Method —ASCender

Scaled Dot-product Attention

$$\text{Attention}(Q, K, V) \;=\; \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$

- $Q \in \mathbb{R}^{n \times d_k}$: **Query** 행렬
- $K \in \mathbb{R}^{m \times d_k}$: **Key** 행렬
- $V \in \mathbb{R}^{m \times d_v}$: **Value** 행렬
- d_k : Key (또는 Query) 벡터의 차원
- $QK^\top \in \mathbb{R}^{n \times m}$: Query 와 Key 의 내적 점수
- $\frac{1}{\sqrt{d_k}}$: 점수 크기 (scale) 를 조정하는 정규화 상수
- softmax$(\cdot)$: 각 Query 에 대해 Key 방향으로 확률 분포를 만듦 (attention weight)
- 최종적으로 V 를 weighted sum 하여 attention 결과를 얻음

표기법 및 설정 (Notation & Setup)

- 입력 시퀀스 $X = [x_1, \dots, x_n]^\top \in \mathbb{R}^{n \times d_{\text{model}}}$, 임베딩된 토큰 $x_i \in \mathbb{R}^{d_{\text{model}}}$.
- 선형 변환: $Q = XW_Q,\ K = XW_K,\ V = XW_V$ with $W_Q, W_K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_V \in \mathbb{R}^{d_{\text{model}} \times d_v}$.
- $$(QK^\top)_{ij} = q_i^\top k_j.$$
- 기본 어텐션 스코어 (Base attention scores): $S_{ij}^{\text{base}} = \frac{q_i^\top k_j}{\sqrt{d_k}}$.
- softmax$_j$ 는 인덱스 j 에 대한 행 방향 소프트맥스.
- 선택적 위치 임베딩 π_i (절대/상대/회전).
- Multi-head 인덱스 h 는 명확한 경우 생략하며, 모든 항목은 헤드별로 정의됨.

$$S_{ij} = S_{ij}^{\text{base}} + \beta_{ij}^{\text{align}} + \beta_{ij}^{\text{sep}} + \beta_{ij}^{\text{coh}} + \beta_{ij}^{(\text{other})}, \quad A_{ij} = \text{softmax}_j(S_{ij}), \quad \text{Attn}(X) = AV.$$

여기서 $\beta^{(\text{other})}$ 는 ALiBi, 상대 위치 바이어스 등 기존 항과의 병용을 허용.

학습된 잠재 기하 (Learned Latent Geometry)

토큰들을 군집화하기 위해 학습된 거리 공간 (learned metric space) 을 도입.

- 잠재 좌표 (latent coordinates):

$$z_i = x_i U, \quad U \in \mathbb{R}^{d_{\text{model}} \times d_z}, \quad z_i \in \mathbb{R}^{d_z}$$

여기서 $x_i \in \mathbb{R}^{1\times d_{\text{model}}}$ 는 입력 토큰 임베딩의 row-vector 표현.

- 거리 (distance):

$$d_{ij} = \|z_i - z_j\|_2$$

- 의미 유사도 (semantic affinity):

$$a_{ij} = \frac{h_i^\top h_j}{\|h_i\|\,\|h_j\|}, \quad h_i = x_i P, \quad P \in \mathbb{R}^{d_{\text{model}}\times d_a}$$

잠재 좌표 z 는 토큰 간의 기하적 가까움 (geometry) 을, 의미 유사도 a_{ij} 는 토큰 간의 의미적 비슷함 (semantics) 을 각각 반영.

근접 이웃 정의 (Neighborhoods)

$$\mathcal{N}_k(i) = \text{TopK}_{j\neq i}(a_{ij}) \quad \text{(의미 기반 Top-}k\text{)}, \qquad w_{ij}^{\tau} = \exp\!\left(-\frac{\|z_i - z_j\|^2}{\tau^2}\right).$$

$\mathcal{N}_k(i)$ 는 토큰 i 와 가장 의미적으로 유사한 상위 k 개 이웃의 집합을 의미하며, w_{ij}^{τ} 는 잠재 좌표 공간에서의 거리 기반 가중치 (temperature τ 포함).

정렬 바이어스 β^{align}

토큰 i 가 의미론적 이웃의 **평균 방향**에 **정렬 (alignment)** 되도록 유도한다.

토큰 i 의 **지역 방향 (local heading)** 은 다음과 같이 정의한다:

$$u_i = \frac{\sum_{l\in\mathcal{N}_k(i)} \tilde{k}_l}{\left\|\sum_{l\in\mathcal{N}_k(i)} \tilde{k}_l\right\|_2}, \qquad \tilde{k}_l = \frac{k_l}{\|k_l\|_2}.$$

토큰 쌍 (i, j) 의 정렬 점수는 다음과 같다:

$$r_{ij}^{\text{align}} = \tilde{k}_j^\top u_i \in [-1, 1], \qquad \beta_{ij}^{\text{align}} = \lambda_{\text{align}} \cdot \gamma_{\text{align}}(i) \cdot r_{ij}^{\text{align}}.$$

여기서 게이팅 함수는

$$\gamma_{\text{align}}(i) = \sigma\big(\alpha_{\text{align}} \cdot \text{Var}_{l\in\mathcal{N}_k(i)}[\tilde{k}_l]\big),$$

으로 정의되며, 이웃 방향의 **분산**이 낮을수록 (= 일관될수록) 정렬 항이 강해진다.
(이웃들의 방향성이 **한쪽으로 일관**될수록 정렬 바이어스가 강해지도록 설계)

분리 바이어스 β^{sep}

토큰 i 주변의 **과밀 (crowding)** 및 **중복 (redundancy)** 을 억제한다.

지역 밀집도 (local density) 는

$$\rho_i = \sum_{l \neq i} w_{il}^{\tau_{\text{sep}}}, \qquad \eta_i = \min\Big(1, \frac{\rho_i}{\kappa}\Big).$$

중복 커널은 다음과 같다:

$$\phi_{ij}^{\text{red}} = w_{ij}^{\tau_{\text{sep}}} \cdot \max\big(0,\, a_{ij} - \delta\big).$$

따라서 분리 바이어스는

$$\beta_{ij}^{\text{sep}} = -\lambda_{\text{sep}} \cdot \eta_i \cdot \phi_{ij}^{\text{red}}.$$

(**밀집도** ρ_i 가 높을수록, 그리고 **의미 유사도** a_{ij} 가 임계값 δ 이상일수록 $i \leftrightarrow j$ 연결을 **억제**한다. 이는 근거리 과밀과 장거리 불필요한 상호작용을 동시에 억제)

응집 바이어스 β^{coh}

토큰 i 를 **잠재 중심 (centroid)** 으로 끌어당겨 의미적 **군집 (cohesion)** 을 형성한다.

지역 중심은

$$c_i = \frac{\sum_l w_{il}^{\tau_{\text{coh}}} z_l}{\sum_l w_{il}^{\tau_{\text{coh}}}}.$$

토큰 쌍 (i, j) 의 응집 점수는

$$r_{ij}^{\text{coh}} = -\|z_j - c_i\|_2^2, \qquad \beta_{ij}^{\text{coh}} = \lambda_{\text{coh}} \cdot \gamma_{\text{coh}}(i) \cdot \frac{r_{ij}^{\text{coh}}}{\tau_{\text{coh}}}.$$

여기서

$$\gamma_{\text{coh}}(i) = \sigma\big(\alpha_{\text{coh}} \cdot \text{Var}_l[z_l]\big),$$

으로, 지역 잠재 공간이 지나치게 흩어져 있으면 응집 효과를 약화시킨다.
(토큰 j 가 i 의 **응집 중심** c_i 에 가까울수록 보너스를 받아 의미적 군집이 강화)

정규화 및 안정화

행 단위 (토큰 i 단위) 정규화로 바이어스의 **드리프트**를 방지한다:

의사코드 (Pseudocode)

Per head h:

```
Inputs: X (n x d_model), WQ, WK, WV, U, P,
params {k, tau_sep, tau_coh, lambda_align, lambda_sep, lambda_coh, omega_*}

Q = X WQ; K = X WK; V = X WV
Z = X U        # latent coords
H = X P        # semantic proj for affinity

# (1) base scores (optionally add standard pos. biases)
S_base = (Q @ K^T) / sqrt(d)

# (2) neighborhoods & kernels
a = cosine(H, H)                    # (n x n) or windowed
N_k = topk_indices(a, k)            # per row
d2 = pairwise_sqdist(Z, Z)          # windowed if long seq
w_sep = exp(-d2 / tau_sep); w_coh = exp(-d2 / tau_coh)

# (3) Alignment
k_hat = normalize_rows(K)
u = normalize_rows(sum_over_neighbors(k_hat, N_k))
r_align = row_dot(k_hat, u)       # r_align[i,j] = k_hat[j]·u[i]
beta_align = lambda_align * gate_align(u, N_k) * r_align

# (4) Separation
rho = row_sum(w_sep)                # local density
eta = clip(rho / kappa, 0, 1)
phi_red = w_sep * relu(a - delta)
beta_sep = - lambda_sep * outer(eta, 1s) * phi_red

# (5) Cohesion
c = rowwise_weighted_centroid(Z, w_coh)
```

```
r_coh = - rowwise_sqdist_to(Z, c)
beta_coh = (lambda_coh / tau_coh) * gate_coh(Z) * r_coh
```