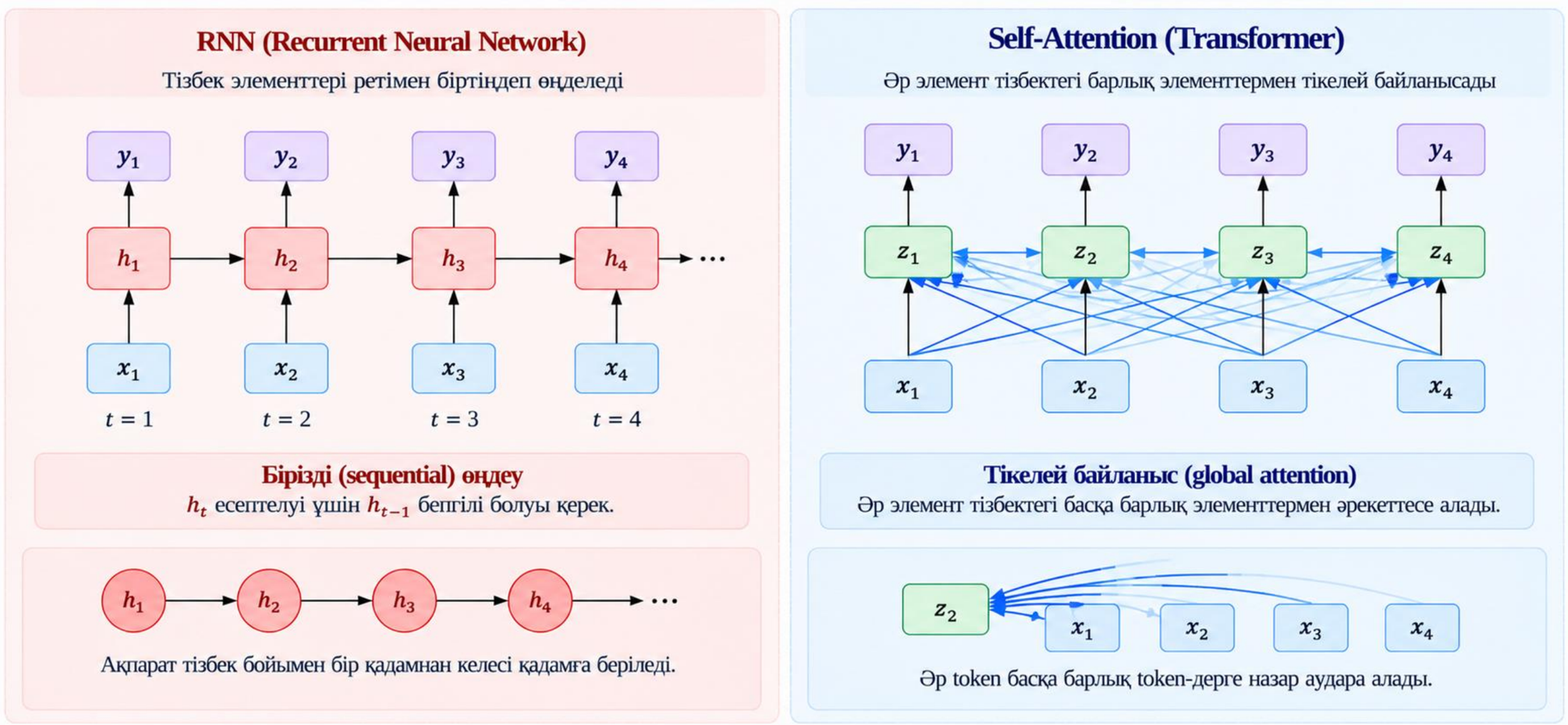
Заманауи терең нейрондық желілер және Transformer архитектурасы

Тізбектегі алыс байланыстарды тікелей модельдеу

Дәрістің негізгі міндеттері

- Q, K, V және Attention есептеуін түсіндіру.
- Encoder, Decoder, masking және cross-attention қызметтерін ажырату.
- Оқыту, генерациялау және есептеу күрделілігін салыстыру.

RNN және Self-Attention: ақпарат ағыны



Тікелей байланыстардың қолжетімділігі attention маскасына тәуелді.

Дереккөз: 10-дәріс конспектісі, 2-бет, 10.1-сурет.

Query, Key және Value

$$Q = XW_Q,\ K = XW_K,\ V = XW_V$$

Query — сұраныс

Ағымдағы позицияға қандай ақпарат қажет екенін сипаттайтын проекция.

Key — кілт

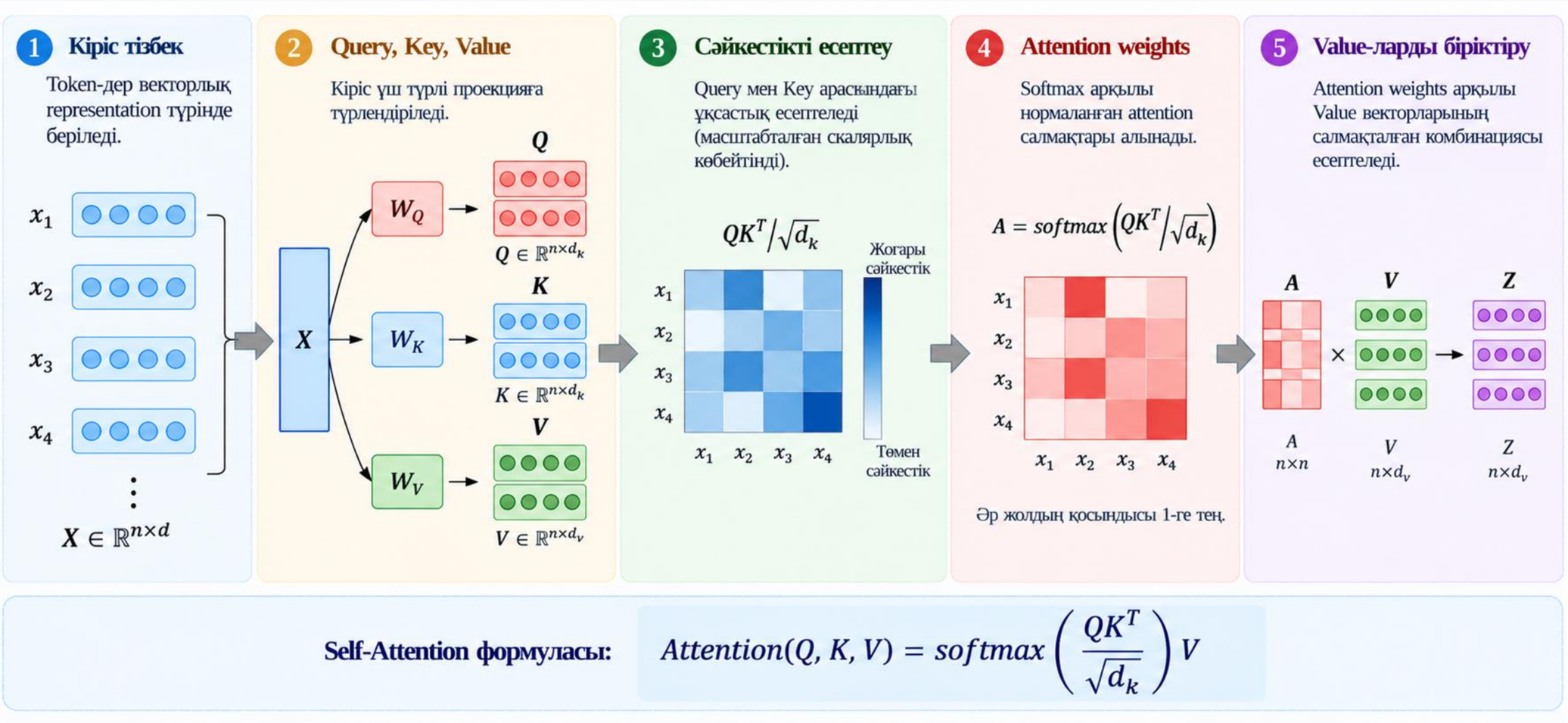
Басқа позициялармен сәйкестікті есептеуге арналған проекция.

Value — мән

Салмақталған қосындыда біріктірілетін ақпараттық вектор.

Self-Attention: Q, K және V бір кіріс тізбегінен қалыптасады.

Self-Attention есептеуінің толық схемасы



Дереккөз: конспект, 3-бет, 10.2-сурет.

Scaled Dot-Product Attention: формула және өлшем

$$\mathbf{Attention(Q,K,V) = softmax(QK^T / \sqrt{d_k})V}$$

Қадам	Өрнек	Өлшем
Сәйкестік	$S = QK^T / \sqrt{d_k}$	$n \times n$
Салмақтар	$A = \text{softmax}(S)$, әр жол бойынша	$n \times n$
Жаңа көрініс	$Z = AV$	$n \times d_v$

$\sqrt{d_k}$ масштабы үлкен dot-product мәндерінің softmax-ты қанықтыруын азайтады.

Attention: конспектідегі шағын мысал

$q = [1; 1]$, $k_1 = [1; 0]$, $k_2 = [0; 1]$, $k_3 = [1; 1]$.

Элемент	$q \cdot k_j$	$(q \cdot k_j)/\sqrt{2}$	Softmax салмағы
1	1	0,707	≈ 0,248
2	1	0,707	≈ 0,248
3	2	1,414	≈ 0,503

$z \approx 0{,}248v_1 + 0{,}248v_2 + 0{,}503v_3$; үшінші Value-дың үлесі үлкенірек.

Multi-Head Attention

$$\text{head}_i = \text{Attention}(QW_iQ, KW_iK, VW_iV)$$

$$\text{MultiHead} = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O$$

Әр head жеке үйренілетін проекцияда байланыстарды есептейді. Нәтижелер біріктіріліп, шығыс кеңістігіне проекцияланады.

Белгілі бір head міндетті түрде бір ғана тілдік немесе семантикалық қасиетті кодтамайды.

Позициялық ақпарат: тізбектің ретін беру

X_input = X_embedding + X_position

Классикалық синусоидалық кодтау:

PE(pos, 2i) = sin(pos / 10000^(2i/d_model))
PE(pos, 2i + 1) = cos(pos / 10000^(2i/d_model))

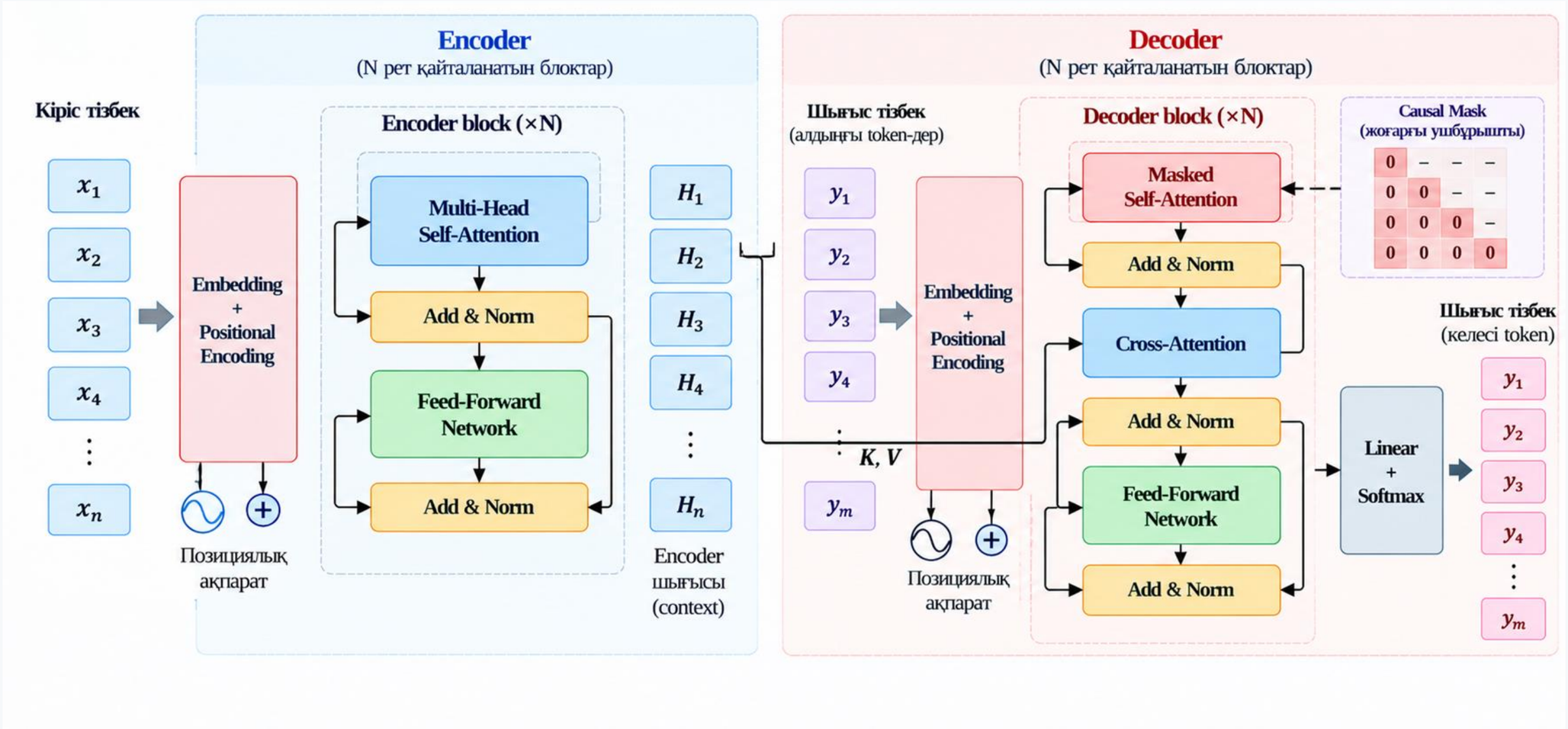
Қосу — классикалық тәсіл; үйренілетін, салыстырмалы және басқа позициялық әдістер де бар.

FFN, residual connection және normalization

Компонент	Өрнек	Қызметі
FFN	$\sigma(xW_1 + b_1)W_2 + b_2$	Әр позицияның белгілерін түрлендіру
Residual	$x + F(x)$	Тікелей ақпарат және градиент жолы
LayerNorm	LayerNorm(x + Sublayer(x))	Белгілерді нормалау

Attention позициялар арасында ақпарат алмасады.
FFN әр позицияда бірдей параметрлермен жұмыс істейді.
σ — белсендіру функциясы; ол міндетті түрде sigmoid емес.

Классикалық Transformer Encoder–Decoder



Дереккөз: конспект, 5-бет, 10.3-сурет.

Causal mask және Cross-Attention

$$S_{masked} = QK^T / \sqrt{d_k} + M$$

Masked Self-Attention

Болашақ позициялар үшін $M = -\infty$.
Softmax-тан кейін олардың салмағы 0.

Decoder кірісі бір позицияға ығыстырылып, алдыңғы дұрыс токендерді береді.

Cross-Attention

Q — Decoder көріністерінен.
K және V — Encoder шығысынан.

Шығыс тізбегі бастапқы тізбектің контекстін пайдаланады.

Decoder-only модельде Encoder-ге бағытталған cross-attention міндетті компонент емес.

Оқыту және авторегрессиялық генерация

$$L = -\Sigma_t \log p(x_t \mid x_{<t})$$

Training

Дұрыс алдыңғы токендер белгілі.
Causal mask болашақты жабады.

Көп позицияның болжамы мен шығыны бір forward pass-та есептеледі.

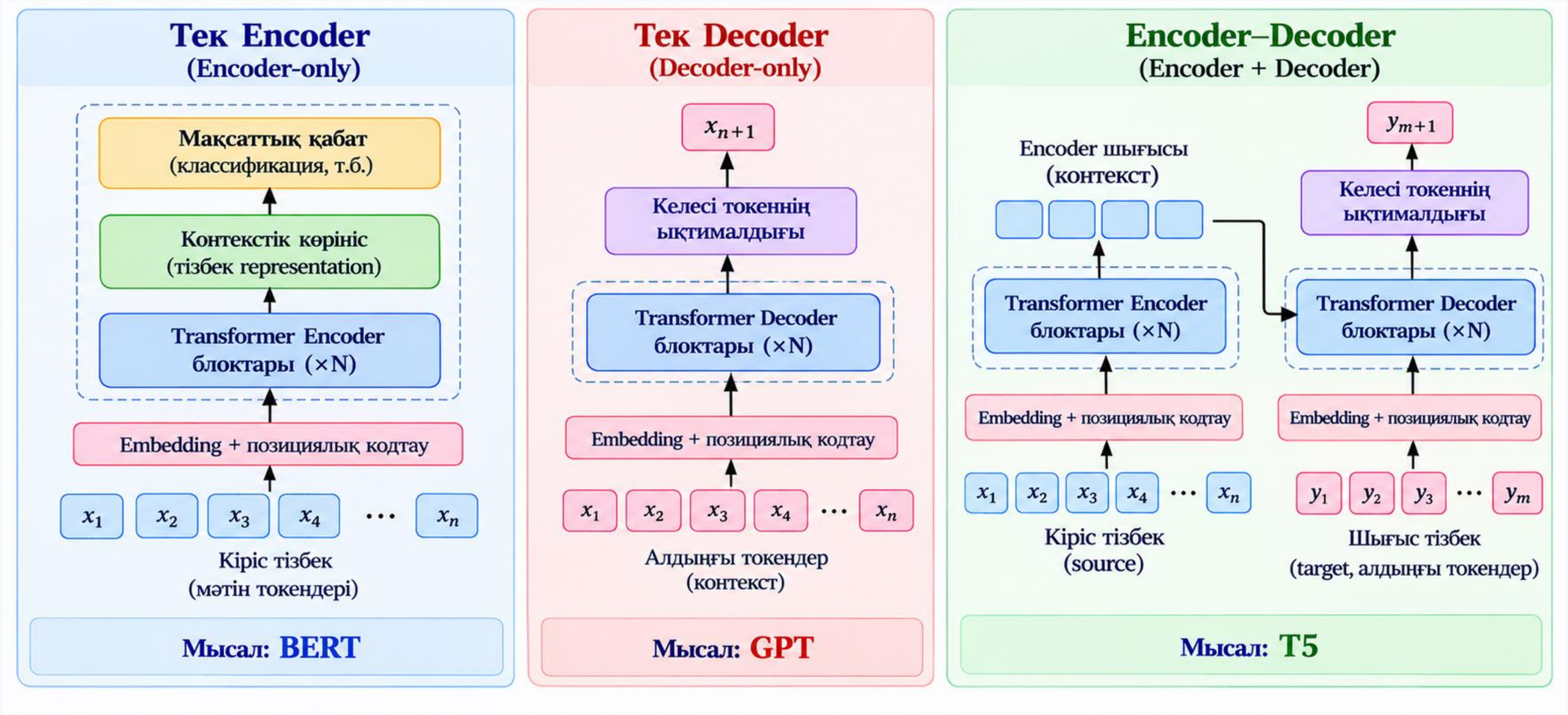
Autoregressive inference

Келесі токен алдын ала белгісіз.

Токен таңдау → контекстке қосу → келесі токенді есептеу.
Процесс қадаммен жалғасады.

Оқытудағы параллельдік авторегрессиялық генерация толық параллель дегенді білдірмейді.

Transformer архитектураларының үш отбасы



Дереккөз: конспект, 7-бет, 10.4-сурет. Мысалдар: BERT, GPT, T5.

Стандартты Self-Attention күрделілігі

$$QK^T \in \mathbb{R}^{n \times n}$$

Тізбек екі есе ұзарса

Attention матрицасы:
$n^2 \rightarrow (2n)^2 = 4n^2$.

Стандартты жұптық есептеу көлемі шамамен төрт есе өседі.

Ұзын контекст мәселесі

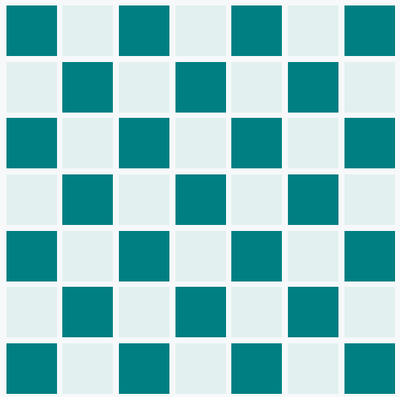
Белгі өлшемдері тұрақты кезде есептеу n бойынша квадраттық өседі.

Толық attention матрицасын сақтау да $O(n^2)$ жад қажет етеді.

Sparse/local attention және тиімді іске асырулар есептеу не жад шығынын азайтуға бағытталады.

Vision Transformer: кескіннен токендерге

224 × 224 кескін → 16 × 16 patch → 196 patch



Patch-тер

→ **Embedding** → **Transformer**

Сызба — шартты. Нақты мысал: 14 × 14 = 196 patch; арнайы токендер есептелмеген.

Transformer мен RNN/LSTM: қашан қайсысы?

Сипаттама	RNN / LSTM	Transformer
Ақпарат ағыны	Рекуррентті күй арқылы	Attention байланыстары арқылы
Реттілік	Уақыттық есептеуде берілген	Позициялық механизм қолданылады
Training параллельдігі	Уақыт бойымен шектеулі	Көп позицияда жоғары

Таңдау: міндет, дерек көлемі, тізбек ұзындығы, кідіріс және ресурстар.
Ірі модельдер деректер сапасына тәуелді және көп ресурс талап етуі мүмкін.

Attention салмағы модель шешімінің толық түсіндірмесі немесе себеп-салдар дәлелі емес.

Курстың ортақ идеясы: пайдалы көріністі үйрену

Архитектура	Көріністі қалыптастыру механизмі
MLP	Қабаттардағы бейсызық түрлендірулер
CNN	Жергілікті кеңістіктік белгілер
RNN / LSTM / GRU	Уақыттық жасырын күй
Автоэнкодер / VAE / GAN	Жасырын құрылым және генерация
Transformer	Контекстік attention байланыстары

Transfer Learning — бұрын үйренілген көріністі жаңа міндетте қайта пайдалану.

Бақылау сұрақтары

01 Q, K және V қалай қалыптасады, олардың қызметі қандай?

02 Scaled Dot-Product Attention қалай есептеледі және softmax қай бағытта қолданылады?

03 Multi-Head Attention мен позициялық ақпарат не үшін қажет?

04 FFN, residual connection және LayerNorm қандай міндет атқарады?

05 Causal mask пен cross-attention қалай ажыратылады?

06 Encoder-only, decoder-only және encoder–decoder қандай есептерге қолданылады?

07 Training параллельдігі, генерация реті және $O(n^2)$ шектеуі қалай түсіндіріледі?

Дереккөз және иллюстрациялар

Негізгі дереккөз

«10-дәріс. Заманауи терең нейрондық желілер және Transformer архитектурасы». Пайдаланушы ұсынған конспект, 10 бет.

Конспектідегі төрт түпнұсқа схема сақталды

10.1-сурет: RNN және Self-Attention, 2-бет → 2-слайд.
10.2-сурет: Self-Attention есептеуі, 3-бет → 4-слайд.
10.3-сурет: Encoder–Decoder, 5-бет → 10-слайд.
10.4-сурет: модель отбасылары, 7-бет → 13-слайд.

Конспектідегі q және k мысалы үшін softmax салмақтары есептелді. Patch сызбасы — осы презентацияға арналған шартты иллюстрация. Маскалау, өлшемдер және параллельдік шарттары нақтыланды.