

# 电商订单截图VLM demo 标注指南 v2.0

## 1. 任务目标

- **A 结构化抽取：**从订单截图抽取并归一为5个核心字段，形成结构化记录
- **B 自动核验：**对金额字段做一致性校验（Reasoning R1：附加校验，不影响抽取任务统计）

## 2. 输入

电商订单详情页截图（平台包括：TB/JD/XHS）

截图可能包含：缩小、遮挡、字段缺失。

## 3. 输出字段

### 每个字段统一结构

- value：合法值/NULL
- field\_state：OK/MISSING/UNCLEAR
- evidence：截图原文片段（尽量包含字段名+值）

### value 全局规则

- OK → 必须填合法值
- MISSING/UNCLEAR → value = null

### field\_state全局规则

- OK：字段值完整、清晰、可识别，无歧义
- MISSING：字段值不存在
- UNCLEAR：字段值存在，但因部分缺失、遮挡、字段名不匹配等原因导致歧义、不唯一的

### 字段名及合法值

1. order\_outcome: 订单结果 (枚举: SUCCESS/CLOSED)
2. paid\_amount: 实付金额 (数字, 两位小数, 单位人民币)
3. original\_amount: 原始金额/商品总价/应付金额 (数字, 两位小数, 单位人民币)
4. discount\_amount: 优惠金额/总优惠 (数字, 两位小数, 单位人民币)
5. order\_time: 下单时间 (时间字符串, 保持原格式)

## 4. 字段定义、标注规则与示例

### 4.1 order\_outcome (订单结果)

**字段定义:** 订单最终业务结果, 仅二分类。

- SUCCESS: 订单正常完成/交易成功/已完成/已签收/已付款且未关闭 (以页面状态为准)
- CLOSED: 订单交易关闭/已取消/超时关闭/已关闭 (以页面状态为准)

#### 标注规则

1. 必须能在页面找到支撑判断的状态原文 (写在 evidence)
2. 如果页面存在“交易关闭/已取消/超时关闭”等明确字样 → 直接判 CLOSED
3. 如果页面存在“已完成/交易成功/已签收”等明确字样 → 直接判 SUCCESS
4. 如果状态区域部分遮挡, 或缩小到无法辨识, 且页面没有其他明确状态信息 → UNCLEAR
5. 如果状态区域被完全遮挡, 无字段名 → MISSING

#### 示例

- 示例 A
  - field\_state: OK
  - value: SUCCESS
  - evidence: 订单状态: 已完成
- 示例 B
  - field\_state: OK
  - value: CLOSED
  - evidence: 交易关闭
- 示例 C

- field\_state=UNCLEAR
- value: null
- evidence: 订单状态能看到字段名，但字段值看不清

## 4.2 paid\_amount (实付金额)

**字段定义：**用户实际应支付金额（不含货币符号，只保留数字）。

### 标注规则（优先级从高到低）

1. 优先抽取明确写有：实付/实付款/实付金额 的数值
2. 若无“实付”，但存在 合计/付款金额/应付金额 且语义明确为最终支付，可作为实付
3. 若出现多个金额（商品价/运费/优惠/合计）但无法唯一确定最终支付金额 → UNCLEAR
4. 页面没有任何最终支付相关字段 → MISSING

### value取值规则

- 输出纯数字，如 23.80
- 取两位小数

### 示例

- **示例 A (OK)**
  - field\_state = OK
  - evidence: 实付 ¥23.80
  - value: 23.80
- **示例 B (MISSING)**
  - field\_state=MISSING
  - value: null
  - evidence: “”

## 4.3 original\_amount (原始金额)

**字段定义：**用于与优惠金额形成“原始-优惠=实付”校验口径的未优惠前金额。常见来源：商品总价/原价/应付金额/总额。

### 标注规则：

1. 若页面明确出现 商品总价/总额/原价 → 取该值
2. 若页面出现 应付金额 且同页有明确 优惠金额 → 可取应付金额作为 original\_amount
3. 视觉锚点确认：当页面没有“商品总价/应付金额/原价”等字段名，但满足以下条件之一，可以判为 OK。
  - 条件 A：位置锚点
    - 金额紧贴在商品条目旁边（商品标题/规格附近）
    - 且该金额呈“原价常见样式”（灰色、小字、可能带划线、旁边还有更醒目的实付/优惠信息）
  - 条件 B：样式锚点
    - 金额为灰色小字 / 非主金额展示层级
    - 与实付金额的视觉层级明显不同（实付更大更显眼）
4. 无总价/应付/总额等字段，且视觉锚点不可确认 → MISSING
5. 若有疑似原始金额但看不清或多个候选无法唯一 → UNCLEAR

### 示例

- 示例 A (显示商品总价)
  - field\_state = OK
  - value: 39.90
  - evidence: 商品总价 ¥39.90
- 示例 B (显示应付金额)
  - field\_state = OK
  - value: 39.90
  - evidence: 应付金额 ¥39.90
- 示例 C (MISSING)
  - field\_state=MISSING

- value: null, field\_state=MISSING
- evidence: “”

## 4.4 discount\_amount (优惠金额)

**字段定义：**从 original\_amount 中减去的优惠总额（总优惠）。可来自：优惠/优惠合计/立减/券/满减等。

### 标注规则（优先级）

1. 若页面有 优惠合计/总优惠/优惠金额 → 直接取合计
2. 若优惠分项展示（如券+活动）且没有合计：
  - 分项都清晰可读 → 允许求和（并在 evidence 中写明“分项求和”）
  - 分项不全或不可读 → UNCLEAR
3. 页面完全没有优惠字段 → MISSING

### 示例

- 示例 A (显示合计优惠)
  - field\_state = OK
  - value: 6.10
  - evidence: 合计优惠 ¥6.10
- 示例 B (分项求和)
  - field\_state = OK
  - value: 5.00
  - evidence: 店铺券 -3.00 + 活动立减 -2.00 (分项求和)
- 示例 C (UNCLEAR)
  - field\_state=UNCLEAR
  - value: null
  - evidence: 优惠 (能看到字段名但金额看不清/被遮挡)

## 4.5 order\_time (下单时间)

**字段定义：**订单创建/下单对应的时间字段（不是支付/发货/收货时间）。

## 标注规则

1. 只抽取带有明确提示词的时间：下单时间/创建时间
2. 若页面只有“支付时间/付款时间/成交时间”而无下单时间 → MISSING
3. 若存在多个时间，但无法判断哪个是下单 → UNCLEAR
4. 输出保持页面原格式（如 2025-12-01 13:22）

## 示例

- 示例 A (OK)
  - field\_state = OK
  - value: 2025-12-01 13:22
  - evidence: 下单时间: 2025-12-01 13:22
- 示例 B (MISSING)
  - field\_state=MISSING
  - value: null
  - evidence: “”
- 示例 C (UNCLEAR)
  - field\_state=UNCLEAR
  - value: null
  - evidence: 能看到时间，但无对应的“下单时间”字段名

## 5. Reasoning (R1) : 金额一致性校验 (附加)

**触发条件：**仅当 paid\_amount/original\_amount/discount\_amount 三个字段的 field\_state 均为 OK。

**校验：**paid == original - discount

**输出**

- reasoning\_result: YES / NO / SKIP
- reasoning\_explain:
  - YES: 写出计算式
  - NO: 写出计算式 + 可能原因 (口径差异/运费/抽取错误)
  - SKIP: 写明哪个字段不是 OK

注意: Reasoning 不作为“抽取是否成功”的门槛; 它是用来发现抽取错误或口径歧义的。

## 6. 附: Prompt与输出 JSON 模板

### Prompt

1 你是一个严谨的信息抽取与校验器。请从这张电商订单详情页截图中抽取并校验字段。严格遵守以下规则:

2

3 【输出格式】

4 - 只输出一个合法JSON (不要代码块, 不要解释, 不要多余文字)。

5 - JSON必须包含且仅包含这些顶层键:

6 order\_outcome, paid\_amount, original\_amount, discount\_amount, order\_time, reasoning

7

8 【字段定义】

9 - order\_outcome: 订单结果, 只允许输出 "SUCCESS" 或 "CLOSED"

10 - paid\_amount: 实付金额 (数字, 保留两位小数)

11 - original\_amount: 原始金额/商品总价/应付金额 (数字, 保留两位小数)

12 - discount\_amount: 优惠金额/立减/共减 (数字, 保留两位小数)

13 - order\_time: 下单时间/创建时间 (格式 "YYYY-MM-DD HH:MM:SS"; 若页面只有到分钟则用 "YYYY-MM-DD HH:MM") ,

14

15 【状态state】

16 每个字段都必须输出:

17 - value: 若OK则给具体值; 若非OK则必须为null

18 - state: 只能是 "OK" / "MISSING" / "UNCLEAR"

19 - evidence: 从截图中摘取短可证明的原文片段 (必须包含数值以及数值附近的关键词作为语义锚点, 如“实付款/共减/合计/应付/商品”等; 若非OK则给导致缺失/不清晰的证据, 如“被遮挡/看不清/页面缺少该字段”)

20

21 判定规则:

22 - OK: 字段在截图中清晰存在且可确定唯一值

23 - MISSING: 截图中没有该字段 (页面确实未展示)

24 - UNCLEAR: 字段位置疑似存在但看不清/被遮挡/存在多个候选且无法确定

25

```

26  【order_outcome映射】
27  - 订单状态原文若表示交易成功/已完成/已付款等 → value="SUCCESS"
28  - 订单状态原文若表示交易关闭/已关闭/超时关闭等 → value="CLOSED"
29  - 若无法映射或看不清 → state="UNCLEAR", value=null
30
31  【金额规范化】
32  - 去掉货币符号与逗号，只输出数字字符串，例如 "1425.45"
33  - 若截图金额为整数，也输出两位小数，例如 "99.00"
34
35  【校验任务 reasoning】
36  - 如果 paid_amount、original_amount、discount_amount 三者 state 都为 OK：
37    校验是否满足 paid_amount = original_amount - discount_amount
38    若满足 reasoning.result="YES" 否则 "NO"
39    reasoning.explain 用一句话写出等式或指出不一致
40  - 否则：
41    reasoning.result="SKIP"
42    reasoning.explain 说明跳过原因（指出哪个字段非OK及其state）
43
44  【示例输出结构】（结构示意，实际值请基于图片）
45  {
46    "order_outcome": {"value": "SUCCESS", "state": "OK", "evidence": "交易成功"},
47    "paid_amount": {"value": "1425.45", "state": "OK", "evidence": "实付款 1425.45"}, 
48    "original_amount": {"value": "2999.00", "state": "OK", "evidence": "... 2999"}, 
49    "discount_amount": {"value": "1573.55", "state": "OK", "evidence": "共减 1573.55"}, 
50    "order_time": {"value": "2025-12-17 17:43:14", "state": "OK", "evidence": "下单时间 2025-12-17 17:43:14"}, 
51    "reasoning": {"result": "YES", "explain": "1425.45 = 2999.00 - 1573.55"} 
52  }

```

## 输出模版

```

1  {
2    "order_outcome": {"value": "SUCCESS|CLOSED|null", "field_state":
3      "OK|MISSING|UNCLEAR", "evidence": ""}, 
4    "paid_amount": {"value": 0.0, "field_state": "OK|MISSING|UNCLEAR",
5      "evidence": ""}, 
6    "original_amount": {"value": 0.0, "field_state": "OK|MISSING|UNCLEAR",
7      "evidence": ""}, 
8    "discount_amount": {"value": 0.0, "field_state": "OK|MISSING|UNCLEAR",
9      "evidence": ""}, 
10   "order_time": {"value": "string|null", "field_state": "OK|MISSING|UNCLEAR",
11     "evidence": ""}, 
12   "reasoning_result": "YES|NO|SKIP",
13   "reasoning_explain": ""
14 }

```

## 输出示例

```
1  {
2      "order_outcome": {"value": "SUCCESS", "state": "OK", "evidence": "交易成功"},
3      "paid_amount": {"value": "1425.45", "state": "OK", "evidence": "实付款 1425.45"},
4      "original_amount": {"value": "2999.00", "state": "OK", "evidence": "... 2999"},
5      "discount_amount": {"value": "1573.55", "state": "OK", "evidence": "共减 1573.55"},
6      "order_time": {"value": "2025-12-17 17:43:14", "state": "OK", "evidence": "下单时间 2025-12-17 17:43:14"},
7      "reasoning": {"result": "YES", "explain": "1425.45 = 2999.00 - 1573.55"}
8 }
```