lin Z, Akin H, Rao R, Hie B, Zhu Z, Lu W et al (2023) Evolutionary-scale prediction of atomic-level protein structure with a language model. Science 379(6637):1123–1130. 10.1126/science.ade2574
5. Huang X, Pearce R, Zhang Y (2020) EvoEF2: accurate and fast energy function for computational protein design. Bioinformatics 36(4):1135–1142. 10.1093/bioinformatics/btz740
6. Das R, Baker D (2008) Macromolecular Modeling with Rosetta. Annual Review of Biochemistry. ;77(Volume 77, 2008):363–82. 10.1146/annurev.biochem.77.062906.171838
7. Dauparas J, Anishchenko I, Bennett N, Bai H, Ragotte RJ, Milles LF et al (2022) Robust deep learning–based protein sequence design using ProteinMPNN. Science 378(6615):49–56. 10.1126/science.add2187
8. Bai P, Miljković F, Liu X, De Maria L, Croasdale-Wood R, Rackham O et al (2025) Mask-prior-guided denoising diffusion improves inverse protein folding. Nat Mach Intell 7(6):876–888. 10.1038/s42256-025-01042-6
9. Gao Z, Tan C, Chacón P, Li SZ PiFold: Toward effective and efficient protein inverse folding [Internet]. arXiv; 2023 [cited 2026 Jun 12]. Available from: http://arxiv.org/abs/2209.12643 10.48550/arXiv.2209.12643
10. Yi K, Zhou B, Shen Y, Lió P, Wang Y (2023) Graph Denoising Diffusion for Inverse Protein Folding. Adv Neural Inf Process Syst 36:10238–10257
11. Song Z, Huang T, Li L, Jin W, SurfPro Functional Protein Design Based on Continuous Surface [Internet]. arXiv; 2024 [cited 2026 Jun 12]. Available from: http://arxiv.org/abs/2405.06693 10.48550/arXiv.2405.06693
12. Zhang Z, Ou C, Cho Y, Akiyama Y, Ovchinnikov S (2025) Artificial intelligence methods for protein folding and design. Curr Opin Struct Biol 93:103066. 10.1016/j.sbi.2025.103066
13. Gainza P, Wehrle S, Van Hall-Beauvais A, Marchand A, Scheck A, Harteveld Z et al (2023) De novo design of protein interactions with learned surface fingerprints. Nature 617(7959):176–184. 10.1038/s41586-023-05993-x
14. Li D, Shen L, Song M, Li D, Liu J, Liu X (2026) SurfFold: a unified model for protein inverse folding by integrating surface and structural information. Bioinformatics 42(1):btaf666. 10.1093/bioinformatics/btaf666
15. Wang C, Zhou Y, Wang Z, Zhai Z, Shen J, Zhang K (2026) Alternate Geometric and Semantic Denoising Diffusion for Protein Inverse Folding. In: Ribeiro RP, Pfahringer B, Japkowicz N, Larrañaga

P, Jorge AM, Soares C et al (eds) Machine Learning and Knowledge Discovery in Databases. Research Track. Springer Nature Switzerland, Cham, pp 350–366. doi:10.1007/978-3-032-06066-2_21

16. Rives A, Meier J, Sercu T, Goyal S, Lin Z, Liu J et al (2021) Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proceedings of the National Academy of Sciences. ;118(15):e2016239118. 10.1073/pnas.2016239118
17. Jain S, Wallace BC Attention is not Explanation. In: Burstein J, Doran C, Solorio T, editors. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) [Internet]. Minneapolis, Minnesota: Association for Computational Linguistics; 2019 [cited 2026 Jun 13]. pp. 3543–56. Available from: https://aclanthology.org/N19-1357/ doi:10.18653/v1/N19-1357
18. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D et al (2020) Array programming with NumPy. Nature 585(7825):357–362. 10.1038/s41586-020-2649-2
19. Cock PJA, Antao T, Chang JT, Chapman BA, Cox CJ, Dalke A et al (2009) Biopython: freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics 25(11):1422–1423. 10.1093/bioinformatics/btp163
20. Orengo C, Michie A, Jones S, Jones D, Swindells M, Thornton J (1997) CATH – a hierarchic classification of protein domain structures. Structure 5(8):1093–1109. 10.1016/S0969-2126(97)00260-8
21. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H et al (2000) The Protein Data Bank. Nucleic Acids Res 28(1):235–242. 10.1093/nar/28.1.235
22. Zhou X, Chen G, Ye J, Wang E, Zhang J, Mao C et al (2023) ProRefiner: an entropy-based refining strategy for inverse protein folding with global graph attention. Nat Commun 14(1):7434. 10.1038/s41467-023-43166-6
23. Löffler P, Schmitz S, Hupfeld E, Sterner R, Merkl R (2017) Rosetta:MSF: a modular framework for multi-state computational protein design. PLoS Comput Biol 13(6):e1005600. 10.1371/journal.pcbi.1005600
24. Henikoff S, Henikoff JG (1992) Amino acid substitution matrices from protein blocks. Proceedings of the National Academy of Sciences. ;89(22):10915–9. 10.1073/pnas.89.22.10915

Figures

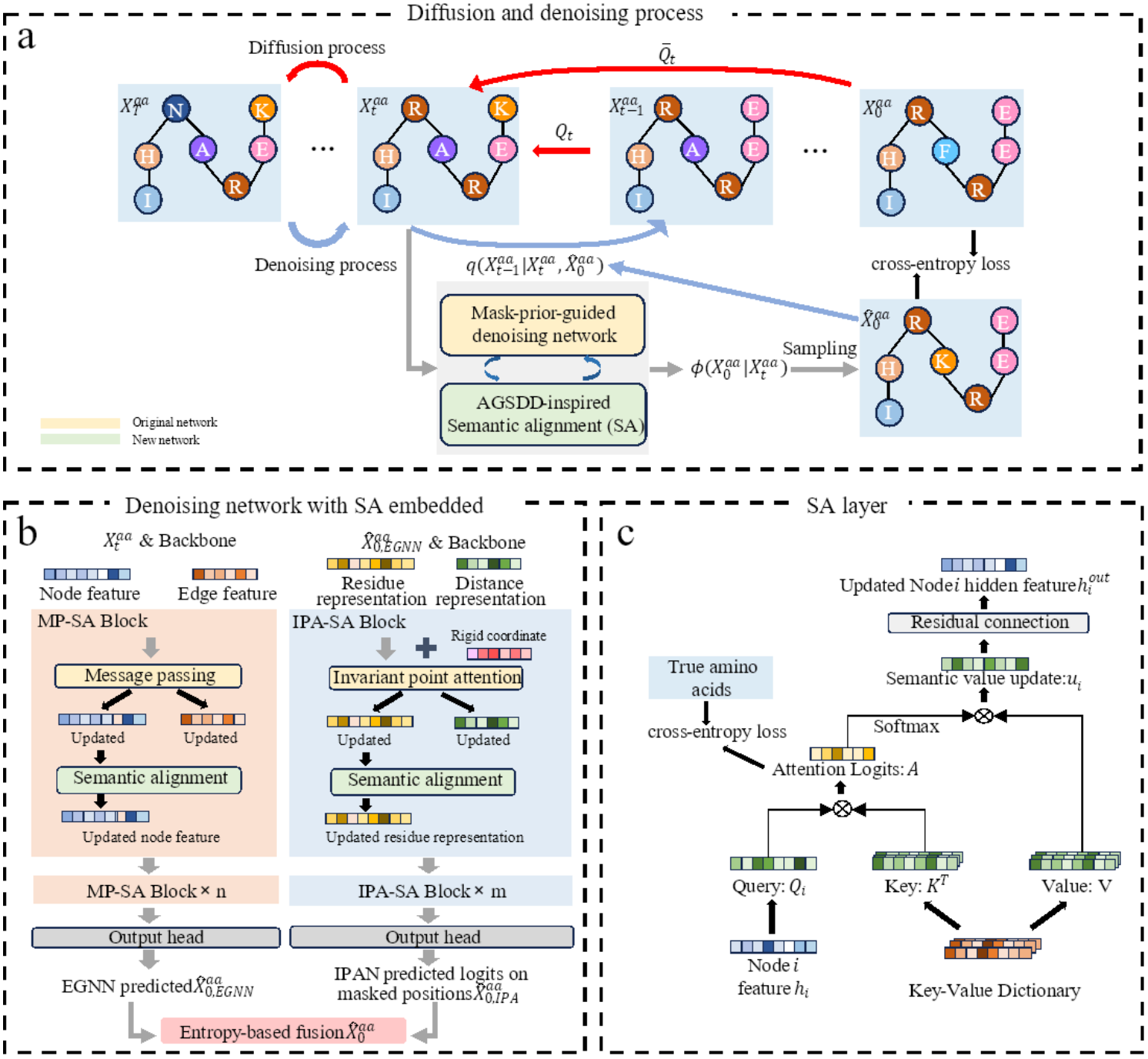

Figure 1

The overall framework of AGSDD-inspired-MapDiff. a, Diffusion and denoising process. The diffusion process progressively adds random discrete noise through the transition matrix, and the denoising process denoises noisy sequences through iterative prediction by the denoising network and cross-entropy minimization. b, Denoising network with semantic alignment (SA) layer embedded. EGNN predicts base logits, and IPAN refines masked positions. c, SA layer. Based on a cross-attention mechanism, SA aligns hidden node/residue representations with learnable amino-acid type embeddings.

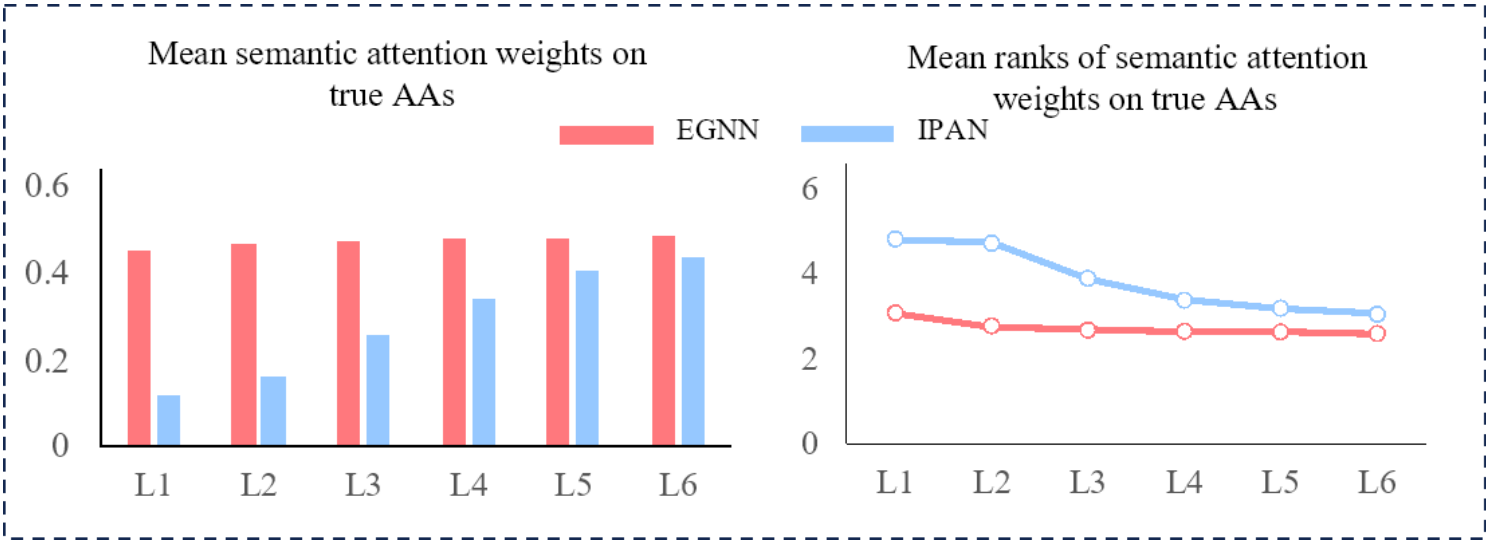

Figure 2

Semantic attention weights across layers in EGNN and IPAN. The left panel shows the mean semantic attention weights on true AAs, and the right panel illustrates the mean ranks of semantic attention weights on true AAs.