DiGSA: Differentiable Graph-Structured Sparse Attention for Large-Scale Face Clustering

Anonymous authors
Paper under double-blind review

Abstract

Large-scale face clustering relies on K-nearest-neighbor (KNN) graphs that are corrupted by false-positive (FP) edges linking visually similar yet distinct identities, as well as false-negative (FN) disconnections that fragment single identities. The strongest existing denoising method, Diff-Cluster, subtracts attention distributions and applies hard Top-K masks, but this design introduces two structural bottlenecks: non-differentiable discrete support selection that blocks gradient flow and necessitates multi-branch mixture-of-experts (MoE) compensation, and sequence-centric positional encodings that fail to capture local graph topology. To address both bottlenecks, we present DiGSA, a Differentiable Graph-Structured Sparse Attention framework comprising two coupled modules. For the first bottleneck, Differentiable Adaptive Sparse Attention (DASA) replaces hard Top-K heuristics with continuous entmax sparse distributions controlled by learnable per-head temperature, making support selection score-adaptive and piecewise differentiable while consolidating multi-branch MoE routing into a single branch. For the second, Graph-Structured Differential Attention (GSDA) injects degree and multi-hop random-walk encodings into queries and keys while applying opposite adjacency-guided biases to the primary and secondary attention maps, encouraging the subtraction to separate same-community evidence from cross-community noise. Across MS1M (584K–5.21M faces), MSMT17, and DeepFashion, DiGSA matches or consistently exceeds the strongest sparse differential baseline, while consolidating the multi-branch mixture-of-experts into a single branch that reduces parameters by 14% and FLOPs by 33%.

1 Introduction

Organizing web-scale facial imagery into identity clusters requires balancing scalability with structural reliability. State-of-the-art pipelines extract deep representations from recognition models such as ArcFace and construct candidate K-nearest-neighbor (KNN) graphs in feature space (Guo et al., 2016; Deng et al., 2019; Zhu et al., 2021). These candidate graphs are inherently corrupted by high-dimensional noise: false-positive (FP) edges connect visually similar yet distinct identities, while false-negative (FN) disconnections arise across extreme variations in pose, illumination, age, expression, and sensor quality. Face clustering therefore requires robust graph relation learning over noisy candidate topologies rather than direct global geometry.

Classical clustering assumptions fail on unconstrained facial embeddings: K-Means assumes convex isotropic clusters, hierarchical agglomerative clustering (HAC) suffers irreversible linkage errors, and DBSCAN collapses under global density fluctuations (Lloyd, 1982; Sibson, 1973; Ester et al., 1996). In deep face spaces, a single identity frequently splits into disjoint visual manifolds, while distinct identities with shared demographic or lighting attributes form dense local clumps. These complex distributions require learning local pairwise connectivity on the graph structure rather than fitting global geometric prototypes.

Graph neural networks and attention-based architectures improve local connectivity estimation, yet dense message passing risks amplifying the very graph corruption it seeks to correct. GCNs such as L-GCN, LTC, GCN(V+E), and STAR-FC propagate node features across local subgraphs to pre-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

dict linkage confidence (Wang et al., 2019; Yang et al., 2019; 2020; Shen et al., 2021), but when false-positive edges participate in feature aggregation, cross-identity noise contaminates neighboring embeddings. Transformer models such as Clusformer, FaceT, and LCEPCE expand the relational receptive field (Nguyen et al., 2021; Ye et al., 2021; Shin et al., 2023), but dense self-attention assigns positive probability mass to hard-negative distractors, impairing cluster discrimination without strict sparsity controls.

The current state-of-the-art, Diff-Cluster, employs a Mixture-of-Experts Sparse Differential Transformer (MoE-SDT) that denoises candidate relations by subtracting attention distributions and applying hard Top-K masks, but suffers from two structural flaws (Zhang et al., 2026). First, hard Top-K masking is non-differentiable: the model cannot backpropagate through the thresholding decision to adaptively expand or contract active support during training. This truncation addresses only the FP problem by discarding suspicious edges, but cannot recover FN edges. If a true same-identity neighbor falls outside the Top-K boundary, the zero-gradient mask permanently severs that connection. Because hard boundaries trigger abrupt step-function discontinuities, MoE-SDT further requires a complex multi-branch mixture-of-experts (MoE) architecture to hedge against discrete boundary misclassifications, inflating both parameter count and computational cost. Second, applying Rotary Position Embedding (RoPE) imposes an arbitrary one-dimensional sequence ordering onto sampled KNN neighbor lists, misrepresenting the non-Euclidean graph topology (Su et al., 2024).

We present DiGSA, an end-to-end differentiable graph-structured sparse attention framework that unifies score-dependent sparsity with adjacency-guided differential scoring. The key insight is that sparse support should be modeled as a continuous entmax distribution rather than hard Top-K masking, yielding more informative subgradients for edges near the active/inactive boundary while still assigning exact zeros to irrelevant distractors. This formulation addresses both FP and FN noise simultaneously: entmax produces exact-zero probabilities for false-positive distractors (FP suppression), while the continuous support boundary allows gradient flow to recover missed same-identity connections by increasing their attention scores above the entmax threshold (FN recovery). By replacing sequence encodings with degree and multi-hop random-walk statistics, and injecting opposite adjacency-guided biases into the primary and secondary maps, DiGSA assigns distinct physical roles to the differential components: the primary map reinforces graph-supported community relations, while the secondary map models residual cross-community noise.

Our primary contributions are:

1. **Differentiable Adaptive Sparse Attention (DASA):** We replace discrete Top-K masking with continuous entmax-(1.5) sparse attention modulated by learnable per-head temperature. DASA renders active support selection end-to-end differentiable, consolidating multi-branch MoE routing into a single differentiable sparse branch. Unlike hard Top-K, which only suppresses false positives, the continuous entmax support boundary enables gradient-based recovery of false-negative connections, simultaneously addressing both FP and FN noise.
2. **Graph-Structured Differential Attention (GSDA):** We replace sequence orderings with Graph Positional Encoding (GraphPE), comprising degree statistics and multi-hop random-walk return probabilities, and introduce adjacency-guided differential biases, physically separating same-community retention from cross-community noise subtraction.
3. **Comprehensive Empirical Validation:** Across all five large-scale MS1M benchmarks (584K to 5.21M faces), DiGSA consistently outperforms Diff-Cluster under identical evaluation protocols. Component-level ablations and efficiency measurements further validate the proposed mechanisms. Notably, DiGSA reduces parameters by 14%, FLOPs by 33%, and achieves a $1.42\times$ throughput improvement over Diff-Cluster.

2 Related Work

Graph-Based Relation Learning for Face Clustering. Face clustering has shifted from geometric partitioning to local connectivity learning on candidate graphs. Classical methods relying on global prototypes or uniform density thresholds degrade under unconstrained facial variations (Lloyd, 1982; Sibson, 1973; Ester et al., 1996). Otto et al. (2017) established the pairwise F-score

protocol at scale. Subsequent work improved candidate graph construction (Zhan et al., 2018; Guo et al., 2020), refined relation prediction via GCN-based linkage confidence (Wang et al., 2019; Yang et al., 2019; 2020; Shen et al., 2021; Liu et al., 2021; Zhang et al., 2023; Jia et al., 2025), and enhanced candidate topologies through structure-aware and vision-language guidance (Wang et al., 2022b;a; Shin et al., 2023; Shen et al., 2023). Standard GCNs, however, propagate indiscriminately along existing edges; when KNN graphs contain false-positive links, aggregation blurs identity boundaries, motivating attention-based edge reweighting.

Differential and Sparse Transformer Attention. Transformer self-attention expands relational receptive fields beyond local graph convolutions, but dense score computation is vulnerable to relational noise (Vaswani et al., 2017). Clusformer and FaceT capture long-range structural dependencies (Nguyen et al., 2021; Ye et al., 2021), yet dense softmax attention assigns positive weights to false neighbors. Differential Transformer introduces subtractive denoising by computing the difference between two normalized attention distributions (Ye et al., 2024). Diff-Cluster adapts this with hard Top-K sparsity masks and multi-branch MoE routing (Zhang et al., 2026), but discrete thresholding blocks backpropagation through the support boundary. DiGSA replaces hard masks with differentiable entmax distributions, enabling end-to-end gradient flow through support selection.

Differentiable Sparsity and Graph Structural Encodings. Fixed sparse patterns such as Sparse Transformer and BigBird reduce memory but cannot dynamically filter structural noise (Child et al., 2019; Zaheer et al., 2020). Sparsemax (Martins & Astudillo, 2016) and Entmax (Peters et al., 2019) produce exact zeros with continuous subgradients, but standard entmax uses a fixed global temperature giving every head the same sparsity regime. DiGSA replaces this with a learnable per-head temperature, so each head can independently sharpen or relax its support. Dot-product attention is permutation-invariant and topology-agnostic (Dwivedi & Bresson, 2021), so Graph Transformers inject positional signals like degree distributions and random-walk return probabilities encoding community retention and node centrality (Dwivedi et al., 2022). Prior face clustering methods apply RoPE to KNN neighbor lists (Su et al., 2024), but neighbor order in a face subgraph is arbitrary, so DiGSA injects degree and random-walk encodings into queries and keys instead of sequence positions. Standard Graph Transformers add a structural bias to a single attention map, which cannot separate supported neighbors from cross-community distractors. DiGSA instead applies opposite adjacency-guided biases to the two score maps, biasing the primary map toward supported neighbors and the secondary map toward non-adjacent distractors, so their subtraction cancels common-mode noise. These three components jointly make sparse support selection and topological reasoning differentiable within a single attention core, rather than independent add-ons to existing entmax or Graph Transformer designs.

3 METHOD

3.1 PROBLEM SETUP AND OVERVIEW

Given N face images, a frozen face-recognition model extracts one embedding $\mathbf{x}_i \in \mathbb{R}^d$ per image. We construct a candidate KNN graph $\mathcal{G}_c = (\mathcal{V}, \mathcal{E}_c)$ by connecting each node to its top-K nearest neighbors under cosine similarity. Let $\mathbf{X} \in \mathbb{R}^{N\times d}$ denote the node features, $\mathcal{S} \in \mathbb{R}^{N\times K}$ the similarity matrix recording cosine similarities between each node and its top-K nearest neighbors, and $\mathcal{N} \in \mathbb{R}^{N\times K}$ the corresponding adjacency matrix. The model predicts an edge probability p_{ij} for each candidate pair $(i, j) \in \mathcal{E}_c$; after which the same graph-clustering backend as the baseline, the Map Equation method, recovers the identity clusters (Rosvall et al., 2009).

The closest baseline, Diff-Cluster, combines differential attention with hard Top-K sparse masks (Zhang et al., 2026). Differential attention forms two score maps and subtracts their normalized attention distributions; MoE-SDT further applies a hard Top-K mask M to enforce sparsity:

$$\mathrm{DiffAtt}(\mathbf{X}) = \big(M \odot S(\tfrac{Q_1 K_1^T}{\sqrt{d_h}}) - \lambda\, M \odot S(\tfrac{Q_2 K_2^T}{\sqrt{d_h}})\big)V, \tag{1}$$

where $S(\cdot)$ is softmax, M is the binary Top-K mask applied after softmax to zero out entries outside the top-K ranks, $\odot$ is element-wise multiplication, and λ is a learnable scalar (Ye et al., 2024). MoE-SDT extends this single-mask design into a multi-branch mixture-of-experts structure applying several hard masks around an externally predicted neighbor boundary with learnable mixing weights,

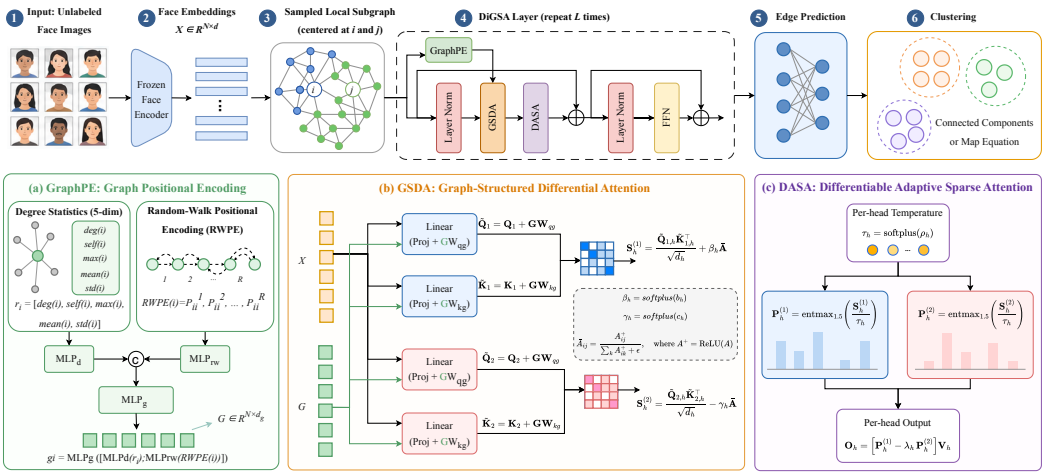

Figure 1: Overview of DiGSA. A sampled KNN subgraph provides node features and graph structure. GraphPE encodes degree and random-walk signals, GSDA builds graph-biased differential score maps, DASA produces differentiable sparse attention before edge prediction and clustering.

compensating for boundary errors after a discrete Top-K decision. Its fundamental limitation is that support selection remains discontinuous: edges just outside the boundary receive no attention, and small score changes not crossing the Top-K threshold cannot expand or shrink the support.

DiGSA keeps the candidate graph, the edge-prediction target, and the final clustering stage unchanged, replacing only the attention core with differentiable graph-structured sparse attention. As shown in Figure 1, Graph Positional Encoding (GraphPE) first summarizes local and multi-hop topology. GSDA then injects this topology into the two differential score maps with opposite adjacency-guided biases. DASA finally turns the scores into sparse entmax distributions controlled by learnable per-head temperature, making sparse support a trainable part of relation learning. The final model thus uses a single differentiable sparse branch rather than a mixture of hard-mask branches.

3.2 Graph Positional Encoding

GraphPE provides structural information that sequence positional encodings cannot reliably represent on sampled KNN graphs. The neighbor order in a face subgraph is arbitrary, so RoPE-style sequence positions inject an artificial ordering not reflecting the underlying graph topology (Su et al., 2024). We instead encode each node using degree statistics and random-walk return probabilities, directly describing local density and community retention. All structural features use per sampled subgraph of n nodes, where n is the subgraph size used during training and inference.

Adjacency Preprocessing. We first preprocess the adjacency matrix for a consistent structural signal. From the subgraph adjacency matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$, computed as $\mathbf{A} = \mathbf{X}_s\mathbf{X}_s^\top$ from the sampled subgraph features $\mathbf{X}_s \in \mathbb{R}^{n\times d}$, we remove negative entries via $\mathbf{A}^+ = \mathrm{ReLU}(\mathbf{A})$ to retain only non-negative scores. For GraphPE feature extraction, we further symmetrize the matrix to ensure an undirected graph structure:

$$\hat{\mathbf{A}} = \frac{\mathbf{A}^+ + (\mathbf{A}^+)^\top}{2}, \tag{2}$$

while for the GSDA attention bias, we apply row normalization so each row sums to one, preserving relative connection strengths without scale distortion:

$$\bar{A}_{ij} = \frac{A^+_{ij}}{\sum_k A^+_{ik} + \epsilon}, \tag{3}$$

where ϵ is a small constant for numerical stability. All degree and random-walk features in GraphPE are computed from $\hat{\mathbf{A}}$, while the GSDA adjacency-guided bias uses $\bar{\mathbf{A}}$. Both are $n \times n$ subgraph-local matrices, making per-batch overhead modest and independent of the full graph size.

Degree Features. The degree feature of node i is a five-dimensional vector summarizing its weighted neighborhood:

$$\mathbf{r}_i = [\deg(i),\ \mathrm{self}(i),\ \max(i),\ \mathrm{mean}(i),\ \mathrm{std}(i)], \tag{4}$$

where $\deg(i) = \sum_j \hat{A}_{ij}$ is the total weighted degree, $\mathrm{self}(i) = \hat{A}_{ii}$ the self-loop weight, $\max(i) = \max_j \hat{A}_{ij}$ the strongest neighbor similarity, and $\mathrm{mean}(i)$, $\mathrm{std}(i)$ the average and standard deviation of neighbor similarities over $\mathcal{N}(i)$. Together, these statistics capture local density, connectivity strength, and neighborhood heterogeneity.

Random-walk Positional Encoding. To capture multi-hop structure, let $\mathbf{D} = \mathrm{diag}(\deg(1), \ldots, \deg(n))$ be the diagonal degree matrix and $\mathbf{P} = \mathbf{D}^{-1}\hat{\mathbf{A}}$ the random-walk transition matrix, where P_{ij} gives the probability of transitioning from node i to node j in one step. For a maximum walk length R, the random-walk positional encoding of node i is

$$\mathrm{RWPE}(i) = \big[(\mathbf{P})_{ii},\ (\mathbf{P}^2)_{ii},\ \ldots,\ (\mathbf{P}^R)_{ii}\big], \tag{5}$$

where $(\mathbf{P}^k)_{ii}$ is the return probability of a k-step random walk from node i back to itself. High return probabilities indicate a densely connected community, while low return probabilities suggest boundary or bridge nodes. Since $\mathbf{P}$ is an $n \times n$ subgraph-local matrix, computing $\mathbf{P}^k$ for $k = 1, \ldots, R$ requires only $O(n^2 R)$ operations per subgraph and is performed on-the-fly during training; we cache the diagonal return probabilities for reuse across layers.

Fusion and Injection. The two structural signals are projected and fused into a graph position vector:

$$\mathbf{g}_i = \mathrm{MLP}_g\big([\mathrm{MLP}_d(\mathbf{r}_i);\ \mathrm{MLP}_{rw}(\mathrm{RWPE}(i))]\big), \tag{6}$$

where MLP_d and MLP_{rw} are independent two-layer perceptrons projecting the degree feature $\mathbf{r}_i \in \mathbb{R}^5$ and the random-walk feature $\mathrm{RWPE}(i) \in \mathbb{R}^R$ into a shared embedding space, $[\cdot;\cdot]$ denotes concatenation, and MLP_g fuses the concatenated features into $\mathbf{g}_i \in \mathbb{R}^{d_g}$. Stacking all n node vectors yields $\mathbf{G} = [\mathbf{g}_1; \mathbf{g}_2; \ldots; \mathbf{g}_n] \in \mathbb{R}^{n \times d_g}$.

For layer ℓ, GraphPE is injected into queries and keys but not values:

$$\tilde{\mathbf{Q}} = \mathbf{Q} + \mathbf{G}\mathbf{W}_{qg}, \quad \tilde{\mathbf{K}} = \mathbf{K} + \mathbf{G}\mathbf{W}_{kg}, \tag{7}$$

where $\mathbf{W}_{qg} \in \mathbb{R}^{d_g \times d_h}$ and $\mathbf{W}_{kg} \in \mathbb{R}^{d_g \times d_h}$ are learnable projection matrices mapping $\mathbf{G}$ into the query and key spaces, respectively. This allows graph structure to influence relation selection while preserving the semantic content carried by face embeddings in the value vectors.

3.3 GSDA: Graph-Structured Differential Attention

Effective relation denoising on face graphs requires assigning distinct structural roles to the primary and secondary attention maps. Standard Differential Transformer subtracts two unconstrained self-attention maps over identical feature representations (Ye et al., 2024), but in unconstrained graph spaces both maps can converge to similar distributions or capture redundant semantic noise, reducing the subtraction to arbitrary scalar reweighting. GSDA resolves this by injecting opposite topological inductive biases into the two score distributions based on the candidate adjacency matrix $\bar{\mathbf{A}}$.

Specifically, GSDA uses the normalized adjacency as a learnable structural guide rather than a hard constraint, computing graph-biased score maps as follows:

$$\mathbf{S}_h^{(1)} = \frac{\tilde{\mathbf{Q}}_{1,h}\tilde{\mathbf{K}}_{1,h}^{\top}}{\sqrt{d_h}} + \beta_h \bar{\mathbf{A}}, \tag{8}$$

$$\mathbf{S}_h^{(2)} = \frac{\tilde{\mathbf{Q}}_{2,h}\tilde{\mathbf{K}}_{2,h}^{\top}}{\sqrt{d_h}} - \gamma_h \bar{\mathbf{A}}, \tag{9}$$

where h indexes the attention head, and $\beta_h = \mathrm{softplus}(b_h)$, $\gamma_h = \mathrm{softplus}(c_h)$ are strictly positive head-specific scaling parameters initialized at $b_h = c_h = 0.5$, giving $\beta_h, \gamma_h \approx 0.97$. Adding $+\beta_h \bar{\mathbf{A}}$ forces the primary map $\mathbf{S}_h^{(1)}$ to prioritize candidate neighbors supported by existing local connectivity, whereas subtracting $-\gamma_h \bar{\mathbf{A}}$ drives the secondary map $\mathbf{S}_h^{(2)}$ to attend to non-adjacent, residual, or

cross-community distractors. Subtracting the secondary map from the primary map thus explicitly cancels common-mode cross-community noise while preserving intra-identity structural support. The graph-biased score maps $\mathbf{S}_h^{(1)}$ and $\mathbf{S}_h^{(2)}$ serve as input to DASA, which applies differentiable sparse normalization to produce the final attention output.

We note that $\bar{\mathbf{A}}$ serves as a soft bias, not a direct label: the learned projections $\tilde{\mathbf{Q}}, \tilde{\mathbf{K}}$ still dominate the score computation, and β_h, γ_h remain modest after training. The bias steers attention toward structurally plausible regions without replacing the learned relation features, and the entmax-normalized differential output can override it when semantic evidence contradicts the graph structure.

3.4 DASA: Differentiable Adaptive Sparse Attention

End-to-end trainable support selection requires replacing discrete Top-K truncation with continuous entmax probability distributions. In Diff-Cluster, applying a hard Top-K mask sets weights outside the top K ranks to exact zero, creating a non-differentiable step function that blocks gradient flow to pruned edges (Zhang et al., 2026). DASA overcomes this by transforming the score vectors into continuous entmax distributions.

Entmax Sparse Normalization. For an input score vector $\mathbf{z}$, entmax is defined as:

$$\mathrm{entmax}_\alpha(\mathbf{z})_i = \left[(\alpha-1)\, z_i - \tau(\mathbf{z})\right]_+^{1/(\alpha-1)}, \tag{10}$$

where $\tau(\mathbf{z})$ is the unique threshold normalizing the distribution to sum to unity, computed via bisection over the sorted score vector in $O(n \log n)$ time for a support of size n. Although sorting is non-differentiable, the entmax mapping is piecewise differentiable: within each region of constant active support, the output is a smooth function of the scores with a well-defined subgradient. At support boundary transitions, the subgradient may be set to zero for the transitioning element without affecting convergence (Martins & Astudillo, 2016). We select $\alpha = 1.5$ (entmax-1.5) as it provides exact zero probabilities for distractors with smoother subgradients than sparsemax (α=2) near the support boundary (Martins & Astudillo, 2016; Peters et al., 2019). Unlike hard Top-K masking, entmax yields exact-zero probabilities through a score-dependent, piecewise differentiable transformation, allowing the edge-prediction loss to shape sparse attention scores directly.

Learnable Per-head Temperature. To enable dynamic neighborhood sizing across layers and attention heads, DASA modulates entmax sharpness via a learnable per-head temperature parameter T_h (distinct from the entmax threshold $\tau(\mathbf{z})$). For head h, the temperature and resulting sparse attention distributions are computed as:

$$T_h = \mathrm{softplus}(\rho_h), \tag{11}$$

$$\mathbf{P}_h^{(m)} = \mathrm{entmax}_{1.5}\left(\frac{\mathbf{S}_h^{(m)}}{T_h}\right), \tag{12}$$

where $m \in \{1, 2\}$ indexes the primary and secondary graph-biased score maps. A smaller learned T_h sharpens the entmax threshold to retain only high-confidence adjacent neighbors, whereas a larger T_h expands active support to aggregate broader context. This allows different heads to specialize in different sparsity regimes: some focusing on the most confident local connections, others maintaining wider receptive fields.

Differential Output. The final DiGSA differential output for attention head h combines the continuous sparse distributions as:

$$\mathbf{O}_h = \left[\mathbf{P}_h^{(1)} - \lambda_h\, \mathbf{P}_h^{(2)}\right]\mathbf{V}_h, \tag{13}$$

where λ_h is the differential coefficient following the standard exponential parameterization of Differential Transformer (Ye et al., 2024), kept unchanged to isolate our architectural contributions to graph-structured scoring and differentiable sparse support. Compared to the original Differential Transformer, which uses unconstrained softmax over plain dot-product scores, DiGSA replaces softmax with entmax for differentiable sparsity and injects opposite adjacency-guided biases into the two score maps, making the subtraction structurally meaningful rather than arbitrary.

Table 1: Main comparison on MS1M under the Diff-Cluster protocol. DiGSA uses the same training split, KNN construction, and downstream clustering pipeline as Diff-Cluster. Higher is better for both metrics.

Method	584K		1.74M		2.89M		4.05M		5.21M	
	F_P	F_B	F_P	F_B	F_P	F_B	F_P	F_B	F_P	F_B
K-Means (Lloyd, 1982)	79.21	81.23	73.04	75.20	69.83	72.34	67.90	70.57	66.47	69.42
HAC (Sibson, 1973)	70.63	70.46	54.40	69.53	11.08	68.62	1.40	67.69	0.37	66.96
DBSCAN (Ester et al., 1996)	67.93	67.17	63.41	66.53	52.50	66.26	45.24	44.87	44.94	44.74
L-GCN (Wang et al., 2019)	78.68	84.37	75.83	81.61	74.29	80.11	73.70	79.33	72.99	78.60
LTC (Yang et al., 2019)	85.66	85.52	82.41	83.01	80.32	81.10	78.98	79.84	77.87	78.86
GCN(V+E) (Yang et al., 2020)	87.93	86.09	84.04	82.84	82.10	81.24	80.45	80.09	79.30	79.25
Clusformer (Nguyen et al., 2021)	88.20	87.17	84.60	84.05	82.79	82.30	81.03	80.51	79.91	79.95
STAR-FC (Shen et al., 2021)	91.97	89.96	88.28	86.26	86.17	84.13	84.70	82.63	83.46	81.47
FaceT (Ye et al., 2021)	91.12	90.50	89.07	86.84	86.78	85.09	84.10	84.67	83.86	83.86
Ada-NETS (Wang et al., 2022b)	92.79	91.40	89.33	87.98	87.50	86.03	85.40	84.48	83.99	83.28
Chen et al. (Chen et al., 2022)	93.22	92.18	90.51	89.43	89.09	88.00	87.93	86.92	86.94	86.06
FaceMap (Yu et al., 2022)	94.27	92.55	91.31	89.67	89.32	88.20	87.74	87.11	86.37	86.29
Wang et al. (Wang et al., 2022a)	94.94	93.67	91.74	90.81	89.50	89.15	87.04	87.81	85.40	86.76
LCEPCE (Shin et al., 2023)	94.64	93.36	91.90	90.78	90.27	89.28	88.69	88.15	87.35	87.28
CLIP-Cluster (Shen et al., 2023)	94.22	–	91.44	89.44	89.95	87.75	88.93	86.78	87.99	85.85
FC-ESER (Liu, 2024)	95.28	93.85	92.94	91.54	91.61	90.38	90.44	89.50	89.40	88.80
Diff-Cluster (Zhang et al., 2026)	95.46	94.14	93.14	91.87	91.73	90.71	90.89	89.78	90.08	89.14
DiGSA (Ours)	**95.60**	**94.14**	**93.32**	**91.90**	**92.11**	**90.72**	**91.06**	**89.81**	**90.26**	**89.17**

4 EXPERIMENTS

4.1 PROTOCOL

We follow the MS-Celeb-1M protocol of Diff-Cluster (Zhang et al., 2026), training on part 0 and evaluating on parts 1, 3, 5, 7, and 9, which span 584K–5.21M faces. DiGSA and Diff-Cluster use $K{=}80$ KNN graphs, identical face embeddings, identical clustering backend, and identical edge-probability threshold. We report Pairwise F-score (F_P), BCubed F-score (F_B), and normalized mutual information (NMI) (Shi et al., 2018; Bagga & Baldwin, 1998; Strehl & Ghosh, 2002). DiGSA contains 2 transformer layers and 8 attention heads, with $d_h{=}64$, $d_g{=}64$, $R{=}8$ random-walk steps, and subgraphs of $n{=}160$ nodes. We initialize the entmax temperature at $T_h{=}1.0$ and the adjacency-bias parameters at $b_h{=}c_h{=}0.5$. Training uses SGD for 8 epochs with a learning rate of $1{\times}10^{-4}$, weight decay of $1{\times}10^{-4}$, and batch size 512 on 4 NVIDIA A6000 GPUs. Results on MSMT17 (Guo et al., 2016) and DeepFashion (Liu et al., 2016) are reported in the supplementary material.

4.2 MAIN COMPARISON

DiGSA achieves the highest F_P and F_B across all five MS1M scales, with its margin over Diff-Cluster widening as the graph grows. At 584K, DiGSA attains $F_P{=}95.60/F_B{=}94.14$ vs. Diff-Cluster's $95.46/94.14$ ($+0.14$ on F_P). The gain is architectural, not capacity-driven: DiGSA uses 14% fewer parameters (Table 5) yet improves accuracy, because DASA's continuous entmax-1.5 distribution replaces the non-differentiable hard Top-K mask, giving truncated same-identity edges a gradient path for recovery. DiGSA surpasses FC-ESER ($95.28/93.85$) and LCEPCE ($94.64/93.36$).

The advantage scales with data size: the F_P gap widens to $+0.38$ at 2.89M (92.11 vs. 91.73) and remains $+0.18$ at 5.21M (90.26 vs. 90.08). Larger graphs contain more false-negative edges, as extreme pose, illumination, or age variation splits one identity into disconnected components that the fixed $K{=}80$ neighborhood cannot fully cover. Diff-Cluster's hard Top-K mask permanently severs these cross-component links, whereas DASA's continuous entmax boundary lets gradients recover them. The smaller margin at 5.21M likely reflects both methods approaching a performance floor. Classical methods collapse entirely at scale: K-Means drops from 79.21 to 66.47, HAC to $F_P{=}0.37$, and DBSCAN to 44.94, showing that convex or global-density assumptions cannot handle the manifold-split structure of deep face embeddings.

4.3 COMPONENT ABLATIONS

Table 2 isolates the contribution of each component. Removing DASA (replacing entmax with hard Top-K) causes the largest drop: F_P falls by 0.10 and NMI by 0.08. This confirms that differentiable sparse support is the most important architectural choice, as it restores gradient flow to truncated same-identity edges. Removing GSDA (replacing graph-structured differential attention with plain RoPE-based differential attention) costs 0.05 on F_P and 0.05 on NMI, showing that injecting opposite adjacency biases into the two score maps makes the differential subtraction structurally meaningful rather than an arbitrary scalar reweighting. Both components contribute, and their effects compound in the full model.

Table 2: Component ablation on MS1M 584K.

Configuration	F_P	F_B	NMI
DiGSA full	**95.60**	**94.14**	**98.46**
w/o GSDA (RoPE)	95.55	94.13	98.41
w/o DASA (Top-K)	95.50	94.10	98.38

Table 3 dissects the sparse normalization choice. Softmax, which produces no exact zeros, scores lowest, confirming that explicit sparsity is necessary for denoising distractor edges. Entmax-1.5 with a learnable per-head temperature T_h is best, outperforming both a fixed temperature and entmax-2.0. The learnable T_h lets each head specialize in a different sparsity regime, some sharpening the threshold to retain only high-confidence neighbors, others widening the receptive field, which a single global temperature cannot achieve. Entmax-2.0 (sparsemax) underperforms entmax-1.5 because its subgradients are less smooth near the support boundary, making support transitions less stable during training.

Table 3: DASA sparse-support ablation on MS1M 584K.

Sparse mechanism	F_P	F_B	NMI
Softmax	95.44	93.94	98.44
Entmax-1.5, learnable T_h	95.60	94.14	98.46
Entmax-1.5, fixed T	95.48	94.08	98.40
Entmax-2.0	95.56	94.09	98.42

Table 4: GSDA graph-structure ablation on MS1M 584K.

Structural setting	F_P	F_B	NMI
No positional encoding	95.52	94.10	98.39
+ RoPE	95.55	94.13	98.41
+ GraphPE (no adj. bias)	95.57	94.14	98.43
+ GraphPE + adj. bias (full)	95.60	94.14	98.46

Table 4 traces the contribution of graph structure within GSDA. Removing all positional encoding costs 0.08 on F_P and 0.07 on NMI, showing that pure dot-product scores lack the topological context needed to distinguish intra-identity neighbors from cross-community distractors. Adding RoPE recovers only 0.03 on F_P (95.55), because RoPE imposes an artificial sequence order on the arbitrarily ordered KNN neighborhood that does not reflect graph topology. Replacing RoPE with GraphPE (degree statistics + random-walk return probabilities) yields a further 0.02 (95.57), as it encodes local density and community retention directly. The full model, which adds the opposite adjacency biases $\pm\beta_h\bar{\mathbf{A}}$ on top of GraphPE, reaches $95.60/98.46$. The adjacency bias is the final structural ingredient: it steers the primary map toward connectivity-supported neighbors and the secondary map toward residual distractors, making the differential subtraction cancel common-mode cross-community noise while preserving intra-identity support.

4.4 IMPLEMENTATION EFFICIENCY

Table 5 shows that DiGSA is not only more accurate but also substantially more efficient than Diff-Cluster. Parameters drop by 14% (11.12M→9.59M) because DiGSA replaces the multi-branch MoE-SDT structure, which maintains several hard-mask branches with separate mixing weights, with a single differentiable sparse branch. FLOPs fall by 33% (0.7495G→0.5046G) and latency by 29% (3.456ms→2.441ms), yielding a $1.42\times$ throughput improvement (289.4→409.7 samples/s). The FLOPs reduction exceeds the parameter reduction because entmax-1.5 produces exact-zero probabilities for distractors, skipping the corresponding value aggregation at inference, whereas Diff-Cluster's hard Top-K mask still requires computing full softmax distributions before truncation. This efficiency gain is practically significant: at 5.21M faces, the $1.42\times$ throughput advantage

translates into a substantial wall-clock reduction for the relation-prediction stage, which dominates the clustering pipeline.

Table 5: Matched relation-model implementation measurements averaged over 100 runs. Lower is better for parameters, FLOPs, and latency; higher is better for throughput.

Metric	Diff-Cluster	DiGSA	Change
Parameters (M)	11.12	9.59	14% ↓
FLOPs (G)	0.7495	0.5046	33% ↓
Latency (ms)	3.456	2.441	29% ↓
Throughput (samples/s)	289.4	409.7	1.42× ↑

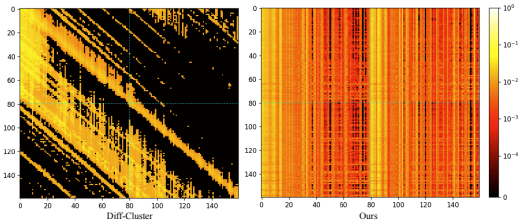

Figure 2: Attention maps under an underestimated k_{off} on MS1M 584K.

4.5 Qualitative Boundary-Error Analysis

The boundary-error diagnostic in Table 6 and Figure 2 explains why DiGSA outperforms Diff-Cluster at the edge level. We examine the underestimated-k_{off} regime, where the assumed neighborhood size is too small and many same-identity edges fall outside the top-K support. In all three scenarios (B_under1–3), Diff-Cluster activates only 17.6, 30.5, and 30.0 within-block attention entries, whereas DiGSA activates 67.3, 67.2, and 62.8, roughly 2–4× more. The "Truncated same-ID recovered" column shows that entmax retains 46.4, 37.5, and 39.0 same-identity entries that Top-K prunes to exact zero. These recovered entries are not noise: DiGSA predicts "same" with probability 1.00 in all three cases, while Diff-Cluster's prediction confidence degrades from 0.999 to 0.941 to 0.716 as the boundary error grows. The mechanism is straightforward: DASA's entmax-1.5 assigns graded, non-zero attention to edges just beyond the Top-K cutoff, so their scores can still be updated by the edge-prediction loss and pushed above the entmax threshold. Diff-Cluster's hard mask zeroes them out, blocking any gradient and freezing the error. Figure 2 visualizes this contrast: beyond the dashed boundary, Diff-Cluster's attention drops to a hard zero, while DiGSA maintains a smoothly decaying sparse tail that preserves same-identity connectivity.

Table 6: Edge-level support diagnostic on MS1M 584K. "Within-block active" counts nonzero attention entries in the same-identity neighborhood, and "Truncated same-ID" counts entries pruned by hard Top-K but retained by entmax. The scenario type gives the ground-truth relation.

Scenario	k_{off}	K_{GT}	Within-block active		Trunc. same-ID recovered	Pred. (prob)	
			Diff-Cl.	DiGSA		Diff-Cl.	DiGSA
k_{off} underestimated (FN-dense)							
B_under1	20	64	17.6	67.3	46.4	same(.999)	same(1.00)
B_under2	40	68	30.5	67.2	37.5	same(.941)	same(1.00)
B_under3	40	69	30.0	62.8	39.0	same(.716)	same(1.00)

5 Conclusion

This paper presented DiGSA, a differentiable graph-structured sparse attention framework for large-scale face clustering on noisy KNN graphs. DiGSA replaces non-differentiable hard Top-K masking with DASA, an entmax-1.5 sparse attention mechanism controlled by learnable per-head temperatures, and replaces sequence-style positional cues with GSDA, which injects degree statistics, multi-hop random-walk encodings, and opposite adjacency-guided biases into both differential attention maps. These designs preserve the standard KNN construction and downstream clustering backend while making sparse support selection and graph-aware denoising trainable within a single attention branch. Under the matched Diff-Cluster protocol, DiGSA achieves the best F_P and F_B across five MS1M scales from 584K to 5.21M faces, with gains reaching 0.38 in F_P and stable results across random seeds. Ablations confirm the contributions of DASA, GSDA, GraphPE, and learnable-temperature entmax-1.5, while boundary-error diagnostics show that DiGSA retains 2–4× more within-identity active entries and recovers same-identity support truncated by hard Top-K. DiGSA reduces parameters by 14%, FLOPs by 33%, and latency by 29%, yielding a 1.42× throughput improvement, with additional MSMT17, DeepFashion, and LCEPCE plug-in results supporting its effectiveness as an accurate and efficient attention core for graph-based relation learning.

REFERENCES

Amit Bagga and Breck Baldwin. Algorithms for scoring coreference chains. In *The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*, pp. 563–566, 1998.

Yingjie Chen, Huasong Zhong, Chong Chen, Chen Shen, Jianqiang Huang, Tao Wang, Yun Liang, and Qianru Sun. On mitigating hard clusters for face clustering. In *European Conference on Computer Vision*, pp. 529–544, 2022.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4690–4699, 2019.

Vijay Prakash Dwivedi and Xavier Bresson. A generalization of transformer networks to graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 6878–6886, 2021.

Vijay Prakash Dwivedi, Ladislav Rampasek, Michael Galkin, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Long range graph benchmark. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pp. 226–231, 1996.

Senhui Guo, Jing Xu, Dapeng Chen, Chao Zhang, Xiaogang Wang, and Rui Zhao. Density-aware feature embedding for face clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. MS-Celeb-1M: A dataset and benchmark for large-scale face recognition. In *European Conference on Computer Vision*, pp. 87–102. Springer, 2016.

Hongjie Jia, Ying Zhi, Qirong Mao, and Heping Song. Enhanced face clustering with neighbor structure refinement. *IEEE Transactions on Computational Social Systems*, 2025.

Junfu Liu, Di Qiu, Pengfei Yan, and Xiaolin Wei. Learn to cluster faces via pairwise classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3845–3853, 2021.

Junjie Liu. Face clustering via early stopping and edge recall. *arXiv preprint arXiv:2408.13431*, 2024.

Ziwei Liu, Ping Luo, Shiyou Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1096–1104, 2016.

Stuart Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2): 129–137, 1982.

André F. T. Martins and Ramón Fernandez Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proceedings of the International Conference on Machine Learning*, pp. 1614–1623, 2016.

Xuan-Bac Nguyen, Duc Toan Bui, Chi Nhan Duong, Tien D. Bui, and Khoa Luu. Clusformer: A transformer based clustering approach to unsupervised large-scale face and visual landmark recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10847–10856, 2021.

Charles Otto, Dayong Wang, and Anil K. Jain. Clustering millions of faces by identity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

Ben Peters, Vlad Niculae, and André F. T. Martins. Sparse sequence-to-sequence models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1504–1519, 2019.

Martin Rosvall, Daniel Axelsson, and Carl T. Bergstrom. The map equation. *The European Physical Journal Special Topics*, 178(1):13–23, 2009.

Shuai Shen, Wanhua Li, Zheng Zhu, Guan Huang, Dalong Du, Jiwen Lu, and Jie Zhou. Structure-aware face clustering on a large-scale graph with 10^7 nodes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9085–9094, 2021.

Shuai Shen, Wanhua Li, Xiaobing Wang, Dafeng Zhang, Zhezhu Jin, Jie Zhou, and Jiwen Lu. CLIP-cluster: CLIP-guided attribute hallucination for face clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20786–20795, 2023.

Yichun Shi, Charles Otto, and Anil K. Jain. Face clustering: Representation and pairwise constraints. *IEEE Transactions on Information Forensics and Security*, 13(7):1626–1640, 2018.

Junho Shin, Hyo-Jun Lee, Hyunseop Kim, Jong-Hyeon Baek, Daehyun Kim, and Yeong Jun Koh. Local connectivity-based density estimation for face clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13621–13629, 2023.

Robin Sibson. SLINK: An optimally efficient algorithm for the single-link cluster method. *The Computer Journal*, 16(1):30–34, 1973.

Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3:583–617, 2002.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Yaohua Wang, Fangyi Zhang, Ming Lin, Senzhang Wang, Xiuyu Sun, and Rong Jin. Robust graph structure learning via multiple statistical tests. *Advances in Neural Information Processing Systems*, 35:32083–32096, 2022a.

Yaohua Wang, Yaobin Zhang, Fangyi Zhang, Ming Lin, Yuqi Zhang, Senzhang Wang, and Xiuyu Sun. Ada-NETS: Face clustering via adaptive neighbour discovery in the structure space. *arXiv preprint arXiv:2202.03800*, 2022b.

Zhongdao Wang, Liang Zheng, Yali Li, and Shengjin Wang. Linkage based face clustering via graph convolution network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1117–1125, 2019.

Lei Yang, Xiaohang Zhan, Dapeng Chen, Junjie Yan, Chen Change Loy, and Dahua Lin. Learning to cluster faces on an affinity graph. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2298–2306, 2019.

Lei Yang, Dapeng Chen, Xiaohang Zhan, Rui Zhao, Chen Change Loy, and Dahua Lin. Learning to cluster faces via confidence and connectivity estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13369–13378, 2020.

Jinxing Ye, Xiaojiang Peng, Baigui Sun, Kai Wang, Xiuyu Sun, Hao Li, and Hanqing Wu. Learning to cluster faces via transformer. *arXiv preprint arXiv:2104.11502*, 2021.

Tianzhu Ye, Li Dong, Yuqing Xia, Yutao Sun, Yi Zhu, Gao Huang, and Furu Wei. Differential transformer, 2024.

Xiaotian Yu, Yifan Yang, Aibo Wang, Ling Xing, Hanling Yi, Guangming Lu, and Xiaoyu Wang. Facemap: Towards unsupervised face clustering via map equation. *arXiv preprint arXiv:2203.10090*, 2022.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Ali Ahmed. Big Bird: Transformers for longer sequences. In *Advances in Neural Information Processing Systems*, volume 33, pp. 17283–17297, 2020.

Xiaohang Zhan, Ziwei Liu, Junjie Yan, Dahua Lin, and Chen Change Loy. Consensus-driven propagation in massive unlabeled data for face recognition. In *European Conference on Computer Vision*, 2018.

Dafeng Zhang, Jiangbo Guo, and Zhezhu Jin. Context-aware face clustering with graph convolutional networks. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1–5, 2023.

Dafeng Zhang, Yongqi Song, and Shizhuo Liu. Enhancing noise resilience in face clustering via sparse differential transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 12403–12411, 2026.

Zheng Zhu, Guan Huang, Jiankang Deng, Yun Ye, Junjie Huang, Xinze Chen, Jiagang Zhu, Tian Yang, Jiwen Lu, Dalong Du, et al. WebFace260M: A benchmark unveiling the power of million-scale deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10492–10502, 2021.

A AI USE STATEMENT

The authors used AI tools to assist with code writing and manuscript polishing. All AI-assisted code and text were reviewed and verified by the authors, who take full responsibility for the final implementation, experimental results, claims, and conclusions.

B APPENDIX

The additional experiments test four narrower questions: whether the two GSDA maps exhibit distinct attention roles, whether the model is stable across random seeds, whether performance remains stable under a controlled similarity intervention, and whether the module transfers across datasets and a second clustering architecture. We interpret each result within the scope of its experimental design.

B.1 RUN-TO-RUN STABILITY

Table 7: Run-to-run stability of DiGSA on MS1M 584K across three random seeds. Std denotes the sample standard deviation across seeds.

Seed	F_P	F_B	NMI
42	95.60	94.14	98.46
2347	95.59	94.11	98.44
6328	95.61	94.19	98.46
Mean $\pm$ Std	95.60 ± 0.01	94.15 ± 0.04	98.45 ± 0.01

Table 7 confirms DiGSA is highly stable across seeds (std 0.01 on F_P, 0.04 on F_B, 0.01 on NMI). This stems from replacing discrete support selection with a differentiable sparse mapping. In Diff-Cluster's MoE-SDT, the hard Top-K mask induces discrete support-boundary switches that gradients cannot correct, so different seeds converge to distinct support configurations. DASA's piecewise-smooth entmax-1.5 mapping instead makes support selection part of the optimized objective: within each constant-support region the output is smooth, and at boundary transitions the subgradient can be set to zero without harming convergence. The opposite adjacency biases $\pm\beta_h\bar{\mathbf{A}}$ in GSDA further constrain the attention search space, reducing sensitivity to initialization.

B.2 GSDA ROLE ANALYSIS

Table 8: GSDA Primary/Secondary attention-mass ratio (P/S) on MS1M 584K. P/S > 1 means Primary dominates; P/S < 1 means Secondary dominates. Diff-Cluster's P/S is identically 1.0 because both maps share the same Top-K mask.

Scenario	Pair type	k_{off}	P/S within-block	P/S cross-block
A_acc1	same-ID	60	1.155	0.868
A_acc3	same-ID	20	1.545	0.601
B_under1	same-ID	20	1.034	0.966
B_under2	same-ID	40	1.046	0.951
A_cross1	cross-ID	40	0.911	1.221
A_cross2	cross-ID	160	0.755	1.625
C_over1	cross-ID	160	0.801	1.499
Diff-Cluster (any scenario)			1.000 / 1.000	

The selected cases show the map asymmetry targeted by GSDA (Table 8). For all four same-identity pairs, Primary attention dominates within the identity block (P/S > 1) and Secondary attention dominates across it (P/S < 1). All three cross-identity pairs exhibit the reverse pattern. Diff-Cluster

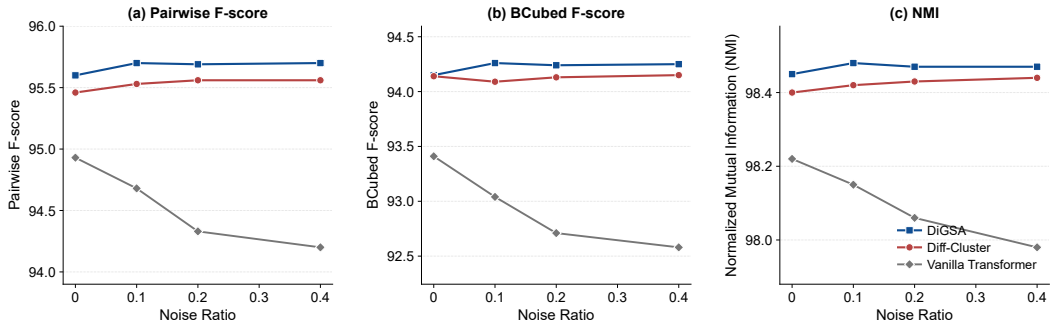

Figure 3: MS1M 584K results under controlled similarity perturbation. The noise ratio is the fraction of randomly selected relation-matrix entries whose similarities are increased before renormalization.

has P/S ratios of 1.000 in both regions because its two maps share the same Top-K mask. The observations are consistent with distinct structural roles for the two maps, but remain case-level diagnostics rather than a population-level or causal estimate.

B.3 Controlled Similarity Perturbation

DiGSA remains above Diff-Cluster at every tested similarity-perturbation ratio (Figure 3). At a 40% perturbation ratio, the Vanilla Transformer decreases from 94.93 to 94.20 in F_P, with corresponding downward trends in F_B and NMI. DiGSA and Diff-Cluster remain stable across all three metrics, and DiGSA reaches 95.70 in F_P at 40%.

This experiment supports stability under the specified intervention, in which randomly selected relation-matrix similarities are increased and then renormalized. It does not establish robustness to arbitrary real-world corruption, because the intervention may also strengthen some same-identity relations.

B.4 Cross-Dataset and Cross-Architecture Transfer

Table 9: Clustering results on MSMT17.

Method	F_P	F_B
GCN(V+E) (Yang et al., 2020)	50.27	64.56
STAR-FC (Shen et al., 2021)	58.80	66.92
Ada-NETS (Wang et al., 2022b)	64.39	73.39
Chen et al. (Chen et al., 2022)	58.59	68.24
FaceMap (Yu et al., 2022)	68.64	75.42
FC-ESER (Liu, 2024)	70.71	75.66
Diff-Cluster (Zhang et al., 2026)	70.83	75.73
DiGSA (Ours)	**70.89**	**75.83**

On MSMT17, DiGSA exceeds Diff-Cluster by 0.06 in F_P and 0.10 in F_B and records the highest values in Table 9. This agreement in direction with MS1M provides evidence of transfer to one person re-identification dataset. The small margins and single transfer dataset do not support a claim of broad cross-domain generalization.

Table 10: Replacing the attention module in LCEPCE. † denotes a replication from open-source code.

Dataset	Method	F_P	F_B
MS1M 584K	LCEPCE + Vanilla†	93.43	92.44
	LCEPCE + Diff	94.68	93.31
	LCEPCE + DiGSA	**94.75**	**93.42**
DeepFashion	LCEPCE + Vanilla†	38.43	60.55
	LCEPCE + Diff	41.76	63.93
	LCEPCE + DiGSA	**42.96**	**64.08**

The plug-in experiment tests DiGSA inside a second clustering framework (Table 10). Relative to LCEPCE + Diff, LCEPCE + DiGSA gains 0.07/0.11 in F_P/F_B on MS1M 584K and 1.20/0.15 on DeepFashion. The consistent direction across both datasets supports plug-in compatibility with LCEPCE. Because the study covers one alternative host architecture and two datasets, it does not establish architecture-independent generality.