

딸깍의 성공은 학습의 성공이 아닙니다.

산출물-이해의 역설에서 인지의 외주화와 정반탈 평가까지

생성형 AI 시대 수행과제와 평가를 다시 묻는 개념적, 실천적 탐색 칼럼

인지적 마찰과 정반탈 평가의 통합 흐름



※ 본 글에서 사용하는 '산출물-이해의 역설', '딸깍 역설', '반응형 인식 매개체', '정반탈'은 교육적 현상을 설명하기 위한 잠정적 표현입니다.

출발점: '딸깍'하면 만들어지는데 왜 이해는 남지 않습니까?

매번 글에 좋은 코멘트와 영감을 주는 질문을 남겨 주셔서 제가 더 감사할 때가 많습니다. 어제 최소 구조의 원리에 대해 글을 올리고 나서 림팔라님의 질문이 또 저를 움직였습니다. 그 분이 오렌지3와 같은 프로그램에서 바이브코딩으로 프로젝트를 바꿨더니 학생들이 별다른 고민 없이 몇 번의 클릭만으로 꽤 멋진 결과를 만들었고 그 결과를 보며 자신이 내용을 잘 안다고 착각하는 듯했다는 이야기였습니다.

저는 그 문장을 아주 오래 들여다봤습니다. 오늘 이동할 일이 많아 버스와 지하철에 있는 시간이 많았는데 이 시간 동안 줄곧 고민했습니다. 왜냐하면 이 문제가 생성형 AI 시대에 제가 가장 걱정하는 지점을 정확히 찌르고 있었기 때문입니다. 조금 웃긴 표현을 빌리자면 생성형 AI가 생기고 나서부터 우리 인간은 꽤 자주 '인지적 구두쇠(cognitive miser)'처럼 행동합니다. 아니죠. 어쩌면 인지적 구두쇠라는 게 본질일 수 있습니다. 어려운 것과 복잡한 것을 배우기 싫어하는... 수학 교사로서 이런 부분을 최근 들어 더 많이 경험합니다. 아이들이 활동을 하거나 문제를 풀다가 충분히 그럴듯한 답이 바로 주어지면 굳이 더 많은 인지 자원을 써서 검증하고 다시 구성하려 하지 않을 수 있습니다.

생성형 AI는 이 성향과 아주 잘 맞습니다. 빠르고 유창하며 그럴듯한 답을 주어 우리를 현혹시킵니다. 현혹시킨다는 표현보다는 우리에게 잘 차려진 만찬을 제공해줍니다. 심지어 막히는 순간을 거의 느끼지 못하게 합니다. 그래서 학습자는 '더 생각해야 할 이유'의 필요성을 느끼지 못하고 그대로 받아들입니다. 지난 Spin-off에서 이야기한 것처럼 이것을 자신이 한 것인양 착각합니다. 이게 아이러니 하게도 아이들이 쓴 도구가 나빠서가 아니라 오히려 도구가 너무 잘 작동하기 때문에 생기는 문제입니다.

'내가 만들었다'와 '내가 이해한다'는 더 이상 같은 말이 아닙니다

예전에는 '과제를 완성했다'와 '그 과제를 수행할 수 있다'가 어느 정도 겹쳤습니다. 친구의 답을 베끼거나 인터넷에서 결과를 가져오는 경우도 있었지만 적어도 상당수의 산출물에는 수행 시간과 시행착오가 함께 남았습니다. 그런데 생성형 AI가 들어오면서 두 문장의 거리가 급격히 벌어지기 시작했습니다.

이제 '내가 이것을 만들었다'와 '나는 이것이 왜 이렇게 만들어졌는지 안다'는 동의어가 아닙니다. 더 엄밀하게 말하면 '내가 만든 것의 핵심 구조를 이해한다'는 주장도 별도의 증거가 필요합니다.

바이브코딩은 이 현상을 가장 극단적으로 보여 줍니다. 자연어로 몇 번 요청했더니 웹사이트가 작동합니다. 그래프가 움직입니다. 데이터가 분석되고 화면도 제법 멋집니다. HTML이나 Python 또는 R로 같은 결과를 직접 만들려 했다면 훨씬 많은 시간과 시행착오가 필요했을 것입니다. 저 역시 그 결과물을 볼 때마다 놀랍니다. 이런 매력때문에 저도 Chat GPT와 Claude에게 많은 일을 맡깁니다. 하지만 과연 이것이 옳은 것인지는 매번 마음 속에서 갈등합니다.

그런데 여기서 역설이 생깁니다. 저는 이 상황을 '산출물-이해(Output-Understanding)의 역설'이라고 부르고 싶습니다. 조금 더 급진적으로는 '딸깍 역설(Mouse-Click Paradox)'이라고 부를 수 있겠습니다. 정립된 학술 용어를 주장하려는 것은 아닙니다. 산출물의 수준이 높아졌다는 사실이 학습자의 이해가 깊어졌다는 증거가 되지 않는 장면을 설명하기 위한 작업용 이름입니다.

산출물-이해의 역설을 한 장으로 보기

학생의 짧은 지시	생성형 AI의 즉시 생성	높은 수준의 산출물	학습자의 이해는 별도 문제
프롬프트 몇 줄	코드와 분석, 문장 생성	작동하는 사이트와 보고서	설명과 전이, 오류 탐지는 확인 필요

그림 1. 말쑥 역설: 산출물의 상승이 이해의 상승을 보장하지는 않는다

제가 생각하는 이해는 최소한 자신의 말로 설명할 수 있는 상태와 관련이 있습니다. 질문을 받았을 때 근거를 들어 답할 수 있어야 합니다. 낯선 상황에 적용할 수 있어야 하며 제시된 상황이나 문제, 설명에 오류가 섞였을 때 무엇이 이상한지 감지할 수 있어야 합니다. 이전 글에서 제안한 '설명, 전이, 오류 탐지, 자연 재구성'의 내재화 관문과도 이어지는 지점입니다.

인지의 외주화: 문제는 말기는 데 있지 않고 무엇을 말기인가에 있습니다

여기서 '인지의 외주화'라는 관점을 붙여 보고 싶습니다. 인지과학에서는 기억하거나 계산해야 할 부담을 메모와 기기 같은 외부 자원에 넘기는 현상을 인지적 오프로딩(cognitive offloading)이라고 설명합니다. 이것은 생성형 AI 때문에 새로 생긴 현상이 아닙니다. 우리는 오래전부터 종이에 메모했고 계산기에 계산을 맡겼으며 검색엔진으로 정보를 찾았습니다.

따라서 인지의 외주화 자체를 학습의 적으로 볼 필요는 없습니다. 오히려 잘 설계된 외주화는 제한된 인지 자원을 더 중요한 판단에 쓰게 합니다. 문제는 생성형 AI가 외주화의 '양'만 늘린 것이 아니라 외주화의 '종류'를 바꾸었다는 데 있습니다.

과거의 도구가 주로 저장과 검색 또는 계산을 대신했다면 생성형 AI는 문제를 정의하고 자료를 해석하며 설명을 만드는 일까지 대신할 수 있습니다. 더 나아가 여러 대안 중 무엇을 고를지 제안하고 그 선택을 정당화하는 문장까지 만들어 줍니다. 이 글에서는 이런 교육적 상황을 직관적으로 '인지의 외주화'라고 부르겠습니다.

인지의 외주화에서 남겨야 할 경계

표 1. 기능의 외주화와 판단의 외주화를 구분하기 위한 교육적 지도

영역	AI 활용의 위치	학습자에게 남겨야 할 것
저장, 검색, 계산, 반복 작업	적극적으로 맡길 수 있음	결과가 과제에 적합한지 확인
초안, 코딩, 분석 후보, 대안 생성	공동 수행이 적합함	비교 기준과 수정 이유를 설명
검증, 가치판단, 선택, 최종 정당화	외주화에 특히 주의	판단권과 책임을 학습자가 보유

그래서 저는 '기능의 외주화'와 '판단의 외주화'를 구분할 필요가 있다고 봅니다. 계산과 반복 코딩은 맡길 수 있습니다. 표 정리와 초안 생성도 상황에 따라 맡길 수 있습니다. 그러나 무엇을 믿을지 결정하는 일은 다릅니다. 어떤 근거가 충분한지 따지는 일도 다릅니다. 무엇을 우선할지 정하고 마지막 선택을 정당화하는 일까지 넘기면 학습자는 결과의 소유자는 될 수 있어도 지식의 판단 주체로 남기 어렵습니다.

ZPD의 '유능한 타자'와 AI는 같은 위치에 있지 않습니다

이 문제는 Vygotsky의 근접발달영역을 떠올리게 합니다. 지난 칼럼에 이어서 또 소환되었습니다. 사실 Vygotsky가 이런 것을 생각했을 때 교사는 거의 모든 정보의 소스였습니다. 전통적으로 교사나 숙련된 동료는 학습자보다 해당 과제에 대해 더 많은 지식과 맥락 판단을 가진 '유능한 타자'로 가정됩니다. 도움을 받으며 할 수 있는 것이 늘어나고 그 과정에서 내면화가 일어날 수 있다는 설명은 여전히 중요합니다.

그러나 AI는 조금 애매하고 이상한 위치에 있습니다. 어떤 순간에는 내가 예상하는 것보다 압도적으로 많은 정보를 다룹니다. 바로 다음 순간에는 너무 자연스럽게 틀린 말을 하기도 합니다. 마치 야누스의 두 얼굴 처럼요. 진실과 오류가 같은 유창성으로 제시될 수 있습니다. 더구나 AI는 자신의 답에 인식적 책임을 지지 않습니다.

그래서 저는 AI를 지난 번 Spin-off에서 이야기 했던 것처럼 '반응형 인식 매개체(responsive epistemic mediator)'라고 부르는 편이 더 적절하다고 봅니다. AI는 학습자의 질문과 표현 방식에 반응하며 사고의 범위를 넓힙니다. 동시에 '이 결과를 믿어도 되는가'라는 부담을 다시 학습자에게 돌려줍니다. 도움을 준다는 점에서는 비계와 닮았지만 그 도움 자체를 검증해야 한다는 점에서는 전통적 비계와 같지 않습니다.

AI가 할 수 있게 만든 것과 내가 이해하게 된 것은 다릅니다

AI와 함께 가능한 수행 범위가 커졌다는 사실만으로 학습자의 내재 구조가 같은 크기로 확장되었다고 볼 수는 없습니다.

그래서 필요한 것은 '어려운 과제'보다 '생각하지 않으면 못 넘어가는 구조'입니다

이 지점에서 수행평가자료의 설계 질문도 바뀌어야 합니다. 저는 이제 '얼마나 어려운 문제를 줄 것인가'보다 '어느 지점에서 학생의 판단 없이는 다음 단계로 넘어갈 수 없게 할 것인가'가 더 중요해졌다고 생각합니다.

앞선 칼럼에서 언급한 인지적 마찰은 학생을 괴롭히기 위한 장치가 아닙니다. 인지의 외주화를 막기 위한 금지 장치도 아닙니다. 외주화된 사고를 다시 학습자의 판단으로 회수하는 체크포인트에 가깝습니다. 그러니까 '내가 지금 이 상황에 대해서 얼마나 알고 있는가?'를 스스로 점검하는 최소한의 장치입니다.

특히 사실판단만으로 끝나는 과제는 AI가 상당 부분을 대신할 수 있습니다. 반면 여러 대안 중 무엇을 선택할지 정하는 과제는 다르며 어떤 기준을 우선할지 묻는 과제도 다릅니다. 수학과 통계, 코딩처럼 정답이 비교적 분명한 분야에서도 이런 판단은 설계할 수 있습니다. 어떤 분석을 채택할 것인지 묻고 오류를 어느 수준까지 허용할지 결정하게 할 수 있습니다. 정확성과 설명 가능성 중 무엇을 우선할지도 논의할 수 있습니다.

인지적 마찰을 수행자료 안에 배치하는 네 단계

1. 예측	2. 검증	3. 반박	4. 정당화
AI를 쓰기 전에 내 생각을 먼저 남긴다	AI 결과에서 의심할 지점을 찾는다	다른 자료와 반례로 결과를 흔들어 본다	최종 선택과 수정 이유를 자기 언어로 설명한다

그림 2. 예측에서 정당화까지: 외주화된 사고를 다시 학습자의 판단으로 회수하는 구조

첫 번째 마찰: AI보다 먼저 예측하게 합니다

가장 먼저 넣고 싶은 장치는 '예측을 먼저 시키는 것'입니다. AI의 가장 큰 부분이 주어진 정보를 갖고 예측하는 것입니다. AI에게 묻기 전에 학생이 자신의 예상부터 기록하게 합니다. 통계 프로젝트라면 두 변수 사이의 관계를 예상하게 할 수 있습니다. 상관계수의 부호와 대략적인 크기를 먼저 적게 할 수도 있습니다. 어떤 그래프가 나올지 스케치하게 해도 좋습니다.

중요한 것은 정답을 맞히는 일이 아닙니다. AI의 답을 보기 전에 자신의 인지적 위치를 만들어 놓는 일입니다. 저는 이것을 검증 기능에 작은 스위치를 켜는 과정이라고 생각합니다. 그래야 결과를 본 뒤 '내가 생각한 것과 다른데?'라는 충동이 생깁니다.

이 장면은 Piaget의 동화와 조절을 떠올리게 합니다. 기존 구조로 새 결과를 설명할 수 있다면 동화가 일어납니다. 설명이 되지 않는다면 구조를 바꿔야 합니다. 여기서 중요한 것은 AI가 정답을 주었다는 사실이 아니라 그 결과가 학습자의 기존 예상과 부딪히는 순간을 남겨 두는 것입니다.

두 번째 마찰: AI의 결과를 바로 채택하지 못하게 합니다

두 번째는 AI의 결과를 곧바로 채택하지 못하게 하는 것입니다. 수행지에 'AI가 제안한 분석'을 적게 한 뒤 '동의하는 부분'을 따로 적게 합니다. 이어서 '의심되는 부분'과 '검증해야 할 부분'을 기록하게 합니다. 마지막으로 실제로 어떤 방법으로 검증했는지와 검증 뒤 판단이 어떻게 달라졌는지를 남기게 합니다.

이렇게 하면 AI의 위치가 달라집니다. AI는 정답을 알려 주는 존재가 아니라 검증해야 하는 제안자가 됩니다. 저는 이 전환이 중요하다고 생각합니다.

수행자료에 남길 수 있는 여섯 칸

AI의 제안 → 내가 동의한 부분 → 의심한 부분 → 검증할 부분 → 실제 검증 → 수정한 최종 판단

세 번째 마찰: '틀릴 가능성'을 과제 안에 정상적으로 넣습니다

세 번째는 틀릴 가능성을 예외가 아니라 정상적인 학습 조건으로 만드는 것입니다. 'AI의 답에서 가장 의심스러운 문장 하나를 찾으시오'라고 묻습니다. '이 답이 틀렸다고 가정하면 어디가 가장 먼저 무너지는가'라고 물을 수도 있습니다. 같은 질문을 다른 방식으로 던졌을 때 결과가 달라지는지도 비교하게 할 수 있습니다.

저는 학생에게 AI를 믿으라고도 하고 싶지 않습니다. 그렇다고 무조건 믿지 말라고 가르치고 싶지도 않습니다. 대신 의심할 이유를 찾게 하고 싶습니다. 근거를 모은 뒤 신뢰 수준을 조정하는 경험이 필요합니다. 'AI는 틀릴 수 있다'는 경고문을 읽는 것보다 검증하지 않으면 다음 단계로 갈 수 없는 과제를 경험하는 편이 훨씬 강합니다.

바이브코딩이라면 더 재미있는 활동이 가능합니다. 학생이 만든 사이트를 다른 모둠에 넘긴 뒤 '이 사이트가 틀린 결론을 내도록 데이터를 넣어 보세요'라고 요청합니다. 극단값을 넣어 보거나 결측치를 넣을 수도 있습니다. 표본 구성을 바꾸거나 사용자가 오해할 만한 입력을 시도할 수도 있습니다. 그러면 목표가 '사이트를 만드는 것'에서 '내가 만든 시스템이 어느 조건에서 유효한지 이해하는 것'으로 이동합니다. 제 교과인 수학에서는 일부러 AI에게 수학문제를 틀리게 풀게 시킵니다. 그리고 나서 아이들에게 검증하게 합니다. 이 '틀림'이라는 것을 이야기하는 순간 우리 아이들은 어느때보다도 많은 것을 생각해야 합니다.

교사 연수도 도구보다 먼저 학습관과 평가관을 다뤄야 합니다

사실 저는 이런 내용을 교사 연수에서 더 많이 다뤄야 한다고 생각합니다. 도구 사용법만 가르치는 연수는 빠르게 낡습니다. 오늘의 인기 도구가 몇 달 뒤에는 다른 도구로 바뀔 수 있기 때문입니다. 그 아래에 '학습이 무엇인가'라는 질문이 있어야 합니다. 어떤 인지 기능을 AI에 맡길지 결정해야 합니다. 학생의 판단을 어디에

남길지도 정해야 합니다. 평가에서 무엇을 학습의 증거로 볼지도 논의해야 합니다. 이 바닥이 없으면 기술은 화려하지만 수업은 약해집니다.

저는 우리 교육이 새로운 교수학습 방법을 받아들일 때 철학보다 형식을 먼저 수입하는 경우가 있다고 느낍니다. 거꾸로 수업(flipped learning)도 학습자 주도성과 교실 시간의 재구성이라는 철학보다 '영상을 미리 보는 수업'으로 축소되어 이해된 장면이 있었습니다. 거꾸로 수업이 지향하는 가치나 목표는 사실 꽤 괜찮은 것이라고 볼 수 있습니다. 하지만 그것을 이식함에 있어서 앞서 말한 것처럼 철학과 가치를 무시한채 그저 형식과 기술만을 수입했기에 그 숭고한 가치와 철학이 우리나라에서 꽃피우지 못하게 된 것입니다. 물론 이것은 지극히 제 개인적 해석입니다. 다만 생성형 AI 활용도 같은 길을 반복할 수 있다는 우려는 분명합니다. 프롬프트 작성법과 앱 제작법만 남고 학습의 의미와 지식의 소유를 묻는 질문이 사라지면 도구는 남지만 교육은 앓아집니다.

물론 저도 웹앱과 홈페이지 등을 만들고 있지만 이점에 대해서 늘 경계하고자 노력합니다. 바이브 코딩으로 인해 많은 결과물들이 나오는 것은 환영할만한 일입니다. 하지만 이것을 만들기 전에 우리는 우리 자신에게 자문해봐야 합니다. '과연 이것은 어떤 가치와 철학을 갖고 만든 것인가?' 이 질문이 거창할 수 도 있겠습니다. 하지만 이 가치와 철학이라는 것은 내 수업에서 어떻게 주어진 도구를 어떤 목적을 갖고 사용할 것인지에 대한 성찰만 있어도 대개 해결됩니다.

완성된 결과가 아니라 '결정의 흔적'을 평가해야 합니다

예전 수행평가는 결과물을 많이 봤습니다. 웹사이트가 잘 작동하는지 확인했습니다. 그래프가 정확한지 봤습니다. 발표자료의 완성도도 평가했습니다. 그러나 AI 시대에는 표면적 완성도가 학습자의 이해를 대신 증명하지 못합니다. 그래서 저는 '처음에는 무엇을 생각했는가?'를 묻고 싶습니다. 'AI의 어떤 제안을 거절했는가'도 봐야 합니다. 왜 거절했는지 설명하게 해야 합니다. 어떤 오류를 발견했는지와 무엇을 수정했는지도 중요합니다. 마지막 판단의 근거가 무엇인지까지 확인해야 합니다.

즉 산출물(output)보다 판단의 궤적(trjectory)을 평가하는 것입니다. 이것은 단순히 과정 점수를 더 주자는 이야기가 아닙니다. 작업량이 아니라 인식적 결정의 질을 평가 단위로 삼자는 제안입니다. 얼마나 많은 인지를 외주화했는지를 벌점화할 필요도 없습니다. 대신 외주화 이후에도 학습자가 검증권과 선택권을 보유했는지도 봐야 합니다. 최종 정당화의 책임이 누구에게 남았는지도 확인해야 합니다.

마지막 전환: 정반합보다 '정반탈'의 질문을 던져 봅니다

여기서 수행평가의 틀을 한 번 더 바꿔 보고 싶습니다. 편의상 '정반합(正反合)'과 대비해 '정반탈(正反脫)'이라는 표현을 제안해 봅니다. 철학사에서 정립된 헤겔의 공식 용어를 차입하거나 비판하는 것이 아닌 기존 평가 질문에서 벗어나 보자는 일종의 교육적 비유에 가깝습니다.

생성형 AI 수행평가에서 정반합식 접근은 현실적입니다. 'AI 활용 능력도 평가해야 한다'는 입장과 '학생 고유의 역량을 구분해야 한다'는 입장을 함께 반영합니다. 그래서 결과물과 AI 활용 과정을 모두 평가합니다. 저 역시 대학원 과제를 채점할 때 이런 방식을 상당 부분 사용합니다.

그러나 정반탈은 두 입장을 적절히 섞는 데서 멈추지 않습니다. '학생과 AI의 기여를 어떻게 나눌 것인가'라는 질문 자체를 벗어나려 합니다. 학생 70%, AI 30%처럼 기여도를 분할하기보다 평가 단위를 '누가 만들었는가'에서 '학생이 무엇을 판단했는가'로 옮기는 것입니다.

정반합에서 정반탈로

표 2. 정반합식 평가와 정반탈식 평가의 차이

비교 질문	정반합식 평가	정반탈식 평가
핵심 질문	학생과 AI의 기여를 어떻게 함께 볼 것인가	기여도 분할이라는 질문을 어떻게 넘어설 것인가
평가 단위	결과물과 AI 활용 과정	학생의 판단과 검증, 수정, 정당화
AI 사용량	기여 비율을 구분하려는 경향	사용량보다 판단권의 보유를 확인
최종 증거	완성도와 과정 기록	판단의 궤적과 자기 설명

AI가 결과의 상당 부분을 만들었더라도 학생이 그 결과를 비판적으로 검토했다면 이야기는 달라집니다. 여러 대안을 비교하고 한 가지를 선택했다면 그것도 학습의 증거가 됩니다. 잘못된 부분을 수정하고 마지막으로 자신의 결정을 정당화할 수 있다면 더 강한 증거가 됩니다. 정반탈에서 핵심은 AI를 덜 쓰는 것이 아니라 판단권을 끝까지 보유하는 것입니다.

결국 남겨야 할 것은 작업이 아니라 판단입니다

AI 사용을 제한한다고 반드시 사고가 일어나는 것은 아닙니다. AI를 자유롭게 사용한다고 반드시 사고가 사라지는 것도 아닙니다. 인지의 외주화가 많다고 해서 언제나 얕은 학습이 되는 것도 아닙니다. 문제는 외주화의 경계가 어디에 그어져 있는가입니다.

AI를 사용하는 동안에도 학습자가 결정해야 할 순간이 남아 있어야 합니다. 먼저 예측해야 합니다. 무엇을 받아들일지 선택해야 합니다. 이상한 점을 의심해야 합니다. 실제로 검증해야 합니다. 때로는 AI의 제안을 버려야 합니다. 마지막에는 자신의 판단을 설명해야 합니다.

그 순간들이 사라지면 아무리 멋진 결과물이 나와도 학습은 얕아질 수 있습니다. 반대로 AI가 계산과 제작의 상당 부분을 담당하더라도 외주화한 결과를 다시 판단으로 회수하는 순간을 수행자료 안에 의도적으로 배치한다면 깊은 학습이 가능할 수 있습니다.

좋은 수행자료의 질문은 달라져야 합니다

'학생에게 무엇을 하게 할 것인가'에서 멈추지 않고 '무엇을 AI에 맡겨도 되는가'와 '어느 순간에 학생의 판단 없이는 더 이상 진행되지 않게 할 것인가'를 함께 물어야 합니다.

한 문장으로 정리하면

생성형 AI 시대의 좋은 수행과제는 AI를 덜 쓰게 만드는 과제가 아니라 판단을 외주화할 수 없게 만드는 과제입니다.

참고문헌

Fiske, S. T. & Taylor, S. E. (1984). Social Cognition. Addison-Wesley.

Gao, L. & Zhang, F. (2026). Cognitive offloading and metacognitive calibration in generative AI-mediated doctoral learning: A grounded theory study of accountable reconstruction. Frontiers in Psychology, 17, 1920947.

Liu, C.-C. et al. (2026). The Design Framework for Pedagogical Agents Supporting Constructivist Learning with Generative AI. Computers & Education, 105744.

Piaget, J. (1977). The Development of Thought: Equilibration of Cognitive Structures. Viking.

Risko, E. F. & Gilbert, S. J. (2016). Cognitive offloading. Trends in Cognitive Sciences, 20(9), 676-688.

Vygotsky, L. S. (1978). Mind in Society: The Development of Higher Psychological Processes. Harvard University Press.

Wood, D. & Bruner, J. S. & Ross, G. (1976). The role of tutoring in problem solving. Journal of Child Psychology and Psychiatry, 17(2), 89-100.