

Azure Prebuilt Model 기초 분석

모델 개요

- 모델 ID: `prebuilt-receipt`
- 최신 버전: v4.0 (GA)
- 지원 문서 유형: 인쇄된 영수증, 손으로 쓴 영수증, 단일 페이지 호텔 영수증 등
- 지원 형식: JPEG, PNG, PDF, TIFF 등
- 입력 요구 사항:
 - 이미지 크기: 최소 50x50 픽셀, 최대 10,000x10,000 픽셀
 - 파일 크기: 무료(F0) 요금제는 최대 4MB, 유료(S0) 요금제는 최대 500MB
 - PDF 및 TIFF의 경우 최대 2,000페이지 처리 가능 (무료 요금제는 처음 두 페이지만 처리)

추출 가능한 필드

Receipt 모델은 다음과 같은 필드를 추출합니다:

- **MerchantName:** 상점 이름
- **MerchantPhoneNumber:** 상점 전화번호
- **MerchantAddress:** 상점 주소
- **TransactionDate:** 거래 날짜
- **TransactionTime:** 거래 시간
- **Total:** 총 거래 금액
- **Subtotal:** 세금 전 금액
- **Tax:** 세금 금액
- **Tip:** 팁 금액
- **Items:** 구매 항목 목록
 - **Description:** 항목 설명
 - **Quantity:** 수량
 - **Price:** 단가
 - **TotalPrice:** 항목 총액

또한, v4.0에서는 `ReceiptType`, `TaxDetails.NetAmount`, `TaxDetails.Description`, `TaxDetails.Rate`, `CountryRegion` 등의 필드와 일반 호텔 영수증의 VAT 테이블 추출을 지원합니다.

Supported document fields 기본

Field	Type	Description	Example
<code>MerchantName</code>	<code>string</code>	Name of the merchant issuing the receipt	Contoso
<code>MerchantPhoneNumber</code>	<code>phoneNumber</code>	Listed phone number of merchant	987-654-3210
<code>MerchantAddress</code>	<code>address</code>	Listed address of merchant	123 Main St Redmond WA 98052
<code>Total</code>	<code>currency</code>	Full transaction total of receipt	\$14.34
<code>TransactionDate</code>	<code>date</code>	Date the receipt was issued	June 06, 2019
<code>TransactionTime</code>	<code>time</code>	Time the receipt was issued	4:49 PM
<code>Subtotal</code>	<code>currency</code>	Subtotal of receipt, often before taxes are applied	\$12.34
<code>TotalTax</code>	<code>currency</code>	Tax on receipt, often sales tax or equivalent	\$2.00
<code>Tip</code>	<code>currency</code>	Tip included by buyer	\$1.00
<code>Items</code>	<code>array</code>		
<code>Items.*</code>	<code>object</code>	Extracted line item	1Surface Pro 6\$999.00\$999.00
<code>Items.*.TotalPrice</code>	<code>currency</code>	Total price of line item	\$999.00
<code>Items.*.Description</code>	<code>string</code>	Item description	Surface Pro 6
<code>Items.*.Quantity</code>	<code>number</code>	Quantity of each item	1
<code>Items.*.Price</code>	<code>currency</code>	Individual price of each item unit	\$999.00
<code>Items.*.ProductCode</code>	<code>string</code>	Product code, product number, or SKU associated with the specific line item	A123
<code>Items.*.QuantityUnit</code>	<code>string</code>	Quantity unit of each item	
<code>CountryRegion</code>	<code>countryRegion</code>	Country or region where the receipt was issued	USA
<code>ReceiptType</code>	<code>string</code>	Type of receipt	Supplies
<code>Payments</code>	<code>array</code>		

Payments.*	object	Extracted payment	Cash \$14.34
Payments.*.Method	string	Method of payment	Cash
Payments.*.Amount	currency	Amount of payment	\$14.34
TaxDetails	array	List of tax details	
TaxDetails.*	object	A single tax detail	Sales Tax(10%) \$10.00 \$1.00 \$11.00
TaxDetails.*.Amount	currency	The amount of the tax detail	\$1.00
TaxDetails.*.Rate	number	The rate of the tax detail	10%
TaxDetails.*.NetAmount	currency	The net amount before tax.	\$10.00
TaxDetails.*.Description	string	The description of the tax detail	Sales Tax

영수증 유형	ReceiptType 예시	추출 필드 (공통)	추가 필드 (특화)
기본 영수증 (receipt)	Supplies, Other	- MerchantName- MerchantPhoneNumber- MerchantAddress- TransactionDate- TransactionTime- Total- Subtotal- TotalTax- Tip- Items (Description, Quantity, Price, TotalPrice)- Payments (Method, Amount)- TaxDetails (Amount, Rate, NetAmount, Description)- CountryRegion- ReceiptType	없음
식사 영수증 (retailMeal)	Meal	- (기본 필드 동일)	- Items..ProductCode - Items..QuantityUnit
신용카드 영수증 (creditCard)	Other (분류명 Other로 반환)	- (기본 필드 동일)	- Payments 배열 (Method, Amount)
주유소 영수증 (gas)	Fuel&Energy.Gas	- (기본 필드 동일)	- TaxDetails (더 세부적으로 구성)
주차 영수증 (parking)	Transportation.Parking	- (기본 필드 동일)	- TaxDetails (더 세부적으로 구성)
호텔 영수증 (hotel)	Hotel	- MerchantName- MerchantPhoneNumber- MerchantAddress- Total- CountryRegion- ReceiptType- TaxDetails	- Balance- ArrivalDate- DepartureDate- MerchantAliases- Items..Date - Items..Category

		(Amount, Rate, NetAmount, Description)	
--	--	--	--

Azure Receipt Prebuilt 모델 한국어 지원 범위

영수증 유형	한국어 지원 여부	비고
기본 영수증 (receipt)	✅ 지원함	일반 영수증 (식당, 상점 등)
식사 영수증 (retailMeal)	✅ 지원함	한국어 메뉴 항목 등 인식 가능
신용카드 영수증 (creditCard)	✅ 지원함	결제 방식, 금액 등 한국어 인식 가능
주유소 영수증 (gas)	✅ 지원함	한국어 항목 인식 가능
주차 영수증 (parking)	✅ 지원함	한국어 인식 가능
호텔 영수증 (hotel)	❌ 지원 안 함	한국어 미지원 (영어 기반만 지원)

Azure Prebuilt Receipt 모델 결과 JSON 구조 분석

1. 전체 구조

- **status:** 분석 성공 여부 (`succeeded`)
- **createdDateTime, lastUpdatedDateTime:** 분석 요청 및 완료 시간
- **modelId:** 사용된 모델 ID (예: `prebuilt-receipt`)
- **analyzeResult:** 실제 분석 결과

2. 주요 필드

(1) content (텍스트 전체)

- OCR로 추출한 전체 텍스트가 연속된 문자열로 저장됨.
- 줄바꿈이나 포맷 없이 모든 내용을 담고 있음.

(2) pages

- 각 페이지별로 OCR 결과 저장.
- **pageNumber, width, height, angle** 등의 메타데이터 포함.
- **words:**
 - OCR로 추출된 단어 단위의 정보.
 - 각 단어마다:
 - **content:** 추출된 단어

- **polygon**: 단어의 위치 좌표 (4개의 점)
- **confidence**: 신뢰도 (0~1)
- **span**: 전체 content에서 해당 단어의 위치 (offset, length)
 - 한글은 음절 단위로 쪼개짐 (예: "코리아세븐" → '코', '리', '아', '세', '븐')
- **lines**:
 - 줄 단위의 정보.
 - **content**: 해당 줄 전체 텍스트.
 - **polygon**: 줄 전체의 위치 좌표.
 - **spans**: **content** 전체에서 이 줄이 어디에 위치하는지 (offset, length)
- **angle**: 문서가 회전된 각도 (전처리로 deskewing 수행됨).

3. span과 polygon 흐름 설명

- **polygon**은 화면상 위치 (이미지의 픽셀 좌표).
- **span**은 **content**에서 해당 단어나 줄의 시작 위치 (offset)와 길이 (length).
- 예시 흐름:
 - 전체 content: "세계 1등 편의점 (주)코리아세븐..."
 - word의 **span**: (offset: 0, length: 1) → 첫 글자 "세"
 - **polygon**: 이미지에서 "세"의 위치 좌표.

이렇게 content와 이미지 위치 정보가 서로 연결되어 있음.

4. documents (추출된 주요 필드)

- **docType**: 자동으로 분류된 영수증 타입 (예: receipt)
- **fields**:
 - 모델이 추출한 주요 필드 값들.
 - 예시:
 - **MerchantName**: "코리아세븐"
 - **TransactionDate**: "2020-06-09"
 - **Total**: "₩13,820"
 - **Items**: 항목별 제품명, 가격, 수량 등.

5. styles

- 텍스트 스타일 정보 (예: Bold, Italic).
 - 한국어 영수증에서는 활용도가 낮을 수 있음.
-

GUI 화면과의 연계

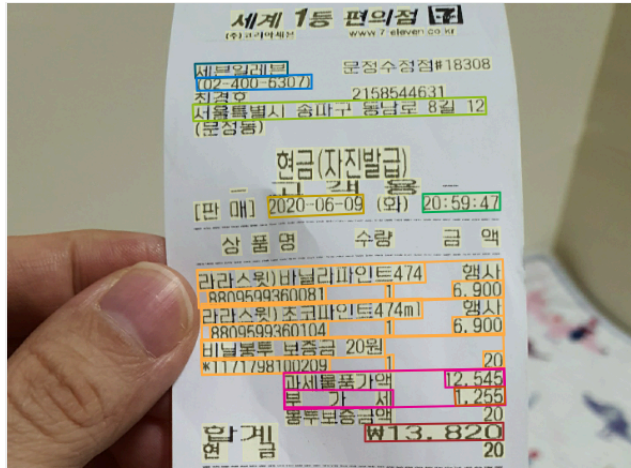
- **GUI의 필드 영역:**
 - documents > fields에 해당.
 - 예: GUI에서 **Total** , **MerchantName** 필드가 출력된 부분.
- **GUI의 원본 OCR 영역:**
 - content, words, lines에 해당.
 - 필드로 인식되지 않은 텍스트들도 포함됨.
- **GUI 상의 바운딩 박스:**
 - words/lines의 polygon 좌표를 시각화한 것.

테스트 결과



테스트 환경 요약

- **모델:** Azure Prebuilt Receipt (receipt.retailMeal 자동 분류)
- **설정:**
 - **전처리 없음** (원본 이미지 사용)
 - 이유: 모델에서 **앵글 보정 등 자체 전처리**를 수행한다고 판단
- **데이터:**
 - **Train 10장 / Test 10장**



<Train_image_01>

✓ Train_image_01.jpg 분석 결과

항목	결과	비고 (문제점)
DocType	receipt.retailMeal	적절 (편의점 식품 구매 → Meal로 분류 OK)
CountryRegion	KOR (99.5%)	정확
Items	3개 항목 인식	정상
MerchantAddress	서울특별시 송파구 동남로 8길 12	세부 필드 오류: Road: 동남로 8길 → OKCity: 서울특별시 송파 → 잘못된 CityDistrict: 구 → 잘못된 (송파구로 잡혀야)
MerchantName	세븐일레븐	정상
MerchantPhoneNumber	+8224006307	정상
ReceiptType	Supplies	Supplies도 맞긴 하지만 → Meal 더 적합할 수도 있음
TaxDetails	정상 (금액, 설명, 네트 금액 정확)	과세물품가액/부가세 라인 제대로 인식
Total	₩13,820	정확
TotalTax	₩1,255	정확
TransactionDate	2020-06-09	정확
TransactionTime	20:59:47	정확
특이사항 (봉투 보증금)	누락	커스텀 모델 필요성 판단 가능 (회귀 항목)

🔍 문제점 요약:

1. 주소 필드 오류 (City / CityDistrict)

- City: "서울특별시 송파" → "서울특별시" 만 잡혀야

- CityDistrict: "구" → "송파구" 로 인식 필요

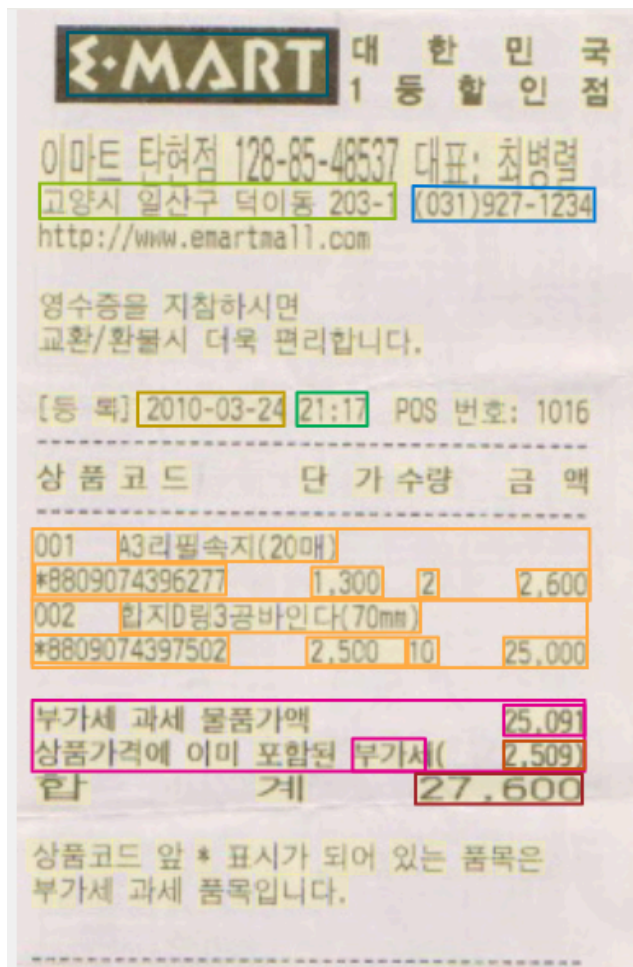
→ 한국 주소 포맷에 완벽하게 맞지 않음

2. ReceiptType 세부 분류 애매

- Supplies 로 나왔지만 → Meal도 적절할 수 있음
→ 식품 중심인지, 잡화 포함인지에 따라 판단 필요

3. 봉투 보증금 미인식

- 특이 항목 → Prebuilt 모델로는 한계 → 커스텀 모델 필요



<Train_image_02>

✓ Train_image_02.jpg 분석 결과

항목	결과	비고 (문제점)
DocType	receipt.retailMeal	무난 (편의점 + 식품 구매 → Meal 적절)
CountryRegion	KOR (99.5%)	정확
Items	2개 항목 인식	정상 (수량, 금액 정확)

MerchantAddress	고양시 일산구 덕이동 203-1	세부 필드 오류:House: 고양시 일산구 → 잘못됨 (일산구만 되어야)Suburb: 덕이동 → 정상 StreetAddress: 203-1 → 정상
MerchantName	&.MART (93.0%)	E마트 인식 오류 (E가 로고화 되어 그림처럼 처리됨)
MerchantPhoneNumber	+82319271234	정상
ReceiptType	Supplies (98.5%)	Supplies도 가능하나 Meal로 잡는 것도 고려
TaxDetails	정상 (금액, 설명, 네트 금액 정확)	OK
Total	₩27,600	정확
TotalTax	₩2,509	정확
TransactionDate	2010-03-24 (99.5%)	날짜 비현실적 (이건 테스트용 문서 특성일 가능성 높음)
TransactionTime	21:17:00	정확

문제점 요약:

1. 주소 필드 오류 (House)

- House: "고양시 일산구" → "일산구" 만 남겨야
- 한국 주소 체계의 구조적 차이로 추정됨

2. 상점 이름 오류 (MerchantName)

- &.MART → E마트 인식 실패 (로고 형태로 처리됨)
- 후처리로 보정 필요 (ex: 패턴 매칭 or 커스텀 OCR)



<Train_image_03>

✓ Train_image_03.jpg 분석 결과

항목	결과	비고 (문제점)
DocType	receipt	적절 (아이템 없는 단순 영수증 → retailMeal 아님)
CountryRegion	KOR (99.5%)	정확
Items	없음	문제 없음 (원본에도 항목 없음)
MerchantName	CU을지로파인애비뉴점 (98.2%)	정확
ReceiptType	Communication (92.8%)	매매:통신비 관련 영수증으로 인식 → 편의점이면 Supplies or Meal이 더 적절
Subtotal	₩1,819 (99.4%)	정확
TaxDetails	정상:Amount: ₩181 , Description: 부가세	정확

TotalTax	₩181 (96.3%)	정확
TransactionDate	2019-06-11 (99.5%)	정확
TransactionTime	17:41:08 (99.5%)	정확

문제점 요약:

1. ReceiptType 분류 오류 (Communication)

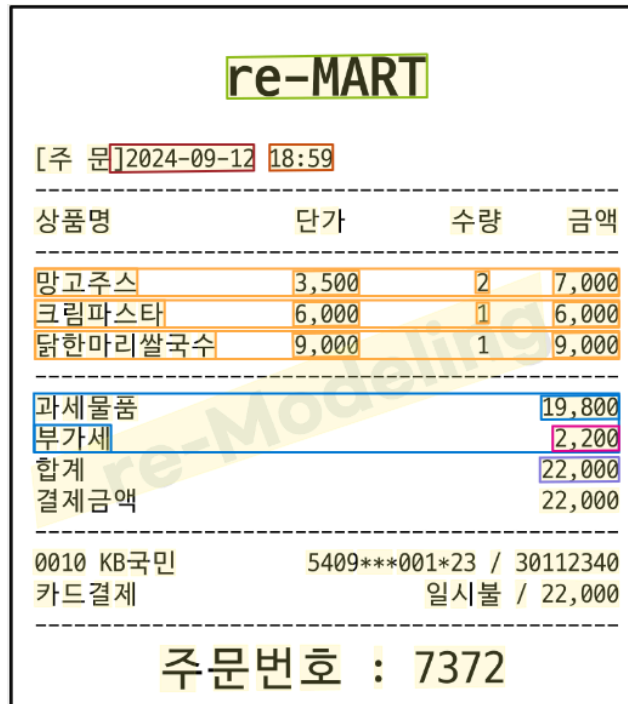
- 편의점인데 **Communication**으로 잡힘
- 단순 합계/세금 정보만 있어서 잘못 분류된 듯
 - 항목(Items) 부재가 영향을 준 가능성
 - Supplies나 Meal로 분류되면 더 적절할 것

2. 전자영수증 특성:

- 텍스트 선명 (OCR 품질 좋음)
- 아이템 없음 → **Prebuilt 모델 한계 확인** (아이템 기반 분류에 의존)

향후 개선 방향:

- ReceiptType 분류 보완:
 - 항목이 없는 경우 → **별도 로직 필요** (예: 상점 이름 기반 재분류)
- 전자영수증 전용 모델 검토:
 - **아이템 없는 영수증**을 잘 처리하는 커스텀 모델 or 분류 기준 필요



<Train_image_04>

- 특징:
 - 데이터 생성기로 만든 가짜 영수증
 - OCR 품질 높고, 글자 선명 (디지털 생성)

✓ Train_image_04.jpg 분석 결과

항목	결과	비고 (문제점)
DocType	receipt.retailMeal	적절 (식품 포함 가게 영수증 → Meal OK)
CountryRegion	KOR (99.5%)	정확
Items	3개 항목 인식	정상
MerchantName	re-MART (97.8%)	정확
ReceiptType	Supplies (97.3%)	Supplies도 무난하지만 Meal이 더 적절할 수도 있음
TaxDetails	정상 (금액, 설명, 네트 금액 정확)	OK
Total	₩22,000	정확
TotalTax	₩2,200	정확
TransactionDate	2024-09-12 (99.5%)	정확 (미래 날짜지만 → 가짜 데이터 특성)
TransactionTime	18:59:00	정확

🔍 문제점 요약:

- 별다른 문제 없음
- ReceiptType (Supplies):
 - Meal 로 분류해도 괜찮을 것 같은 경우 → 세부 조정 필요성 확인
-

현스시
신용카드전표-고객용

[영수증] 20160705-01-0003

가맹점명: 현스시
 사용자명: 김현수
 사업자번호: 217-17-24044
 전화번호: 055-267-2827
 주소: 경상남도 창원시 의창구 신월로 2

*** 신용승인정보 [1] ***

[카드종류] NH기업카드
 [카드번호] 5531-71**-*****-1534
 [유효기간] ** / **
 [결제방법] 일시불
 [전표매입사] NH카드
 [판매금액] 303,638
 [부가세] 30,362
 [승인금액] 334,000
 [승인번호] 30000609
 [승인일시] 2016-07-05 12:15:07
 [가맹점번호] 130349186

판매금액: 303,638
 부가세: 30,362
 총합계: 334,000

<Train_image_05>

- 특징:
 - 고액 결제 영수증 (30만원대)
 - 세금 관련 필드 중복

✅ Train_image_05.jpg 분석 결과

항목	결과	비고 (문제점)
----	----	----------

DocType	receipt	무난 (가게 이름이 '스시' → Meal로도 가능할 듯)
CountryRegion	KOR (99.5%)	정확
MerchantAddress	경상남도 창원시 의창구 신월로 2	디테일 문제: StreetAddress OK, House OK, City/State OK Road/CityDistrict 세분화가 부족
MerchantName	현.스시 (97.6%)	정확
MerchantPhoneNumber	+82552672827	정확
ReceiptType	Other (98.6%)	매매 (Meal/Supplies가 더 적절할 수도)
TaxDetails (2개)	중복 발생: - 첫 번째: OK (부가세) - 두 번째: '가세'로 오인	OCR 문제로 중복 추출: 부가세 → 가세 인식 오류
Total	₩334,000	정확
Subtotal	₩303,638	정확
TotalTax	₩30,362	정확 (다만 TaxDetails 중복 있음)
TransactionDate	2016-07-05 (99.3%)	날짜 오래됨 (문서 자체 문제 가능성)
TransactionTime	12:15:07	정확

문제점 요약:

1. 주소 디테일 문제 (주소 필드 세분화 오류)

- CityDistrict → 빠짐
- Road → 잘렸거나 누락
- 한국 주소 체계와 서양 모델 구조 차이

2. TaxDetails 중복 문제

- 첫 번째: 정상 (부가세)
- 두 번째: OCR 오류로 "가세" 인식 → 중복 항목 발생
→ 후처리로 중복 제거 필요

3. ReceiptType (Other)

- Meal/Supplies가 더 적절할 듯 (스시집 → Meal 가능성)



<Train_image_06>

- 특징:
 - 카카오택시 전자 영수증
 - 형식이 일반 영수증과 다름
 - 가게 이름, 품목 없음

Train_image_06.jpg 분석 결과

항목	결과	비고 (문제점)
DocType	receipt.retailMeal	❌ 오류 (→ 택시 영수증인데 Meal로 잘못 분류)
CountryRegion	KOR (99.5%)	정확
MerchantName	우 (49.6%)	❌ 잘못된 인식 (택시 영수증 특성상 정보 없음)
ReceiptType	Meal (99%)	❌ 완전 오류 (→ Transportation.Taxi 등 필요)

Total	₩4,600 (57.1%)	정확하지만 Confidence 낮음
TransactionDate	2020-01-23 (99.5%)	정확
TransactionTime	17:52:00 (99.5%)	정확

문제점 요약:

1. DocType, ReceiptType 완전 오류

- 택시 영수증인데 Meal 로 인식 → Prebuilt 모델의 한계
- 택시용 영수증 subtype 없음 (한국형 영수증 포맷 대응 부족)

2. MerchantName 인식 실패

- 실제 이름 없음 → 의미 없는 글자 (우) 인식
- 전자영수증 특성상 필드 누락이 많음

3. Total 인식은 OK

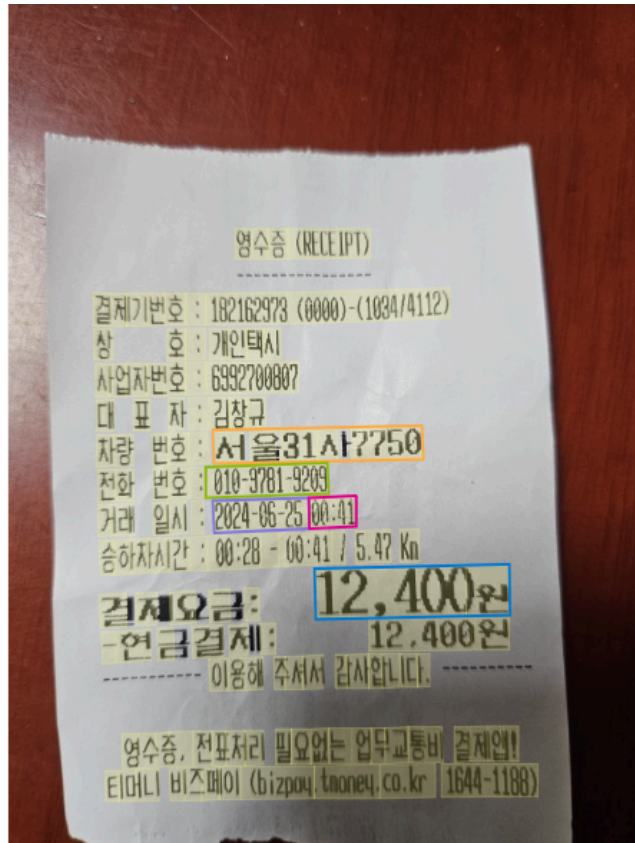
- Confidence는 낮지만 값은 정확

향후 개선 방향:

- SubType 문제 해결:
 - Azure Prebuilt Receipt 모델이 택시/교통 유형에 대해 부족
 - → Custom 모델로 택시 영수증 전용 모델을 학습시키는 게 필요할 수도
- MerchantName 비워두는 처리 필요:
 - 전자영수증에서 특정 필드 비어있을 경우 후처리 로직 필요
- ReceiptType 후처리 로직 추가:
 - 예:
 - 품목 없음 + 일정 가격 범위 + 특정 키워드 (택시 등) → Transportation.Taxi 로 변환

요약:

- 택시 영수증 → Prebuilt 모델 대응 부족
- 커스텀 모델 또는 후처리 로직 필요



<Train_image_07>

- 특징:
 - 일반 종이 영수증 (택시)
 - 인쇄 품질 준수

✓ Train_image_07.jpg 분석 결과

항목	결과	비고 (문제점)
DocType	receipt	OK
CountryRegion	KOR (79.1%)	Confidence가 낮음
MerchantAddress	서울31사7750	✗ 오류: 택시 번호판을 주소로 인식
- HouseNumber	7750	
- Road	서울31사	
MerchantPhoneNumber	+821097819209	정확
ReceiptType	Transportation.Taxi (69.5%)	OK (Confidence는 낮음)
Total	₩12,400 (99.5%)	정확
TransactionDate	2024-06-25 (99.5%)	정확
TransactionTime	00:41:00 (99.5%)	정확

🔍 문제점 요약:

1. MerchantAddress → 택시번호 인식 오류

- 서울31사7750 → 주소로 인식
- 택시번호판을 주소 필드에 넣어버림
- 후처리 필수: `Transportation.Taxi` 타입에서는 주소 필드 제거 or 무시

2. CountryRegion Confidence 낮음 (79.1%)

- 값은 맞지만 확신이 낮음 (한글 OCR 영향?)

3. ReceiptType은 적절하나 Confidence 낮음 (69.5%)

- `Transportation.Taxi`로 잘 인식 → 그러나 확신이 낮아 보완 필요

🚀 향후 대응방안:

• ReceiptType이 Taxi일 때 Address 제거 로직 추가

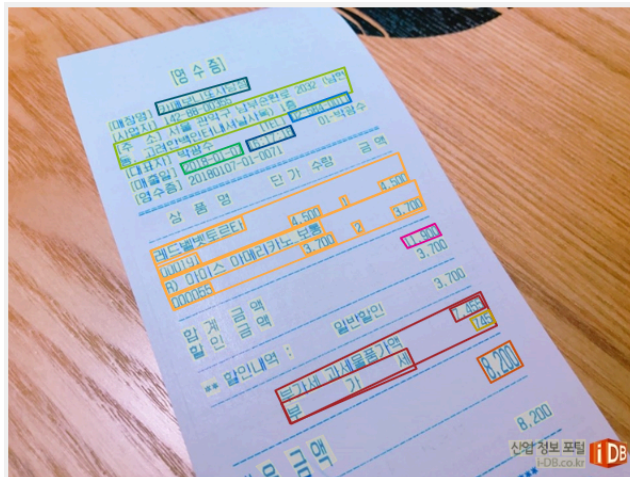
→ 택시 번호판을 주소로 착각하는 현상 방지

• CountryRegion 확신 낮은 경우 후처리 보완

- 예: 금액 단위 (`₩`, `KRW`)가 원화일 경우 자동 KOR로 보정

• ReceiptType Confidence Threshold 강화

- → 택시 영수증 여부 추가 검증 (예: 특정 키워드 "택시", "운행", 번호판 등)



<Train_image_08>

1. DocType: `receipt.retailMeal` 로 분류됐어. 편의점 영수증이니까 적절.

2. CountryRegion: `KOR` → 정확.

3. MerchantAddress:

- 전체 주소: 서울 관악구 남부순환로 2032 (남현 동, 고려한백인터내셔널서빙사목) 1층
- 세부 필드들은 이번에도 정확하게 나왔어.

4. MerchantName: 카페보니또사당점 → 정확.

5. MerchantPhoneNumber: +8225840011 → 정확.

6. ReceiptType: Meal → 정확.

7. Subtotal: KRW 11900

- 오류: 원래 합계는 8,200 원이야. 할인 적용 전 금액인 듯.

8. TaxDetails:

- NetAmount: KRW 7455
- Amount: KRW 745
- Description: 가 세 부 → 공백 이슈는 있지만, 전반적으로 잘 나왔어.

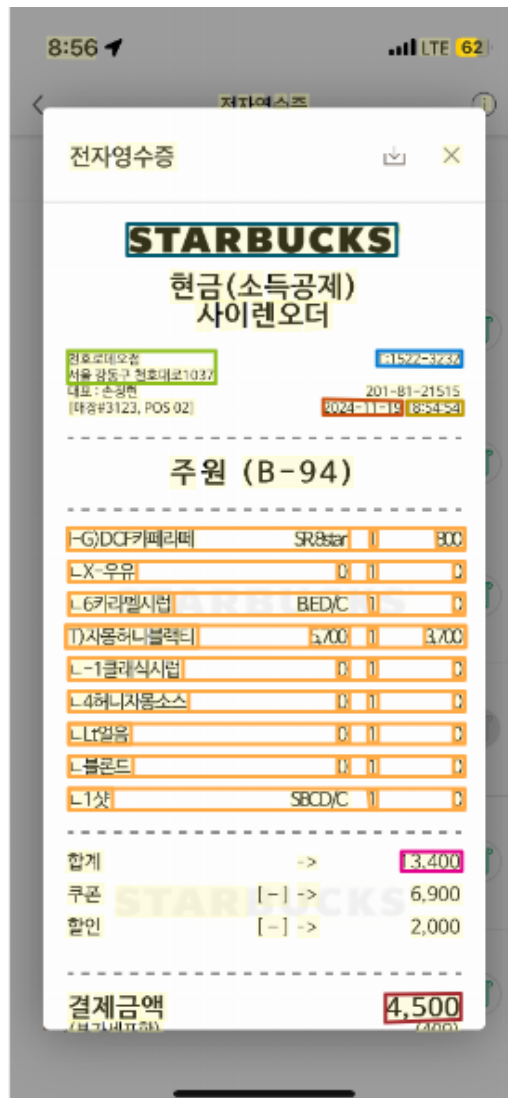
9. Total: KRW 8200 → 정확.

10. TotalTax: KRW 745 → 정확.

11. TransactionDate/Time: 2018-01-07 , 16:17:16 → 정확.

요약

- 주소랑 상호명: 완벽하게 잘 나왔고,
- 금액 필드 중에 Subtotal만 할인 전 금액으로 오류가 있음 (여전히 합계는 정확).



<Train_image_09>

9번 영수증 (스타벅스 전자영수증)

- **DocType:** receipt.retailMeal → 적절함 (음식점, 카페류)
- **CountryRegion:** KOR → 정확
- **Items:** 9개 → 잘 찾음
- **MerchantAddress:**
 - 주소: 천호대로오점 서울 강동구 천호대로 1037
 - 문제점:
 - **Road**: 천호대로오점 서 → 점 이름까지 끌려왔음 (주소와 가게명 혼합된 느낌)
 - **City**: "울" → 서울을 잘못 인식
 - 전체적으로 주소 디테일 박살 (패턴 비슷)

- MerchantName: STARBUCKS → 정확
- MerchantPhoneNumber: +8215223232 → 정확
- ReceiptType: Meal → 적절
- Subtotal: KRW 13400 → 정확
- Total: KRW 4500 →
 - 문제: 할인된 금액(결제금액)으로 Total 인식함 →
 - 할인, 쿠폰 항목 고려 안함
- TransactionDate: 2024-11-19 → 정확
- TransactionTime: 18:54:54 → 정확

핵심 문제:

- 주소 분리 실패 (가게명과 도로명 혼합)
- 할인 항목 무시하고 결제금액만 Total로 인식
(→ Total 계산 로직이 한국 영수증과 안 맞음)



<Train_image_10>

Azure Prebuilt Receipt 모델 테스트 결과 보고서

1. 테스트 환경

- 모델: Azure Document Intelligence **Prebuilt Receipt (Neural)**
- 설정:
 - 전처리 없이 원본 이미지 사용
 - 모델 자체적인 **angle 보정** 포함 판단
- 테스트 데이터:
 - 총 10장
 - 실물 영수증, 생성된 영수증, 전자영수증, 택시 영수증 혼합

2. 테스트 흐름

1. DocType 분류 결과 확인

2. 필드별 인식 정확도 확인

- 주요 필드: CountryRegion, MerchantName, MerchantAddress, TaxDetails, Items, Total, TotalTax 등

3. 문제점 및 오류 패턴 도출

3. 주요 문제점 및 오류 패턴

항목	문제점 / 오류 패턴
DocType (영수증 유형)	- 택시 영수증 → Meal , Supplies 등으로 잘못 분류- 생성된 영수증 포맷 → RetailMeal 등으로 애매하게 분류
MerchantName	- 특수문자/로고형 문구 인식 오류 (&MART , 현_스시 등)- E마트 로고 → 인식 실패 사례
MerchantAddress	- 대체로 틀은 맞으나- City , CityDistrict , Road , Suburb 구분 불명확 (예: 서울 → CityDistrict , 강동구 → Suburb 등)
TaxDetails	- 금액, 설명 인식 정확도는 높으나- Description 누락 (부가세 → 가세 , 가세부 등)- 금액 텍스트 혼합 인식 (7,455 가세 745)
Items (상품명)	- 대부분 정확하게 인식- 아이템 수가 많아도 잘 처리함

4. 잘 인식된 항목

항목	상태
CountryRegion	거의 완벽 (KOR)
TransactionDate, Time	정확하게 인식
Total, TotalTax, Subtotal	금액 항목 대체로 정확

5. 보완 포인트

1) 전처리 강화 필요

- 주소 필드 구분 개선
 - 예: 서울특별시, 강동구, 천호대로 를 명확히 구분하기 위해
 - 한글 주소 **Normalization** → 행정구역 DB 활용 추천
- 특수문자 보정
 - 로고형 문자, 특수문자 (E.MART , &MART) → 후처리로 교정 (정규표현식, 키워드 DB 활용)

2) 커스텀 모델 필요성 검토

- DocType 분류 정확도 개선
 - 택시 영수증, 전자 영수증 → **Classification 커스텀 모델** 적용 가능성 검토
- TaxDetails 디테일 보완
 - 커스텀 **Extraction** 모델을 활용하여 부가세 항목 정확도 개선 가능

3) 추가 실험 필요

- 전처리 적용 실험
 - 회전 보정, 노이즈 제거 등 전처리 후 모델 성능 변화 측정
- Google Document AI vs Azure DIS 성능 비교
 - 한국어 영수증 대상 두 모델 비교 실험 필요

6. 다음 단계 제안

1. 전처리 시나리오 설계 및 테스트

- 회전 보정, 노이즈 제거, 주소 정규화 등 적용 후 Prebuilt 모델 성능 재측정

2. 커스텀 모델 테스트

- Classification + Extraction 커스텀 모델 병행 검토

3. 경쟁 서비스 비교

- Google Document AI와 Azure DIS 한국어 영수증 인식 성능 비교 실험
-

6. 결론 및 실무 적용 판단

실제 현업 기준 판단:

- 만약 디지털 감사나 회계 감사에서 필요한 필드가:
 - **Total (총액)**
 - **Tax (세금)**
 - **MerchantName (상호명)**
 - **TransactionDate/Time (날짜/시간)**

정도의 핵심 필드만 필요하다면:

→ **Prebuilt** 모델만으로도 충분히 사용 가능 (전처리 최소 or 불필요).

- 반대로, 다음과 같은 세부적인 정보가 필요하다면:
 - **지역별 세금 분석** (주소 필드 세분화)
 - **영수증 유형별 통계** (DocType 분류 정확도 개선)
 - **특수 항목** (할인, 봉투 보증금 등)
- **커스텀 모델 학습 + 전처리**가 필요.

작성일: 2025-04-23