Banya AI 3

Orthogonal Data-Operation Tokens and Convolutional Scale-Depth Structure: Label-Free Emergence of Operation-as-Weight

바이토큰: 직교 데이터-연산 토큰과 라벨 없이 창발하는 연산자

한혁진 (bokkamsun@gmail.com) · ORCID 0009-0007-4132-1707

반야 연구 시리즈 제3편

초록

언어를 데이터(data, 내용. 무엇과 비슷한가)와 연산(operation, 변환. 어떻게 바꾸나)의 직교(orthogonal, 서로 수직이라 겹치지 않음) 곱으로 본다. 이 관점에서 토큰(token, 낱말 단위) 하나는 데이터면과 연산면 두 임베딩(embedding, 낱말을 숫자 벡터로 바꾼 좌표)의 직교 쌍으로 존재한다. 이 직교 쌍 토큰을 바이토큰(bi-token)이라 부른다. 바이토큰은 하나의 토큰이 데이터면과 연산면이라는 두 직교 성분을 동시에 갖는 단위이며, 어느 면이 데이터이고 어느 면이 연산인지 미리 라벨하지 않고 기하(직교)와 자리 비대칭(더하기 자리 대 곱하기 자리)만 강제한 것이 특징이다. 데이터면은 잔차 스트림(residual stream, 층을 지나며 값이 더해지는 통로)에 더해지는 내용이고, 연산면은 다음 토큰의 채널 게이트(channel gate, 각 성분을 얼마나 통과시킬지의 문)를 여닫는 곱셈 게이트이다. 어느 토큰이 데이터이고 어느 토큰이 연산인지 라벨을 주지 않고 기하(직교)와 자리 비대칭(더하기 자리 대 곱하기 자리)만 강제하면, 연산면이 어떤 데이터에도 일관된 방향으로 작용하는 일관 연산자로 창발한다. 이를 측정해 다음을 보인다. 연산자의 방향 일관성은 무작위의 10.1배이고, 서로 다른 연산의 72.5퍼센트가 직교하며, 연산면을 켜면 다음 낱말 예측의 교차엔트로피(cross-entropy, 예측 난이도)가 2.79만큼 낮아진다(2.99에서 0.20으로, 동봉 자작 초등 대화 말뭉치 200창 기준). 이 기여는 학습 분포 안 관찰로, 문체가 같은 홀드아웃 텍스트에서는 +0.3에서 +0.6으로 유지되고 아직 학습 단계에 없는 성인 자연어 문체에서는 나타나지 않는다(6절). 토큰 연산 비율은 중간 대역이 비어 있고, 학습된 토큰은 연산 우세 대역에 분포한다. 채널 혼합은 밀집 행렬 대신 순환 필터뱅크(circulant filterbank, 한 벡터를 돌려 만든 필터의 묶음)로 두어 얕은 층은 큰 붓, 깊은 층은 가는 펜이 되도록 스케일이 깊이를 따라 변하는 필터 사다리를 이룬다. 이 순차 구성이 언어의 통합 관계(순서)를 담고, 순서는 혼합 헤드가 담당한다. 혼합 헤드 16개는 학습 시작 시 모두 같은 거리를 보다가 학습이 끝나면 과거 39칸에서 66칸까지 서로 다른 거리를 나눠 읽도록 분화된다. 끝으로 학습에서 보지 못한 어간과 어미 결합형의 조립 판별을 최소쌍으로 측정한다. 안 본 조합 198개에서 정확도가 뒤집기 교란 86.9%, 어미 내부 뒤바꿈 교란 87.7%로 우연 50%를 크게 넘고, 어려운 교란에서 음절 바이그램(bigram, 이웃 두 글자 통계) 기준선이 47.7%로 무너지는 동안에도 유지된다. 직교의 인과 기여는 직교를 제거하고 같은 시드와 말뭉치로 나란히 재학습한 같은 크기 대조 모델로 확인했다. 대조 모델은 안 본 조합 판별이 86.9에서 76.3퍼센트로 하락해 바이그램 기준선(83.3퍼센트) 아래로 떨어지는 반면 예측 교차엔트로피는 0.28로 원본 0.21에 근접해, 직교 제거는 예측 일반이 아니라 조합 구조를 특이적으로 해친다(6절).

측정

이 글의 모든 정량 수치는 집필 환경에서 측정한 실측값이다(초등 단계 얼린 모델 기준, 환경은

부록 B). 파라미터 개수처럼 구조에서 곧바로 계산되는 값은 사실로 표기한다. 직교 관점의 이론적 배경은 별도 공리 문헌 [1]에 있으며, 이 글의 주장은 그 공리에서 연역한 것이 아니라 측정에서 나온다.

목차

용어 범례

용어를 아래 표에 정의한다. 같은 개념에는 같은 이름만 사용하며, 일곱 편의 본문과 그림이 모두 이 표를 따른다. 코드 기준은 동봉 코드에서 그 개념을 구현한 식별자이고, 일반 용어는 학계에서 통용되는 가장 가까운 이름이다. 다를 때만 칸을 채웠다.

용어 범례 1. 엔진과 신용

용어	뜻	코드 기준	일반 용어
정확한 전치	각 연산의 수반을 손으로 유도해 닫힌형으로 구현한 역방향 계산. 이 엔진과 신용 사슬의 고유명	`banya_core.py`	수동 역전파
수반	순전파의 짝 연산. 기울기를 정확히 거꾸로 되돌려 흘린다	`bwd` 커널들	adjoint
신용	어떤 지점의 은닉에 대한 손실 기울기. 파라미터가 결과에 진 책임	델타(δ)	그레이디언트
신용 사슬	캐시를 역순으로 되짚으며 수반을 곱해 신용을 입력까지 되돌리는 골격	`backward`	오차역전파
수반 정합	수반 식이 유한차분과 같은 값을 내는지 확인하는 검사	각 편 프로브	gradient check
순환행렬 채널 혼합	채널 혼합을 순환행렬로 두고 rfft 성분곱으로 계산하는 연산		순환 합성곱
상대 거리 혼합	거리에만 의존하는 공유 계수로 자리들을 섞는 혼합	`m_mat_w_lag`	Toeplitz 혼합
시간혼합	다른 자리의 정보가 현재 자리로 섞여 드는 경로의 총칭		토큰 혼합
혼합 헤드	혼합을 병렬로 나누는 머리 수		어텐션 헤드
필터 사다리	넓은 필터 몇 개부터 좁은 필터 여러 개까지 스케일이 깊이를 따라 변하는 콘볼루션 채널 변환	`m_mat_w_filter`	콘볼루션 필터뱅크
창	모델이 한 번에 보는 토큰 자리들. 측정 단위로는 고정 길이 텍스트 조각	`T`	컨텍스트 윈도
프로브	논문 수치를 목표값과 함께 재생하는 검증 스크립트	`probe` 폴더	재현 스크립트
얼린 모델	학습을 멈춰 고정한 발달 단계 모델	`model` 폴더 npz	frozen checkpoint
홀드아웃	학습에 쓰지 않은 평가 텍스트		held-out set

용어 범례 2. 바이토큰과 게이트

용어	뜻	코드 기준	일반 용어
바이토큰	토큰 하나를 데이터면과 연산면의 직교 쌍으로 두는 구조	`USE_BITOKEN`	
데이터면	토큰의 내용을 담는 임베딩 면. 더하기 자리로 들어간다	`m_mat_w_data_axis`	토큰 임베딩
연산면	토큰의 연산 지시를 담는 임베딩 면. 곱하기 자리로만 들어간다	`m_mat_w_operate_axis`	
더하기 자리, 곱하기 자리	데이터면이 실리는 잔차 덧셈 위치와 연산면이 작용하는 게이트 곱셈 위치		잔차 연결, 게이팅
게이트	각 채널의 통과량을 0에서 1로 정하는 밸브. 직전 토큰의 연산면이 주입된다	`m_mat_w_gate`	게이팅
한 칸 이동	직전 토큰의 연산면이 다음 자리의 게이트로 주입되는 한 자리 밀림		
연산이 곧 가중치	연산면 임베딩이 라벨 없이 가중치 역할로 창발하는 현상		
일관 연산자	어떤 데이터에도 일관된 방향으로 작용하는 연산자		
방향 일관성	같은 연산이 여러 데이터를 꺾는 방향들의 평균 코사인. 연산자 창발의 지표	`p_direction_consistency`	
회전 연산자 게이트	연산면을 오일러 회전각으로 읽는 게이트. 본문 축약은 회전 게이트	`paper7_rotation.py`	
쿼터니언 게이트	연산면 4성분을 단위 쿼터니언으로 읽어 채널 4개 묶음을 좌곱으로 돌리는 게이트	`install_quaternion`	
기본 연산	게이트가 한 홉 지시로 곧바로 표현할 수 있는 연산		프리미티브
깊이 배제 판별	깊이를 1블록으로 제한해 게이트만 변수로 남긴 의도된 편파 측정	제7편 프로브	
자기 되먹임	자기 출력을 입력으로 되먹이며 끝 상태를 채점하는 채점 방식		자기회귀 롤아웃

용어 범례 3. 되새김과 발달

용어	뜻	코드 기준	일반 용어
되새김	기울기 없이 임베딩을 이웃끼리 당겨 뭉치게 하는 자기조직화 학습	제2편 프로브	자기조직화
동종군	씨앗에서 코사인 최근접 이웃을 모은 뜻 가까운 개념 집합		클러스터
무게중심 당김, 감쇠	평균 쪽으로 당기고 원본 쪽으로 되돌리는 되새김의 두 힘. 균형에서 평형이 창발한다	`ALPHA`, 감쇠율	
월드먼저	언어 이전에 감각, 공간, 논리 축을 먼저 발달시키는 커리큘럼 순서	`banya_world_data`	커리큘럼 학습
발달 단계	스텝 수로 정의하고 사람 성장 이름을 붙인 체크포인트 단계		
축	임베딩 안에 방향으로 형성되는 성질의 눈금. 감각, 순서, 범주		표현 축

용어 범례 4. 어휘와 메타인지

용어	뜻	코드 기준	일반 용어
음절 원자	어휘 바닥층의 음절 단위 토큰	`AtomTokenizer`	문자 단위 토큰
묶음	음절 원자를 묶어 사전에 올린 개념 토큰. 어휘 단위로만 사용한다		서브워드
캐시 사전 토큰화	원자 아래층에 묶음 사전 위층을 얹은 2층 어휘 체계	제5편 프로브	서브워드 토큰화와 대비
확실, 모호, 모름	자기 출력 분포로 갈리는 세 앎 상태	pmax, 엔트로피	불확실성 추정
라우팅	상태에 따라 답의 처리 경로를 배정하는 것		routing
재해석	모호 히든을 문맥과 함께 재순전파해 확실로 끌어올리는 절차	제6편 프로브	
유사도 정리	데이터면 유사도로 개념이 정돈되는 축		
추론 순회	연산면을 따라 추론을 이어 가는 축		
자기 발화	유사도 정리와 추론 순회를 엮어 스스로 말을 잇는 목표 상태		
반야프레임	시리즈의 배경을 이루는 공리 15개 체계. 본문 주장의 근거는 아님		

기호 범례

이 편의 수식과 본문이 함께 사용하는 기호를 정의한다. 한 식 안에서만 쓰는 국소 기호는 각 식 아래의 기호 줄에 적었다.

기호	한글 명칭	정의 및 의미	자료형
d	데이터면 임베딩	토큰의 내용(무엇과 비슷한가)을 담는 임베딩으로 잔차 스트림에 그대로 더해진다. 코드명 mat_w_data_axis	벡터
o	연산면 임베딩	토큰의 변환(어떻게 바꾸나)을 담는 임베딩으로 채널 게이트를 여닫으며 항등(초기값 0)에서 시작한다. 코드명 mat_w_operate_axis	벡터
g	채널 게이트	각 성분을 얼마나 통과시킬지 정하는 0에서 1 사이의 밸브. 식 4-1에서는 자리 첨자를 붙여 g_t로 표기한다	벡터
x_t	자리 t 은닉	문맥 자리 t의 은닉 벡터. 차원은 채널 폭 H	벡터
t	문맥 자리 번호	은닉, 게이트, 연산면이 놓이는 문맥 자리를 가리키는 첨자. $t-1$은 한 칸 앞(직전) 토큰의 자리	정수
H	채널 폭	은닉 벡터의 차원. 기본 설정 1024(표 A-1). 5.2절에서 순환 커널이 블록과 겹당 H개만 필요함을 보인다	정수
b_g	게이트 바이어스	게이트의 편향. 초기값 2.0에서 시작해 게이트가 거의 열린 상태로 출발한다(표 A-1)	벡터
o_{t-1}	직전 토큰의 연산면	직전 토큰의 연산면 임베딩이 한 칸 이동해 현재 자리 게이트 안에 더해져 남의 데이터 통과량을 조절한다. 0이면 게이트는 항등	벡터
m_t	시간혼합 입력	시간혼합으로 자리 t에 들어온 남의(다른 토큰의) 데이터. 게이트 g_t와 성분별 곱으로 변조된다	벡터
n	표본 수	조립 판별 최소쌍 과제에서 각 묶음의 유효쌍 개수(본 조합 33, 안 본 조합 198 등)	정수
p	이항검정 p값	무작위 50% 대비 단측 이항검정 값. 동전던지기로 그만큼 맞힐 확률	스칼라

1. 서론

언어학에는 두 종류의 관계가 있다. 계열 관계(paradigmatic, 서로 바꿔 쓸 수 있는 비슷한 것. 곰과 토끼는 둘 다 동물)와 통합 관계(syntagmatic, 순서로 이어지는 것. "곰은" 다음에 "동물이야")이다. 표준 임베딩은 이 둘을 한 벡터에 뭉갠다. 이 글은 토큰을 두 직교 면으로 나눈다. 데이터면은 무엇과 비슷한가(계열, 내용)를 담고, 연산면은 어떻게 변환되는가(변환, 곱셈)를 담는다. 그리고 그 둘이 순차로 이어지는 것이 통합 관계를 만든다. 이는 언어를 데이터와 연산의 직교 곱으로 보는 관점이며, 이 관점의 형식적 배경은 별도 공리 문헌 [1]에 있다. 이 글은 그 관점이 실제 학습에서 무엇을 만드는지를 측정으로 검증한다.

핵심 물음은 하나이다. 어느 토큰이 연산자인지 라벨을 주지 않고 기하만 강제해도, 연산면이 일관 연산자로 갈라져 나오는가. 측정 결과는 그러하다. 연산면은 항등(곱해도 그대로)에서 출발해 라벨 없이 갈라지며, 어떤 데이터에도 일관 방향으로 작용하는 일관 연산자(무작위의 10.1배)로 창발하고, 켜면 예측을 2.79만큼 돕는다.

1.1 기여

- 직교 데이터-연산 토큰 : 토큰을 데이터면(더하기)과 연산면(곱셈 게이트) 두 직교 임베딩으로 나누는 구조를 제시한다(3절).
- 라벨 없는 연산자 창발 : 연산면이 라벨 없이 일관 연산자로 창발함을 방향 일관성, 직교성, 예측 기여로 측정한다(5절).
- 콘볼루션 스케일 구조 : 채널 혼합을 순환 필터뱅크로 두어 스케일이 깊이를 따라 변하는 필터 사다리를 이루고, 그 파라미터 절감과 층별 역할을 측정한다(5절).
- 혼합 헤드의 순서 분화 : 인과 혼합 헤드가 라벨 없이 서로 다른 과거 거리로 분화되어 순서(통합 관계)를 나눠 담음을 거리커널로 측정한다(5절).
- 조합 판별의 정직한 측정 : 학습에서 보지 못한 어간 어미 결합형의 조립 판별을 최소쌍과 바이그램 기준선으로 측정하고, 주장이 미치는 범위를 한정한다(5절).

2. 관련 연구

인수분해 임베딩(factorized embedding, 임베딩을 두 작은 표의 곱으로 쪼갬)은 파라미터를 감소시키지만 두 인자에 서로 다른 의미역을 두지는 않는다. FiLM(특징별 선형 변조, feature-wise linear modulation) [4]과 게이팅은 별도 조건 신호로 특징을 변조하지만, 그 신호는 같은 토큰이 소유한 임베딩이 아니라 외부에서 온다. 본고는 같은 토큰이 데이터 임베딩과 연산 임베딩을 동시에 소유하고, 연산면이 항등에서 출발해 라벨 없이 연산자로 갈라진다. 이 창발을 방향 일관성으로 정량화한 점, 그리고 채널 혼합에 밀집 행렬 대신 순환 필터뱅크를 써 필터 사다리를 이룬 점이 차별점이다. 순환 연산의 정확한 전치 학습은 본 시리즈 제1편 [2]의 엔진이 제공한다.

토큰 하나에 내용 표현과 변환 표현을 같이 주는 생각은 본고가 처음이 아니다. 낱말마다 벡터(내용)와 행렬(이웃을 바꾸는 연산자)을 함께 준 모델 [5]이 있고, 덧셈으로 합쳐지는 면과 곱셈으로 합쳐지는 면을 같이 학습한 모델 [6]이 있다. 앞 토큰이 한 칸 밀려 다음 토큰의 채널을 곱셈으로 여닫는 장치도 순환 구조 모델 [7]에 이미 있다. 본고가 다른 점은 네 가지다. 첫째, 발상의 출처가 다르다. 본고의 설계는 이 선행들이 아니라 본 시리즈보다 먼저 공개된 공리 문헌 [1]에서 왔다. 그 문헌의 첫 공리는

모든 변화를 데이터 괄호와 연산자 괄호의 직교 합으로 두고, 데이터를 상태가 기록되는 곳, 연산자를 연산이 실행되는 곳으로 가른다. 본고의 바이토큰은 그 공리를 언어에 그대로 옮긴 것이다. 상태가 기록되는 곳이 더하기 자리(잔차에 실리는 데이터면)가 되고, 연산이 실행되는 곳이 곱하기 자리(게이트에 실리는 연산면)가 되며, 두 면의 직교가 두 괄호의 직교에 대응한다. 언어에도 직교 성분이 있으니 토큰을 직교쌍으로 받자는 발상이 여기서 나왔다. 공리는 배경일 뿐 본고 주장의 근거가 아니며, 근거는 측정이다. 둘째, 선행들은 어느 쪽이 연산자인지 모양(행렬 대 벡터)으로 미리 정해 놓고 시작하지만, 본고는 똑같이 생긴 임베딩 두 장에 직교와 자리(더하기 자리 대 곱하기 자리)만 정해 주고 어느 쪽이 연산자로 굳는지는 학습이 스스로 정하게 두고 지켜본다. 셋째, 본고의 연산자는 행렬이 아니라 벡터다. 낱말마다 행렬을 주면 연산자 하나가 채널 폭의 제곱만큼 비싸지만, 본고는 벡터 하나가 공유 게이트를 통해 작용하므로 연산자 비용이 채널 폭에 그친다. [7]의 한 칸 이동은 같은 임베딩에서 나온 표현을 섞는 장치라 내용과 분리된 연산자 정체성이 없지만, 본고는 내용과 별도인 연산 임베딩 표를 토큰이 소유한다. 넷째, 선행들은 과제 점수로 결과를 보고하지만 본고는 연산자의 기하 자체를 측정한다. 방향 일관성(무작위의 10.1배), 연산자끼리 서로 직교하는 비율(72.5%), 켜고 끄기 대조(교차엔트로피 2.99 대 0.20), 그리고 토큰 연산 비율의 중간 대역이 비는 분포까지가 그것이다.

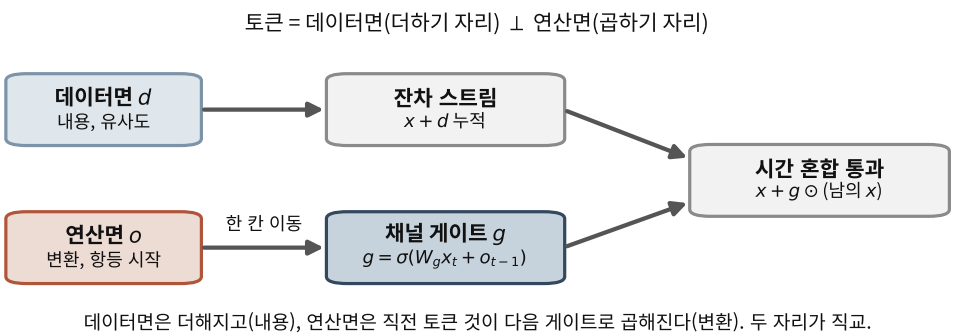
3. 표기와 구조

표 3-1. 기호 범례

기호	한글 명칭	정의 및 의미	자료형
d	데이터면 임베딩	토큰의 내용(무엇과 비슷한가). 잔차에 더해짐. 코드명 mat_w_data_axis	벡터
o	연산면 임베딩	토큰의 변환(어떻게 바꾸나). 게이트를 여닫음. 항등(0)에서 시작. 코드명 mat_w_operate_axis	벡터
g	채널 게이트	각 성분을 얼마나 통과시킬지의 0에서 1 밸브	벡터
x_t	자리 t 은닉	문맥 자리 t의 은닉 벡터	벡터
H	채널 폭	은닉 벡터의 차원. 1024	정수

토큰 하나는 데이터면 d와 연산면 o를 함께 가진다. 데이터면은 잔차 스트림에 그대로 더해지는 내용이다. 연산면은 직전 토큰의 것이 다음 토큰의 채널 게이트로 주입되어, 남의 데이터가 통과하는 양을 조절한다. 이 한 칸 이동(one-step shift)이 통합 관계(순서)의 자리이다. 전체 구조를 그림 3-1에 보인다.

그림 3-1. 직교 데이터-연산 토큰의 구조



데이터면은 더해지고(내용), 연산면은 직전 토큰 것이 다음 게이트로 곱해진다(변환). 두 자리가 직교.

토큰의 데이터면은 잔차 스트림에 더해지는 내용이고(위), 연산면은 직전 토큰의 것이 한 칸 이동해 다음 토큰의 채널 게이트로 주입되어 통과량을 조절한다(아래). 더하기 자리와 곱하기 자리라는 비대칭만 강제하고, 어느

토큰이 데이터인지 연산인지는 라벨하지 않는다.

3.1 순환 필터뱅크 채널 혼합과 필터 사다리

채널을 섞는 부분은 밀집 행렬 대신 순환 필터뱅크로 둔다. 얕은 층은 폭 넓은 큰 필터 몇 개(큰 붓), 깊은 층은 폭 좁은 작은 필터 여러 개(가는 펜)로 스케일이 사다리처럼 내려간다. 이 순환 연산의 정확한 전치는 제1편 [2]의 엔진이 제공하므로 자동미분 없이 학습된다.

4. 방법

연산면은 항등(모두 0, 곱해도 그대로)에서 시작한다. 학습이 진행되며 연산면의 신용은 게이트에서 뽑혀 한 칸 앞 토큰으로 되돌아가고, 그 정확한 전치를 통해 라벨 없이 갱신된다. 데이터면은 잔차에 더해지는 순수 내용이므로 게이트를 거치지 않는다. 이 비대칭이 두 면을 갈라 놓는 유일한 강제이다.

식 4-1. 연산면 게이트 주입과 변조

$$g_t = \sigma\big(W_g\, x_t + b_g + o_{t-1}\big), \qquad x_t^{\text{out}} = x_t + g_t \odot m_t$$

수식

기호: σ는 시그모이드(sigmoid, 값을 0에서 1로 누르는 함수), W_g는 게이트 가중치, b_g는 게이트 바이어스(2.0에서 시작해 거의 열림), o_{t-1}은 직전 토큰의 연산면, m_t는 시간혼합으로 들어온 남의 데이터, $\odot$은 성분별 곱이다.
작동 방식: 직전 토큰의 연산면 o_{t-1}이 게이트 안으로 더해져, 현재 토큰이 남의 데이터를 얼마나 통과시킬지를 조절한다. 연산면이 0이면 게이트는 원래대로(항등), 0이 아니면 통과량을 바꾼다.
실제 동작: 연산면은 "곱하기 자리"에서 게이트를 통해 작용하고 데이터면은 "더하기 자리"에서 잔차에 실린다. 두 작용이 서로 다른 자리라 직교한다. 라벨 없이도 이 자리 차이가 두 면을 갈라 놓는다.

5. 결과

5.1 연산면이 일관 연산자로 창발한다

측정

초등 단계 얼린 모델에서 측정한 결과이다.

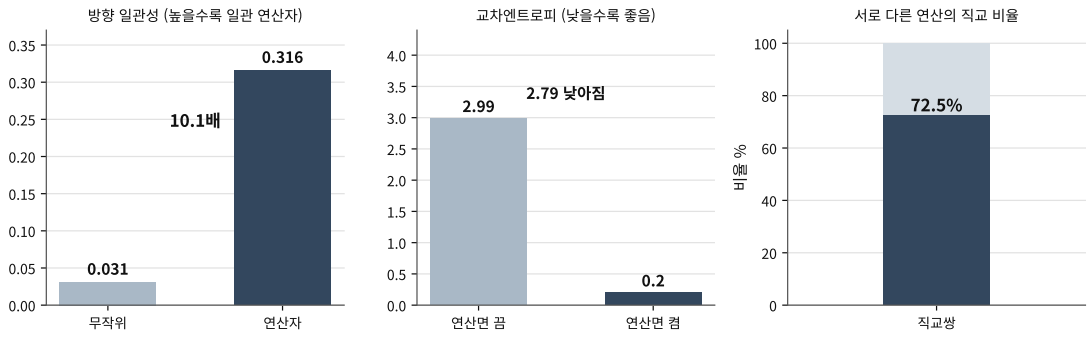
- 방향 일관성 : 같은 연산이 여러 데이터를 꺾는 방향의 평균 유사도가 0.316 이다. 무작위 기준 0.031의 10.1배 . 즉 연산자가 어떤 데이터에도 일관 방향으로 작용한다(일관 연산자).
- 연산 직교성 : 서로 다른 연산 쌍의 72.5퍼센트가 거의 직교(유사도 0.15 미만)이고, 같은 방향 쌍(0.5 초과)은 0퍼센트이다. 연산마다 고유하다.
- 예측 기여 : 연산면을 켜면 다음 낱말 예측 교차엔트로피가 2.99에서 0.20으로 2.79 낮아진

다 (동봉 자작 초등 대화 말뭉치 200창). 연산면이 예측을 크게 돕는다. 분포 밖 이식 범위는 6절에 있다.

- 토큰 정체 분포 : 연산 비율(연산면 노름을 데이터면과 연산면 노름의 합으로 나눈 값)의 분포에서 가운데(0.45에서 0.55)는 0퍼센트, 양끝(0.15 미만 또는 0.85 초과)은 34퍼센트이다. 다만 양끝의 구성은 대칭이 아니다. 데이터쪽 극 33.6퍼센트(915개)는 전부 학습에 한 번도 등장하지 않은 토큰(pad(빈자리 채움), 제어문자, 미사용 자모)이고, 학습된 토큰 1809개만 보면 데이터쪽 극 0개, 연산쪽 극 8개(0.4퍼센트)이며 나머지 전부가 연산 우세 대역(0.55에서 0.85)에 분포한다. 이 분포가 보이는 것은 두 정체로의 대칭 갈림이 아니라, 반반 섞인 토큰이 없는 가운데 공백과 학습 토큰의 연산 우세이다.
- 토큰 내부 직교 : 학습 토큰 1809개 각각에서 자기 데이터면과 자기 연산면 임베딩의 절대 코사인은 평균 0.027, 최대 0.121로 전 토큰이 거의 직교(0.15 미만) 범위이며 무작위 기준 0.031 수준이다. 직교를 강제하는 투영은 코드에 없으므로, 두 면이 같은 예측 손실로 학습되면서도 서로 상관이 생기지 않았다는 관찰이다.
- 연산자 정체 : 연산자 노릇은 문법 부류의 소유가 아니다. 조사와 어미 같은 기능 음절 30개와 명사류 음절 30개의 연산 비율 평균은 0.630 대 0.622로 쌍별 우세율 63.4퍼센트의 약한 기울기뿐이고, 방향 일관성은 기능 0.324, 명사류 0.316, 용언 어간 0.313으로 부류 간 차이가 없다(무작위 0.031). 연산자 노릇은 어휘 전체에 분산되어 있고, 어느 토큰이 연산자로 작용하는지는 곱하기 자리(앞 토큰의 연산면이 다음 자리의 게이트에 들어가는 배선)와 조합이 정한다. 본고가 강제한 것은 연산자의 자리이며, 그 자리를 무엇이 차지하는지는 학습과 문맥의 몫이다.

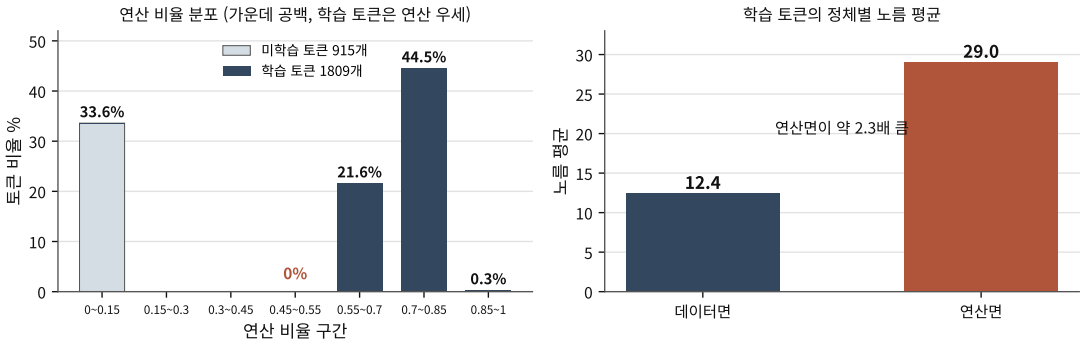
이 측정들이 그리는 직교는 두 겹이다. 표현에서 한 토큰의 데이터면과 연산면이 직교하고(토큰 내부 직교), 처리에서 필터뱅크는 데이터면만 읽으며(5.2절) 연산면은 게이트로만 들어간다(4절). 데이터면은 닮음으로 모이고 연산면은 쓰임으로 갈라지는 조직 원리의 차이는 제2편에서 다룬다.

그림 5-1. 연산자 방향 일관성과 예측 기여



연산의 방향 일관성은 무작위의 10.1배로, 연산면이 어떤 데이터에도 일관 방향으로 작용하는 일관 연산자임을 보인다(왼쪽). 연산면을 켜면 예측 교차엔트로피가 2.99에서 0.20으로 2.79 낮아진다(오른쪽). 연산면은 장식이 아니라 예측에 실제로 기여한다.

그림 5-2. 토큰 연산 비율의 분포, 가운데 공백과 학습 토큰의 연산 우세



토큰의 연산 비율 분포에서 가운데(반반 섞임)는 0퍼센트이다. 데이터쪽 극의 33.6퍼센트는 전부 학습에 등장하지 않은 미학습 토큰이고, 학습된 토큰 1809개는 전부 연산 우세 대역(0.55 이상)에 분포하며 연산쪽 극은 8개이다. 학습 토큰의 노름 평균은 연산면 29.0으로 데이터면 12.4의 약 2.3배이고, 연산면 노름이 0인 학습 토큰은 없다.

5.2 콘볼루션 필터 사다리

채널 혼합을 순환 필터뱅크로 두면 파라미터가 크게 감소한다. 순환 커널은 블록과 겹당 H 개만 있으면 되므로, $H = 1024$일 때 밀집 행렬의 약 105만 개 대비 약 1024배 적다(제1편 [2]의 측정과 동일한 구조적 사실).

측정

필터 사다리를 측정한 결과, 얕은 층에서 깊은 층으로 갈수록 필터가 넓고 적은 것(큰 붓)에서 좁고 많은 것(가는 펜)으로 내려간다.

깊이(블록-겹)	스케일	필터 수	필터 길이	데이터면 무게	연산면 무게
0-0	0	1	1024	3.49	0.00
0-1	1	2	512	4.59	0.00
1-1	2	4	256	17.57	0.00
2-1	3	8	128	8.59	0.00
3-0	3	8	128	11.71	0.00

스케일이 깊이에 따라 사다리처럼 내려간다(1024x1, 512x2, 256x4, 128x8). 그리고 모든 필터의 연산면 무게가 0이다. 즉 순환 필터뱅크는 데이터면(내용)만 읽고 연산면은 게이트가 담당한다. 두 통로가 구조적으로 갈려 있다.

그림 5-3. 깊이에 따른 필터 사다리

모든 필터의 연산면 무게 0 = 필터뱅크는 데이터면만 읽는다. 연산면은 게이트가 담당(그림 3-1).

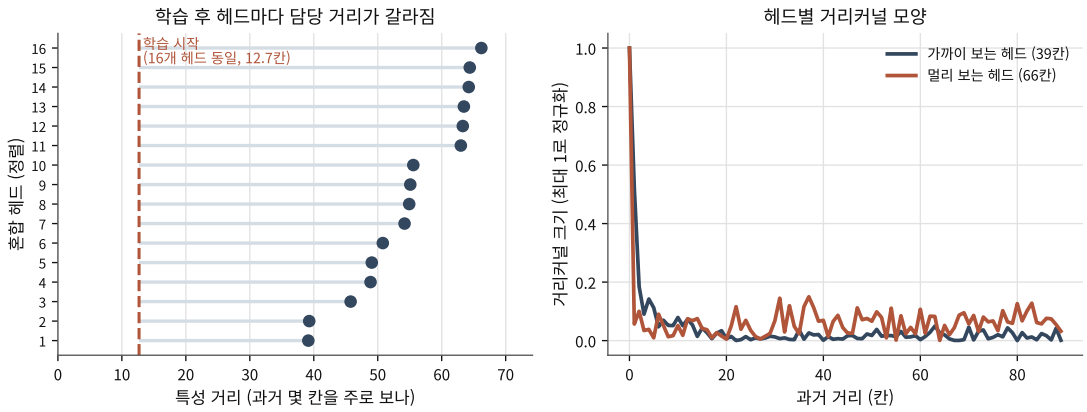
깊이가 깊어질수록 필터가 넓고 적은 것에서 좁고 많은 것으로 내려간다(필터 사다리). 모든 필터의 연산면 무게가 0이므로, 순환 필터뱅크는 데이터면의 내용만 섞고 연산면은 게이트가 따로 담당한다. 채널 혼합(콘볼루션)과 변환(연산면)이 구조적으로 분리되어 있다.

5.3 혼합 헤드가 순서를 담는다

데이터면이 계열 관계(유사도)를 맡고 연산면이 변환(추론)을 맡는다면, 통합 관계인 순서는 어디가 담는가. 이 모델에서 순서는 혼합 헤드가 담당한다. 각 블록에는 인과 마스크(과거만 보고 미래는 가리는 하삼각 가리개)를 쓰는 혼합 헤드가 여럿 있고, 각 헤드는 거리커널(distance kernel, 몇 칸 앞의 토큰을 얼마나 볼지를 거리마다 정한 가중치)을 학습한다. 헤드들이 서로 다른 거리를 맡으면, 모여서 문장의 앞뒤 순서를 여러 폭으로 읽어낸다.

학습된 모델의 혼합 헤드 거리커널을 측정했다. 헤드 16개는 학습 시작 시 모두 같은 거리커널(특성 거리 12.7칸)에서 출발한다. 학습이 끝나면 헤드마다 담당 거리가 갈라져, 과거 39칸을 주로 보는 헤드부터 66칸까지 보는 헤드로 나뉜다(한 블록 기준 헤드간 표준편차 8.6칸, 모든 블록에서 6에서 9칸). 즉 헤드들은 동일하게 출발해 순서의 서로 다른 구간을 나눠 읽도록 분화된다. 순서가 라벨 없이 혼합 헤드에 배치되는 것이다.

그림 5-4. 혼합 헤드가 순서의 서로 다른 거리를 나눠 담는다



왼쪽은 한 블록의 혼합 헤드 16개가 학습으로 담당 거리가 갈라지는 모습이다. 모두 12.7칸(붉은 세로선, 학습 시작)에서 출발해 과거 39에서 66칸으로 분화된다. 오른쪽은 가까이 보는 헤드와 멀리 보는 헤드의 거리커널

모양이다. 가까이 보는 헤드는 앞쪽 거리에 무게가 몰리고, 멀리 보는 헤드는 먼 거리까지 무게가 퍼진다. 순서(통합 관계)가 혼합 헤드에 나뉘어 담긴다.

재현

5.1절과 5.3절의 수치는 부록 C의 준비를 마친 뒤 다음 한 줄로 재생된다.

```
cd probe && python3 paper3_bitoken_probe.py
```

출력에서 방향 일관성 10.1배, 거의 직교쌍 72.5%, 교차엔트로피 끔 2.99 대 켬 0.20, 혼합 헤드 특성 거리 12.7칸에서 39~66칸 분화가 본문 수치와 대응한다.

5.4 안 본 조합의 조립 판별

직교 쌍 구조가 조합을 담는다면, 학습에서 한 번도 보지 못한 어간과 어미의 결합형도 바른 조립과 그른 조립을 갈라낼 수 있어야 한다. 이를 최소쌍(minimal pair, 같은 재료에서 한 요소만 다른 두 후보를 나란히 놓는 판별) 과제로 측정한다.

어간 20개와 받침 유무와 무관하게 형태 변화 없이 붙는 어미 12개로 결합형 240개의 격자를 만든다. 이 체크포인트의 학습이 접근한 말뭉치 전체를 토큰 열에서 전수로 훑어, 표면형이 한 번이라도 등장한 결합형을 본 조합, 0회인 결합형을 안 본 조합으로 가른다. 훑는 목록은 재현 패키지에 동봉된 자작 혼합 말뭉치 23종이다. 학습이 접근했을 수 있는 외부 텍스트 말뭉치(동화 계열)와 사전은 원저작물이 있어 패키지와 훑기에서 제외했으므로, 안 본 조합 중 일부가 그 경로로 노출됐을 가능성이 남는다. 이 한계는 아래 해석에 반영한다. 판별은 주어 프레임 다섯 개(나는, 그는, 우리는, 동생이, 아기가) 뒤에 바른 조립과 교란형을 놓고 로그확률 합을 비교해 바른 쪽이 높으면 정답으로 센다. 교란은 두 단계이다. 1단계는 어간과 어미의 순서 뒤집기(먹다가 대 다가먹)인데, 뒤집힌 열이 말뭉치에 실재해 판별력이 없는 쌍 9개(예 지가 5,834회)는 제외한다. 2단계는 어미 내부 뒤바꿈(먹다가 대 먹가다)으로, 뒤바뀐 이웃 연쇄(가다, 니다)가 말뭉치에서 오히려 흔해 이웃 통계만으로는 어려운 문항이다. 유효쌍은 본 조합 33개와 안 본 조합 198개이다.

비교 기준은 두 가지이다. 무작위 50%와, 같은 말뭉치 23종으로 만든 음절 바이그램(bigram, 이웃 두 글자의 연쇄 빈도) 기준선이다. 바이그램 기준선은 라플라스 평활(Laplace smoothing, 모든 빈도에 1을 더하는 보정)을 얹은 빈도표만으로 같은 최소쌍을 푼다. 표면 전이 통계만으로 이 과제가 얼마나 풀리는지를 이 기준선이 보여준다.

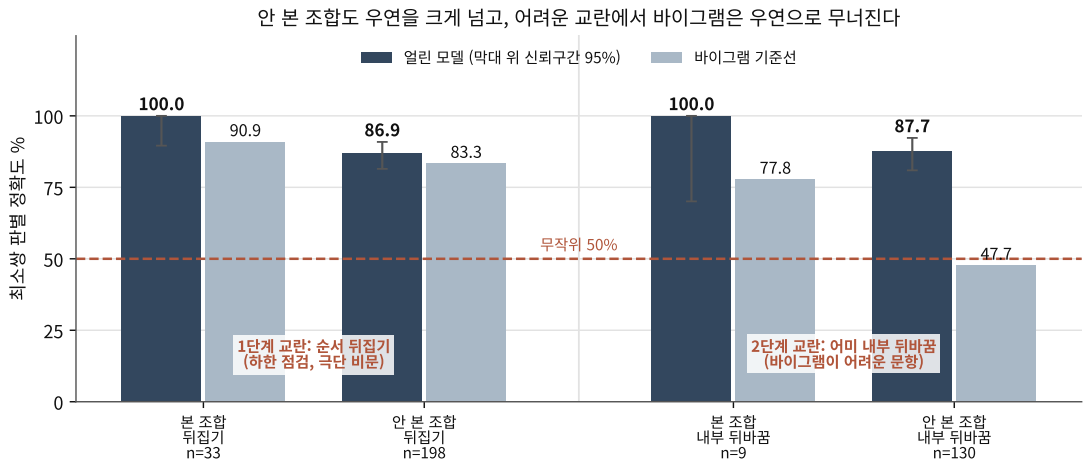
측정

측정 결과이다. p는 무작위 50% 대비 단측 이항검정(binomial test, 동전던지기로 이만큼 맞힐 확률) 값이다.

교란	묶음	n	모델 정확도	바이그램 기준선	모델 p
뒤집기	본 조합	33	100.0%	90.9%	1.2e-10
뒤집기	안 본 조합	198	86.9%	83.3%	6.7e-28
어미 내부 뒤바꿈	본 조합	9	100.0%	77.8%	2.0e-03
어미 내부 뒤바꿈	안 본 조합	130	87.7%	47.7%	1.0e-19

안 본 조합이 두 교란 모두에서 우연을 크게 넘는다(86.9%와 87.7%, p 는 10의 마이너스 19승 이하). 통짜 암기만으로는 설명되지 않는 조립 판별이 존재한다. 쉬운 1단계 교란은 바이그램 기준선도 상당히 풀지만(83.3%) 모델이 앞선다. 어려운 2단계 교란에서는 바이그램이 47.7%로 우연 수준까지 무너지고 로그확률 마진도 -0.50으로 뒤집히는 반면, 모델은 87.7%와 마진 +8.10을 유지한다. 이 교란에서 바이그램이 틀리는데 모델만 맞히는 쌍이 안 본 조합에서 60개이고 그 반대는 8개다.

그림 5-5. 조합 일반화, 모델 대 바이그램 기준선



네 묶음 모두에서 모델(짙은 막대)이 무작위 50%(점선)를 크게 넘는다. 쉬운 뒤집기 교란에서는 바이그램 기준선(옅은 막대)도 높지만, 어려운 어미 내부 뒤바꿈 교란에서는 기준선이 우연 수준으로 무너지고 모델만 유지된다. 오차 막대는 윌슨 95% 신뢰구간이다.

이 측정의 해석 한계를 명시한다. 안 본 조합은 동봉 말뭉치 23종 기준으로 표면형 전체가 0회라는 뜻이며, 훑기에서 제외한 외부 텍스트 경로로 노출됐을 가능성이 남고, 그 구성 바이그램 대부분은 학습에서 다수 목격되었다. 본 조합 판정은 부분 문자열 일치 기준이라 동형이의어(가지, 보고 같은 명사)를 포함할 수 있는 상위집합이고 33개로 표본이 작다. 이 측정이 보인 것은 통짜 암기 이점의 부재, 우연을 크게 넘는 조립 판별, 그리고 어려운 교란에서 표면 통계가 무너져도 유지되는 판별까지이다. 직교 구조가 원인이라는 인과 확인은 직교를 제거한 같은 크기 대조 모델과의 나란한 재학습 비교로 수행했으며 결과는 6절에 있다. 대조 모델은 이 판별에서 바이그램 아래로 하락한다.

재현

이 소절의 조립 판별은 부록 C의 준비를 마친 뒤 다음 한 줄로 재생된다.

```
cd probe && python3 paper3_composition_probe.py
```

출력에서 본 조합 33개 100.0%, 안 본 조합 198개 86.9%, 2단계 교란의 모델 87.7% 대 바이그램 47.7%, 진단 부분집합이 본문 수치와 대응한다.

6. 한계와 다음 측정 과제

기준선은 1인 개발, 자작 정확한 전치 역전파 엔진, 소비자 그래픽 처리장치 한 대라는 레퍼런스 클래스이다. 아래는 결함이 아니라 다음 측정 과제이다.

- 표준 트랜스포머 기준선 (실측) : 같은 월드 커리큘럼, 같은 170,000스텝, 같은 배치 32로 표준 트랜스포머 104M(어텐션과 밀집 변환, AdamW)을 학습해, 같은 창(분포 안 초등 대화, 창 128 토큰 200개, 시드 4벌)에서 나란히 측정했다. 결과와 득실은 표 6-1이다. 읽는 법은 다음과 같다. 표준의 근소한 예측 우위는 창 제곱의 전 쌍 비교와 채널 제곱의 밀집 변환이 사주는 표현 여유의 값이고, 본 엔진은 그 비용을 지불하지 않는 로그선형 혼합으로 그 10% 안에 있으면서 스텝당 70% 더 빠르다. 같은 실행 시간 기준이면 본 엔진이 스텝을 70% 더 도는 셈이다(시간 일치 비교는 다음 과제). 창 128에서는 어텐션의 제곱 비용이 아직 작으므로 0.020의 출처를 전 쌍 비교만으로 단정하지 않으며, 최적화기 관행(AdamW 대 본 엔진의 갱신 회계) 차이도 섞여 있다. 본고의 주장은 예측 우위가 아니라 이 동급 기준선 위에서 관찰된 구조다. 부수 관찰로, 이 표준 모델은 예측 CE가 낮음에도 자유 생성 표본이 문장을 이루지 못했는데, 예측 눈금과 생성 품질이 갈릴 수 있음은 별도 측정 과제로 남긴다.

  표 6-1. 표준 트랜스포머 기준선과 득실

눈금	바이토큰 104M	표준 104M	득실
예측 CE (같은 창 800개)	0.207	0.186	표준이 0.020 낮음 (상대 약 10%)
스텝 속도 (배치 32, 창 128)	80.5 ms	136.7 ms	본 엔진이 70% 더 빠름
자리 혼합 비용	상대 거리 혼합 (로그선형)	어텐션 (창 제곱)	창이 길수록 본 엔진이 이득
채널 변환 비용	필터 사다리, 순환 (선형에서 로그선형)	밀집 변환 (채널 제곱)	채널이 클수록 본 엔진이 이득

- 직교 제거 대조 (실측) : 직교를 제거한(연산면 없는) 같은 크기 모델을 같은 시드, 같은 말뭉치, 같은 170,000스텝으로 나란히 재학습해 5.4절과 같은 프로토콜로 측정했다. 대조 모델은 본 조합 90.9퍼센트(원본 100.0), 안 본 조합 76.3퍼센트(원본 86.9)로 하락하고, 원본이 바이그램 기준선 83.3퍼센트를 넘는 것과 달리 그 아래로 떨어진다. 어려운 2단계 교란에서도 80.0퍼센트(원본 87.7)다. 반면 예측 교차엔트로피는 0.28로 원본 0.21에 근접해, 재학습이 예측 일반은 거의 보상하지만 표면 통계 너머의 조합 지식은 보상하지 못한다. 즉 직교 구조는 조합 일반화에 특이적으로 기여한다. 한계: 재학습은 한 번(시드 하나)이고 GPU 비결정성이 있으므로, 반복 재학습으로 분산을 확인하는 것이 다음 과제다.

- 분포 밖 이식 : 5.1의 예측 기여 +2.79는 학습 분포 안 관찰이다. 같은 얼린 모델로 텍스트만 바꿔 같은 측정(무작위 200창, 창 표본 4벌)을 반복하면 기여는 분포 거리에 따라 감소한다. 학습에 쓰지 않았지만 문체가 같은 동봉 홀드아웃 텍스트(eval_sets.json 의 중등, 고등 각 30조각)에서는 중등 +0.32에서 +0.46, 고등 +0.40에서 +0.60으로 양수가 유지되고, 문체가 다른 성인 자연어(위키백과 무작위 75문서 22.5만 자를 유니코드 정규화와 한글, ASCII 보존 전처리 후 원자 부호화, 미지 토큰 0개)에서는 끔 4.58에서 4.69 대 켬 4.74에서 4.86으로 기여가 -0.13에서 -0.25로 소폭 음수다. 읽는 법은 다음과 같다. 연산면 게이트가 배운 것은 학습 문체의 어법에 조율된 게이트 지시라서, 안 본 텍스트라도 문체가 같으면 이식되고 문체가 다르면 지시가 어긋나 끔보다 소폭 나빠진다. 방향 일관성과 직교성 같은 기하 측정은 입력 텍스트와 무관한 모델 내부 성질이라

그대로 성립한다. 이 체크포인트는 초등 단계까지의 발달 커리큘럼 산물이라 성인 문체는 아직 학습 단계에 없는 어법이고, 이 감소는 결함이 아니라 예상되는 순서다. 커리큘럼을 실제로 이어 학습해 이를 확인했다. 발행 체크포인트에서 중등 구어 [8]와 국어사전 [9]을 얹은 단계(15,000스텝), 이어 성인 소재 자작 말뭉치를 얹은 단계(20,000스텝)를 학습하고 같은 평가 사다리를 다시 측정했다. 직접 학습한 문체는 홀드아웃에서 크게 상승했다(중등 구어 홀드아웃 +0.46에서 +1.69, 성인 소재 홀드아웃 +0.34에서 +3.34). 복습을 계속 받은 초등 대화도 +2.79에서 +3.05로 유지되었다. 성인 자연어(위키)는 사전 비중이 큰 중등 단계에서 -0.13에서 +0.10으로 양수 전환했다가, 사전 비중을 줄이고 성인 소재가 지배한 다음 단계에서 -0.42로 되돌아갔다. 백과 문체에 가장 가까운 학습 재료가 정의문체(사전)임이 특정된 것이다. 복습에 넣지 않은 문체는 단계마다 하락했는데(자작 지식문답 홀드아웃 +0.37에서 -0.99), 게이트 지시가 최신 재료로 재배치되는 간섭이며, 유지할 문체는 복습 몫에 넣어야 함을 뜻한다. 이 이동 동안 기하는 골격이 유지되었다(방향 일관성 10.1배에서 9.2배, 직교쌍 72.5에서 73.4퍼센트, 헤드 분화 39~66에서 40~66칸). 분해의 기저는 이미 형성되어 있고, 이어 학습은 그 위의 지시 내용을 재배치한다는 관찰이다. 이어 학습은 한 번(시드 하나)이며, 백과 문체를 향한 다음 과제는 정의문체 비중의 유지와 세상 지식(뉴스) 말뭉치의 추가다.

- 밀집 대비 콘볼루션 : 순환 필터뱅크를 같은 크기 밀집 채널 혼합과 나란히 놓는 비교는 다음 측정 과제이다(해당 프로브가 기본 경로만 읽어 실행 방식 보완이 필요하다).
- 변조 계열 비교 : FiLM이나 게이팅과 나란히 놓는 비교는 다음 과제이다.
- 게이트의 표현 한계 : 게이트 치역이 0에서 1이라 통과와 차단만 표현하고 부정(부호 뒤집기)을 한 홉에 표현하지 못하며, 연산면 성분이 자기 채널만 조절하는 대각 작용이라 회전이 없다. 연산면을 오일러 회전각으로, 한 단 위에서는 단위 쿼터니언으로 읽는 확장은 후속 논문(제7편)의 주제다.

7. 결론

본고는 토큰을 데이터면과 연산면 두 직교 임베딩으로 나누는 구조를 제시하고, 라벨 없이 기하만 강제해도 연산면이 일관 연산자로 창발함을 측정으로 보였다. 연산의 방향 일관성은 무작위의 10.1배, 서로 다른 연산의 72.5퍼센트가 직교, 연산면을 켜면 예측 교차엔트로피가 학습 분포 안에서 2.79 낮아지며(분포 밖 이식은 6절), 토큰 연산 비율은 중간 대역이 비고 학습된 토큰은 연산 우세 대역에 분포한다. 채널 혼합은 순환 필터뱅크로 스케일이 깊이를 따라 변하는 필터 사다리를 이루되 데이터면만 읽어, 콘볼루션과 변환이 구조적으로 분리된다. 순서는 인과 혼합 헤드가 담당하여, 헤드 16개가 동일하게 출발해 과거 39에서 66칸까지 서로 다른 거리로 분화됨을 측정했다. 학습에서 보지 못한 어간 어미 결합형의 조립 판별도 우연을 크게 넘고 어려운 교란에서 바이그램 기준선이 무너져도 유지됨을 측정했으며, 그 해석은 5.4절의 한정을 따른다. 직교를 제거한 대조 재학습에서는 이 판별이 바이그램 아래로 하락해, 직교 구조의 인과 기여가 확인된다(6절). 데이터와 연산의 직교, 그리고 그 순차 구성이 언어의 계열과 통합 관계에 각각 대응한다.

8. 후속 논문 예고

후속 논문 예고

- 제4편 — 월드먼저 발달 커리큘럼 학습 : 본편의 얼린 모델이 어떻게 학습됐는지, 언어 이전에 감각과 공간과 논리 축을 먼저 발달시키는 학습 순서가 임베딩에 축 구조를 형성하는 과정을 측정으로 다룬다.
- 제5편 — 캐시 사전 토큰화와 동적 임베딩 : 데이터면과 연산면이 얹히는 어휘층을 음절 원자와 묶음 사전의 2층 구조로 다룬다.
- 제6편 — 자기 분포와 기하 모니터링에서 창발하는 제어 신호 : 모델이 자기 출력 분포와 임베딩 기하를 읽어 뽑는 제어 신호를 측정된 통계량으로 다룬다.
- 제7편 — 회전 연산자 게이트와 쿼터니언 게이트 : 본편 게이트의 표현 한계 2종(6절)에서 출발한다. 연산면을 오일러 회전각으로 읽으면 부정이 180도로, 연산의 합성이 각도 덧셈으로 표현되며, 회전의 수반은 켤레 회전이라 닫힌형으로 나온다. 각도 덧셈은 가환이라 순서가 결과를 바꾸는 합성은 그 위의 단을 요구하고, 연산면 4성분을 단위 쿼터니언으로 읽는 쿼터니언 게이트가 그 단이다. 깊이를 뺀 1블록 소형 실측에서 게이트만 바꾼 절제 대조로, 가환 과제(mod 3)와 비가환 과제(삼각형 대칭)의 판별 2종을 다룬다.
- 마지막 편(미완) — 유사도 정리와 추론 순회의 결합 : 데이터면 유사도 정리(제2편 [3])와 연산면 추론 순회를 엮는 자기 발화. 최종 조립이 끝나지 않아 아직 다루지 않는다.

참고문헌

[1] 한혁진 (2026). Banya Framework: An Axiom-Based Science Mining Engine for Deriving Fundamental Physical Constants (반야프레임 공리 15개). Zenodo. https://doi.org/10.5281/zenodo.20301304 . (본 시리즈의 배경 자연관, 선행 공리 문헌)

[2] 한혁진 (2026). 자동미분 없이 수동으로 유도한 정확한 전치로 학습하는 언어모델 엔진. 반야 연구 시리즈 제1편. Zenodo. https://doi.org/10.5281/zenodo.21385447

[3] 한혁진 (2026). 기울기를 전혀 쓰지 않는 되새김 자기조직화 학습. 반야 연구 시리즈 제2편. Zenodo. https://doi.org/10.5281/zenodo.21385516

[4] Perez, E., Strub, F., de Vries, H., Dumoulin, V., Courville, A. (2018). FiLM: Visual Reasoning with a General Conditioning Layer. AAAI. arXiv:1709.07871.

[5] Socher, R., Huval, B., Manning, C. D., Ng, A. Y. (2012). Semantic Compositionality through Recursive Matrix-Vector Spaces. EMNLP-CoNLL.

[6] Mai, F., Galke, L., Scherp, A. (2019). CBOW Is Not All You Need: Combining CBOW with the Compositional Matrix Space Model. ICLR. arXiv:1902.06423.

[7] Peng, B. et al. (2023). RWKV: Reinventing RNNs for the Transformer Era. Findings of EMNLP. arXiv:2305.13048.

[8] 국립국어원 (2021). 국립국어원 구어 말뭉치(버전 1.2). https://kli.korean.go.kr/corpus

[9] 국립국어원. 표준국어대사전. 공공누리 제1유형. https://stdict.korean.go.kr

부록 A. 기본 설정

표 A-1. 바이토큰 설정

항목	값	항목	값
채널 폭 H	1024	연산면 초기값	0 (항등)
게이트 바이어스 초기값	2.0 (거의 열림)	연산면 주입	한 칸 이동
채널 혼합	순환 필터뱅크	스케일 하한	필터 길이 32

부록 B. 측정 환경

표 B-1. 측정에 사용한 컴퓨터 사양

구분	사양
그래픽 처리장치(GPU)	NVIDIA GeForce RTX 3090 Ti (24 GB)
운영체제	Ubuntu 24.04.4 LTS
소프트웨어	Python 3.12.3, numpy 2.5.1, scipy 1.18.0, cupy 14.1.1, CUDA 13.3.1
측정 대상	초등 단계 얼린 모델(약 17만 스텝)의 데이터면, 연산면, 필터

5절의 실측 수치는 위 환경에서 측정했다.

부록 C. 재현 안내

본고의 실측 수치는 공개 재현 패키지로 재생된다. 아래 다섯 단계를 순서대로 하면 된다.

1. 요구 환경. NVIDIA 그래픽 처리장치(GPU)와 CUDA, Python 3.12 가 필요하다. 파이썬 패키지는 압축을 푼 폴더에서 다음으로 설치한다. cupy 는 설치된 CUDA 버전에 맞는 배포판을 사용한다(예: CUDA 13 은 cupy-cuda13x).

```
pip install -r requirements.txt
```

2. 코드 받기. 압축 묶음(zip)을 내려받아 푼다. 저장소로 받으려면 github.com/ubmscoin/banya/banya_ai_paper_code 폴더다.

```
wget https://ubmscoin.github.io/banya/banya_ai_paper_code/
banya_ai_paper_code.zip
unzip banya_ai_paper_code.zip && cd banya_ai_paper_code
```

3. 모델 받기. 얼린 모델 3개(합계 471MB)는 제노도 doi.org/10.5281/zenodo.21383724 에 있다. 다음 한 줄이 내려받기와 무결성 검증(sha256, 파일 지문 대조)을 함께 한다. 손으로 받으려면 위 기록에서 세 파일을 받아 model 폴더에 넣고 `sha256sum -c checksums.sha256` 으로 검증한다.

```
cd model && bash download.sh && cd ..
```

4. 말뭉치 만들기. 말뭉치는 원문 배포 없이 생성기가 만든다. 다음 한 줄이 banya_world_data 폴더에 말뭉치 23종을 전부 생성한다.

```
cd corpus_build && bash build_all.sh && cd ..
```

5. 실측 재생. probe 폴더의 프로브가 각 절의 수치를 재생한다. 본고의 수치는 다음 프로브가 담당한다. 출력에 목표값이 함께 인쇄되므로 본문과 바로 대조된다.

프로브	재생하는 수치	확인할 출력
paper3_bitoken_probe.py	5.1 일관 연산자와 토큰 정체 분포, 5.3 헤드 분화	일관성 10.1배, 직교쌍 72.5%, 교차엔트로피 2.99 대 0.20, 양끝 34%(학습 토큰만 0.4%), 내부 직교 평균 0.027, 부류 일관성 0.324 대 0.316, 헤드 12.7칸에서 39~66칸
paper3_composition_probe.py	5.4 조립 판별	본 33개 100.0%, 안 본 198개 86.9%, 2단계 모델 87.7% 대 바이그램 47.7%
paper3_baseline_probe.py	6절 표 6-1 표준 기준선	같은 창 800개의 교차엔트로피 바이토큰 대 표준, 스텝당 밀리초

표준 기준선의 학습기 banya_bp_pytorch.py 도 동봉한다. 이 학습기는 제3편 전용이 아니라 시리즈 공통 대조 모델이다. 표 6-1의 표준 쪽 채점에는 표준 체크포인트 model/banya_bp_pytorch.pt 가 필요하며, 동봉 학습기로 학습해 만들거나 내려받는다. 동화 말뭉치 3종은 저작권으로 동봉하지 않으므로 동봉 자료만으로 재학습하면 마지막 단계의 혼합이 발행 실행과 약간 다르다.

패키지의 폴더 구성과 파일별 설명은 목차 페이지에 있다.

반야 연구 시리즈 제3편(직교 토큰, 콘볼루션 스케일). 모든 수치는 부록 B 환경에서 측정한 값이다. 배경 자연관은 선행 공리 문헌 [1]로 인용하되 이 글 주장의 근거가 아니다. 제1편(엔진), 제2편(되새김)을 인용한다. 발달 커리큘럼, 캐시 사전 토큰화, 창발 제어 신호는 제4편에서 제6편에서 다룬다.