Alternative Explanation of Point Features in IPA

Study Notes

December 9, 2025

1 Motivation

This note provides an alternative, slightly more streamlined explanation of *point features* in invariant point attention (IPA). The focus is on:

- where the description comes from in the AF2 supplementary material,
- what point features actually are,
- how they fit into the usual Q/K/V picture of self-attention,
- why frames and local/global coordinates appear when distances enter the attention score.

2 Do Point Features Really Appear in AF2?

Yes. All components are present in the AF2 supplementary.

1. In Supplementary Methods 1.8.2:

   > Invariant Point Attention (IPA) (Suppl. Fig. 8 and Algorithm 22) is a form of attention that acts on a set of frames (parametrized as Euclidean transforms (T_i)) and is invariant under global Euclidean transformations (T_{global}) on said frames.

2. Supplementary Figure 8 shows the IPA module; its caption mentions

   > query pts. $(r, h, p, 3)$, key pts. $(r, h, p, 3)$, value pts. $(r, h, p', 3)$, coordinates in local frames, coordinates in the global frame.

3. Algorithm 22 defines IPA as

   $\texttt{InvariantPointAttention}(\{s_i\}, \{z_{ij}\}, \{T_i\}, N_{\text{head}} = 12, c = 16, N_{\text{query points}} = 4, N_{\text{point values}} = 8),$

   and lines 2–3 produce

   - query/key points $\tilde{q}_i^{hp}, \tilde{k}_i^{hp} \in \mathbb{R}^3$, $p = 1, \ldots, N_{\text{query points}}$,
   - value points $\tilde{v}_i^{hp} \in \mathbb{R}^3$, $p = 1, \ldots, N_{\text{point values}}$.

   Thus the paraphrase

Scalar features s_i are projected to queries, keys, values of shape $N_{\text{res}} \times d_{\text{head}}$, and to point features of shape $N_{\text{res}} \times N_{\text{pts}} \times 3$ with $N_{\text{query pts}} = 4$ and $N_{\text{point values}} = 8$,

is an accurate summary of Algorithm 22 and Supplementary Figure 8.

3 Standard Self-Attention: Grounding the Mental Model

Consider standard self-attention. We have a sequence of positions i, each with a feature vector

$$s_i \in \mathbb{R}^{d_{\text{model}}}.$$

For a single head:

$$\begin{aligned} q_i &= W_Q s_i \in \mathbb{R}^{d_{\text{head}}}, \\ k_i &= W_K s_i \in \mathbb{R}^{d_{\text{head}}}, \\ v_i &= W_V s_i \in \mathbb{R}^{d_{\text{head}}}. \end{aligned}$$

For each pair (i, j):

$$\begin{aligned} \ell_{ij} &= \frac{1}{\sqrt{d_{\text{head}}}} q_i^\top k_j, \\ a_{ij} &= \frac{\exp(\ell_{ij})}{\sum_{j'} \exp(\ell_{ij'})}, \\ o_i &= \sum_j a_{ij} v_j. \end{aligned}$$

The pattern is:

1. Define a similarity score between i and j (usually a dot product),
2. Apply softmax to get weights,
3. Use the weights to form a weighted sum of value vectors.

Crucially:

The notion of "attention" is encoded entirely in the score function ℓ_{ij}. We can add terms to ℓ_{ij} (biases, distance penalties, etc.) and still have self-attention.

4 Extra Structure in the AF2 Structure Module

In the structure module, each residue i has:

- the usual scalar feature vector $s_i \in \mathbb{R}^{c_s}$,
- a rigid frame $T_i = (R_i, t_i)$ (rotation R_i and translation t_i).

The frame defines a local coordinate system:

- local coordinates: $\tilde{x} \in \mathbb{R}^3$ describing a point relative to residue i's Cα,
- global coordinates: $x \in \mathbb{R}^3$ in the shared protein frame.

The mapping is:

- local $\to$ global:
$$x = T_i \circ \tilde{x} = R_i \tilde{x} + t_i,$$
- global $\to$ local:
$$\tilde{x} = T_i^{-1} \circ x = R_i^\top (x - t_i).$$

Backbone frames are built from N, Cα, and C via Algorithm 21 and Gram–Schmidt, with the origin at Cα.

So each residue i has:

- a feature vector s_i,
- a local 3D frame T_i.

IPA is the attention mechanism that uses *both*.

5 Two Kinds of Features in IPA: Scalar and Point

Inside IPA, each residue i yields two families of features from s_i:

1. the usual scalar queries, keys, and values,
2. a small set of 3D point features.

5.1 Scalar Q, K, V (Familiar Part)

Algorithm 22, line 1:

$$q_i^h, k_i^h, v_i^h = \text{LinearNoBias}(s_i), \quad q_i^h, k_i^h, v_i^h \in \mathbb{R}^c.$$

For each head h and residue i:

- q_i^h is the query vector,
- k_i^h is the key vector,
- v_i^h is the value vector.

This matches the standard Transformer pattern.

5.2 Point Q, K, V (New Part)

Algorithm 22, lines 2–3:

$$\tilde{q}_i^{hp}, \tilde{k}_i^{hp} = \text{LinearNoBias}(s_i), \qquad \tilde{q}_i^{hp}, \tilde{k}_i^{hp} \in \mathbb{R}^3, p = 1, \ldots, N_{\text{query points}},$$
$$\tilde{v}_i^{hp} = \text{LinearNoBias}(s_i), \qquad \tilde{v}_i^{hp} \in \mathbb{R}^3, p = 1, \ldots, N_{\text{point values}}.$$

Algorithm arguments specify:

$$N_{\text{query points}} = 4, \qquad N_{\text{point values}} = 8.$$

In terms of shapes:

- query points form
$$\tilde{q} \in \mathbb{R}^{N_{\text{res}} \times N_{\text{head}} \times N_{\text{query points}} \times 3},$$
- similarly for key points $\tilde{k}$,
- value points form
$$\tilde{v} \in \mathbb{R}^{N_{\text{res}} \times N_{\text{head}} \times N_{\text{point values}} \times 3}.$$

These 3-dimensional vectors are what the supplementary and Figure 8 call *point features*. They are:

> Learned functions of s_i, like Q/K/V, but living in $\mathbb{R}^3$ and transforming like points under rigid motions.

Per head and residue, we now have:

- scalar query vector q_i^h,
- point query cloud $\{\tilde{q}_i^{hp}\}_p$,
- scalar key and value vectors,
- point key and value clouds.

6 How Point Features Enter the Attention Score

We connect this to the "attention is dot products" view.

Algorithm 22, line 7 gives, for head h:

$$a_{ij}^h = \text{softmax}_j \left(w_L \left[\sqrt{\frac{1}{c}}\, q_i^{h\top} k_j^h + b_{ij}^h - \gamma_h w_C^2 \sum_p \left\| T_i \circ \tilde{q}_i^{hp} - T_j \circ \tilde{k}_j^{hp} \right\|^2 \right] \right).$$

Three components inside the brackets:

1. **Dot product term**
$$\sqrt{\tfrac{1}{c}}\, q_i^{h\top} k_j^h$$
is the standard scaled dot-product similarity.
2. **Pair bias term** b_{ij}^h comes from z_{ij} via a linear map and depends only on the residue pair.
3. **Geometric distance term**
$$-\gamma_h w_C^2 \sum_p \left\| T_i \circ \tilde{q}_i^{hp} - T_j \circ \tilde{k}_j^{hp} \right\|^2.$$

6.1 Geometry Inside the Distance

For each point index p:

- $\tilde{q}_i^{hp}$ is in the local frame of residue i,
- $\tilde{k}_j^{hp}$ is in the local frame of residue j,
- local-to-global mapping yields

$$T_i \circ \tilde{q}_i^{hp} = R_i \tilde{q}_i^{hp} + t_i, \qquad T_j \circ \tilde{k}_j^{hp} = R_j \tilde{k}_j^{hp} + t_j,$$

- subtracting these gives a global 3D displacement and squared distance.

Summing over p,

$$D_{ij}^h = \sum_p \left\| T_i \circ \tilde{q}_i^{hp} - T_j \circ \tilde{k}_j^{hp} \right\|^2,$$

so the geometric contribution is

$$-\gamma_h w_C^2 D_{ij}^h.$$

Because of the minus sign:

- if the query and key point clouds for i and j are close in 3D, D_{ij}^h is small and the penalty is small,
- if they are far apart, D_{ij}^h is large and pushes the logit down.

For each head h:

> Residue i prefers to attend to residues j whose learned key-point pattern is close in 3D to its own learned query-point pattern, given the current frames.

The full score is still "dot product plus bias plus a distance-based bias", so the mental model becomes:

> Attention is a dot product in feature space plus additional similarity/bias terms; IPA adds one that depends on 3D distances between learned point features.

7 Why Frames and Local/Global Coordinates Are Needed

You might ask: why do local/global and frames appear at all?

Reasons:

- Query and key points are stored in local coordinates, one coordinate system per residue,
- Local coordinates are not directly comparable across residues: each residue has its own origin and axes,
- To compute a meaningful distance between a point on residue i and a point on residue j, we must express them in a common coordinate system.

Frames provide this bridge:

- local to global: $T_i \circ \tilde{q}_i^{hp}$ places $\tilde{q}_i^{hp}$ in global 3D,
- global distances: we subtract global points and compute Euclidean norms, which have physical meaning.

If we tried instead to subtract $\tilde{q}_i^{hp} - \tilde{k}_j^{hp}$ directly, we would be mixing two unrelated coordinate systems; the resulting vector would not correspond to any physical displacement.

So:

- point features live in local frames,
- frames T_i convert them to global coordinates for comparison,
- distances are computed in global space, the only place where they make sense across residues.

8 Invariance of the Score Under Global Rigid Motions

There is a second reason for the local/global machinery: IPA should be invariant to where the whole protein sits in space.

The supplement proves that applying a global rigid transform T_{global} to all frames leaves the logits unchanged.

Sketch:

- Replace each frame with $T_i' = T_{\text{global}} \circ T_i$,
- The distance term transforms to

$$\left\| T_i' \circ \tilde{q}_i^{hp} - T_j' \circ \tilde{k}_j^{hp} \right\|^2 = \left\| T_{\text{global}} \circ (T_i \circ \tilde{q}_i^{hp}) - T_{\text{global}} \circ (T_j \circ \tilde{k}_j^{hp}) \right\|^2,$$

- Since rigid transforms preserve distances, this equals

$$\left\| T_i \circ \tilde{q}_i^{hp} - T_j \circ \tilde{k}_j^{hp} \right\|^2,$$

- Dot product and pair bias do not depend on frames, so the entire logit and therefore a_{ij}^h are unchanged.

Thus local/global not only enables distances, but also ensures those distances are invariant under global rotation and translation.

9 Point Features in the Outputs

Point features appear not only in the scores but also in the outputs of IPA. Given weights a_{ij}^h, Algorithm 22 defines three outputs:

1. **Pair-weighted pair representation**

$$\tilde{o}_i^h = \sum_j a_{ij}^h z_{ij},$$

2. **Standard scalar value aggregation**

$$o_i^h = \sum_j a_{ij}^h v_j^h,$$

3. **Point aggregation with frames**

$$\tilde{o}_i^{hp} = T_i^{-1} \circ \left(\sum_j a_{ij}^h (T_j \circ \tilde{v}_j^{hp}) \right).$$

For the point aggregation:

- For each neighbor j, map $\tilde{v}_j^{hp}$ from local to global:

$$x_{j,\text{global}}^{hp} = T_j \circ \tilde{v}_j^{hp},$$

- Take an attention-weighted average in global space:

$$y_{i,\text{global}}^{hp} = \sum_j a_{ij}^h x_{j,\text{global}}^{hp},$$

- Map back into residue i's local frame:

$$\tilde{o}_i^{hp} = T_i^{-1} \circ y_{i,\text{global}}^{hp}.$$

Thus $\tilde{o}_i^{hp}$ is a new point feature in residue i's local frame, summarizing the 3D neighborhood indicated by attention.

The same invariance argument shows that if we apply the same global transform to all frames, the local outputs $\tilde{o}_i^{hp}$ remain unchanged.

So point features appear on both sides of IPA:

- as point clouds in the score,
- as aggregated point clouds in the outputs.

10 Unified Transformer Mental Model

If we strip away geometry, IPA reduces to:

- standard multi-head attention on s_i with
  - scalar $Q/K/V$,
  - dot-product scores,
  - pair bias from z_{ij},

which is entirely ordinary in modern Transformers.

IPA adds:

1. a set of 3D point features (query, key, value points) per residue and head, derived from s_i,
2. an extra term in the score penalizing squared distances between query and key point clouds (after mapping through frames),
3. a point-valued output pathway aggregating value point clouds in global 3D and then mapping back to residue-local coordinates.

Everything still fits the same template:

- compute scalar scores from Q, K, pair bias, and geometric distances,
- apply softmax to get attention weights,
- form weighted sums over scalar values and point values.

The logic becomes:

> Start from standard attention; attach a small 3D sensor array (point features) to each residue via its frame; use these sensors to make attention geometry-aware while preserving SE(3) invariance.

That is all point features are: a method to inject 3D geometry into the familiar Q/K/V framework without sacrificing symmetry.