

20251021_What解构业务PRD v1.4

1. 需求目标

针对给定的小红书多模态内容（图文、视频、音频），从消费者视角进行充分必要的What要素逆向解构，通过层级化递归深入分析，识别和提取内容中所有构成成分“what”，包含但不限于信息要素、视觉要素、音频要素、实体要素、关联模式等，并将这些要素结构化组织且直接关联原始多媒体素材。

核心价值: 通过对大量内容的“What”要素进行聚合分析，提取爆款内容的内容特征与新兴趋势。识别关键要素和价值点，通过专业化解构知识支撑，为创作者提供可参考的、经过验证的成功要素组合，提升内容创作的确定性。

2. 需求详细描述

2.1 系统输入定义

2.1.1 小红书多模态内容（必选）

原始多媒体文件要求:

- **图片内容:** 单图/图集（1-9张），保持原始分辨率，支持JPG/PNG/WEBP格式
- **视频内容:** 短视频文件（15秒-5分钟），保持原始质量，支持MP4/MOV格式
- **音频内容:** 从视频中提取音频轨道或独立音频文件，支持WAV/MP3格式
- **文本内容:** 标题、正文、话题标签，保持原始格式和编码
- **技术元数据:** 分辨率、格式、时长、帧率、文件大小等技术参数

多媒体素材管理要求:

- **文件存储:** 建立统一的素材库存储结构，生成唯一文件ID和访问路径
- **版本管理:** 支持原始文件、处理文件、衍生文件的版本管理
- **完整性保证:** 文件完整性校验、备份机制、访问权限管理
- **命名规范:** 采用统一命名规范，便于关联引用和批量管理

2.1.2 创作者信息（可选）

- **创作者基础信息:** 昵称、头像、认证信息、粉丝数、主要领域
- **创作者历史内容:** 该创作者历史发布的帖子集合（用于What要素对比分析）
- **内容特色标签:** 平台识别的创作者特征标签和内容风格定位

2.1.3 What解构历史基准（可选：账号一致性分析）

- **历史内容样本:** 同一创作者最近20-30个帖子的完整内容和解构结果
- **要素特征库:** 该创作者常见的视觉要素、内容要素、价值要素特征
- **风格基准:** 视觉风格、内容主题、表达方式的基准参考
- **用途说明:** 仅用于更准确地识别当前内容的独特What要素，不进行品牌分析

2.1.4 评论信息（必选，用于提取内容消费者关注的亮点）

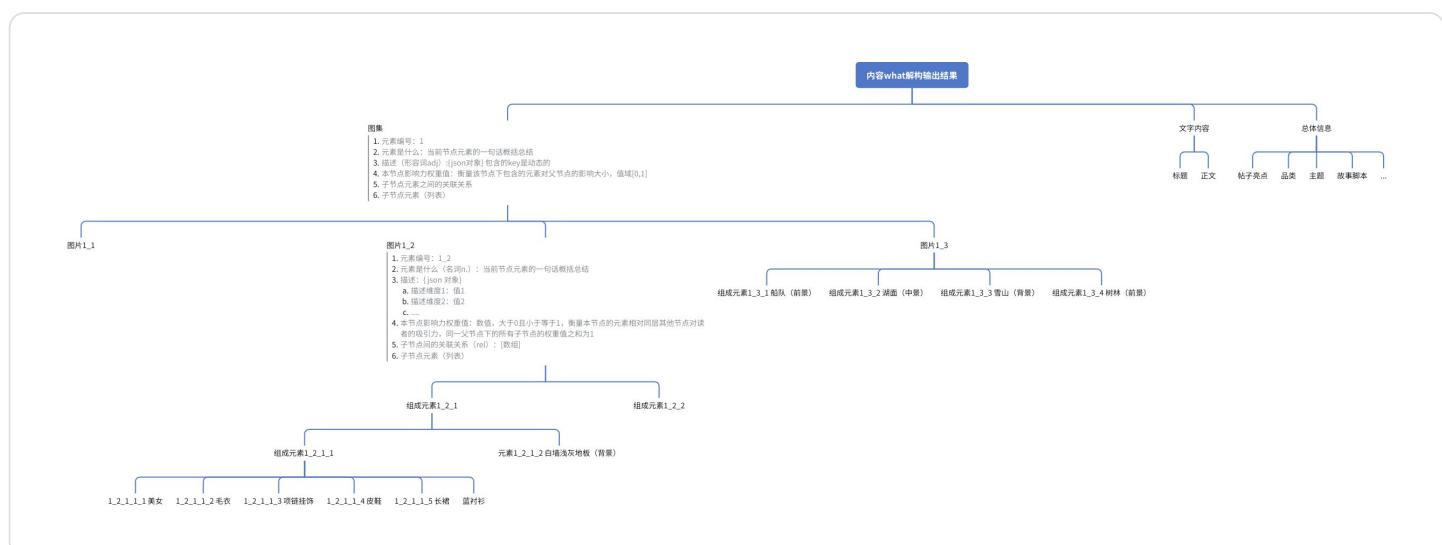
- 评论区的文字内容：用于分析内容消费者对帖子内容的核心关注点，即帖子封面、标题、图片、正文中的哪些元素是吸引他们阅读的亮点，以及能引起他们产生共鸣或其他各种情绪的关键点。

2.2 系统输出定义

2.2.1 输出含义定义

输出一个JSON对象，以层级递归地、自上而下地、从宏观到微观、从整体到局部的解构一篇笔记的全部“What”元素，解构形成一棵“What元素树”。输出结果为key-value组织的层级化要素集合，每个要素包含结构化的核心属性、关键特征、消费者亮点和素材关联，直接关联原始多媒体素材。

输出的what解构结果的树状图数据结构示例：



- 元素的树状结构：表示帖子中每个视觉/文本元素间的“包含”与“被包含”关系。比如，A视觉元素包含B视觉元素，则在树状结构中，A是B的上级父节点，B是A的子节点。举例：
 - 视觉元素：A是图片中一个人物，B是人物身上的一件衣服。
 - 文本元素：A是正文的全文，B是正文的一个段落。

2.2.2 输出格式定义

代码块

```

1  {
2      "帖子整体": {
3          "描述维度1": "值1",
4          "描述维度2": "值2",
5          ...
6      },
7      "帖子包含元素": [
8          {
9              "id": "1",
10             "what": "元素概述",
11             "描述": {
12                 "描述维度1": "值1",
13                 "描述维度2": "值2",
14                 ...
15             },
16             "子节点元素关系": ["关系描述1", "关系描述2", ...],
17             "元素重要性权重": 0.7,
18             "子节点元素": [
19                 {
20                     "id": "1_1",
21                     "what": "1_1元素概述",
22                     "描述": {
23                         "描述维度1": "值1",
24                         "描述维度2": "值2",
25                         ...
26                     },
27                     "子节点元素关系": ["关系描述1", "关系描述2", ...],
28                     "元素重要性权重": 0.2,
29                     "子节点元素": [...]
30                 },
31                 {
32                     "id": "1_2",
33                     "what": "1_2元素概述",
34                     "描述": {
35                         "描述维度1": "值1",
36                         "描述维度2": "值2",
37                         ...
38                     }
39                 }
40             ]
41         }
42     }
43 }
```

```
38     },
39     "子节点元素关系": ["关系描述1", "关系描述2", ...],
40     "元素重要性权重": 0.3,
41     "子节点元素": [...]
42 },
43 ...
44 ]
45 },
46 {
47     "id": "2",
48     "what": "2元素概述",
49     "描述": {
50         "描述维度1": "值1",
51         "描述维度2": "值2",
52         ...
53     },
54     "子节点元素关系": ["关系描述1", "关系描述2", ...],
55     "元素重要性权重": 0.7,
56     "子节点元素": [
57     {
58         "id": "2_1",
59         "what": "2_1元素概述"
60         "描述": {
61             "描述维度1": "值1",
62             "描述维度2": "值2",
63             ...
64         },
65         "子节点元素关系": ["关系描述1", "关系描述2", ...],
66         "元素重要性权重": 0.2,
67         "子节点元素": [...]
68     },
69     {
70         "id": "2_2",
71         "what": "2_2元素概述"
72         "描述": {
73             "描述维度1": "值1",
74             "描述维度2": "值2",
75             ...
76         },
77         "子节点元素关系": ["关系描述1", "关系描述2", ...],
78         "元素重要性权重": 0.3,
79         "子节点元素": [...]
80     },
81     ...
82 ]
83 },
84 ...
```

```
85      ]  
86  }
```

2.2.2.1 JSON对象结构存储

整个解构结果以JSON对象的结构来存储。图中每个视觉元素的树状上下层级关系，需要通过JSON对象中的嵌套结构来表示。比如，A视觉元素包含B视觉元素，则在树状结构中B是A的子节点，B元素应该存在于A元素的“子节点元素”字段中。

2.2.2.2 每个节点所需包含的字段及具体约束说明

- 树状图的顶层根节点元素（即帖子整体节点）
 - 帖子总体的各种描述、总结、归纳...等维度或角度，每一个维度都是一个字段项。包括但不限于：品类、主题、脚本...等维度。
 - 维度的名称和数量是跟随每一篇帖子动态变化的。
 - 一定要包含本帖子最关键/最吸引内容消费者的亮点维度，包括但不限于：内容亮点/情绪共鸣点/创作手法...等维度
 - 描述维度的数量不得少于3个，要尽可能多而全的覆盖这个帖子的所有可描述角度
 - 描述维度有哪些、有多少个，需要根据本帖子的品类、主题、关键词等信息来向知识库动态请求询问，根据知识库返回的结果来确定包含的维度和数量。
 - 向知识库请求的query句式：参见3.2环节中对query的定义
- 中间元素节点：每个中间节点都需包含下述6个字段
 - 节点编号：由数字与下划线 动态组合，数字大小表示元素先后顺序，下划线表示元素在树状结构的所在层级。比如根节点下第一层元素的编号为 1、2、3...，第二层元素的编号为1_1、2_1、3_2...，以此类推。
 - what：
 - 节点元素是文本（标题、正文、话题标签）时：what字段的值是节点文本的原文，不要对其做任何改造、总结、概括。
 - 节点元素是视觉元素（图片、图片的局部）时：what字段的值是一句针对该视觉元素的文本描述。要涵盖该视觉元素的核心元素和核心要点特征。
 - 描述：每个元素节点下的“描述”字段内容中所包含的“描述维度”都是动态变化的、开放的。
 - 描述维度一定要包含待解构元素最关键/最吸引人的亮点维度
 - 描述维度的数量不得少于3个，要尽可能多而全的覆盖这个元素的所有可描述角度

- 描述维度有哪些、有多少个，需要根据待解构元素是什么来向知识库动态请求询问，根据知识库返回的结果来确定包含的维度和数量。
 - 向知识库请求的query句式：刻画描述 “[元素的what字段值]” 核心特征的角度和维度有哪些？请尽可能不重不漏列举全。
- 节点权重值：大于0且小于等于1的数值。衡量该节点的内容对读者观众的吸引力大小。帖子中越是亮点、越是吸引观众关注的元素节点，权重值越大。
 - 约束条件：同一父节点下的各个子节点的权重值之和应固定为1，比如元素1_1下包含3个子节点1_1_1、1_1_2、1_1_3，不论这3个节点的权重值如何设定分配，3者之和应为1。
- 子节点元素关系：string数组，数组中每个元素表示一条该节点所属子节点间的关联关系的文字描述。
- 子节点元素：JSON数组。表示该视觉元素包含的子节点视觉元素，也就是树状结构的子节点元素。
 - 子节点元素的数量是动态不固定的。
- 底层叶子节点：上述“中间元素节点”所包含的6个字段中，除“子节点元素”、“子节点元素关系”2个字段外，其余字段均需存在于叶子节点中。

2.2.3 解构颗粒度定义

解构应递归深入，在一个节点是否要停止递归深入解构，需要动态判断。当满足以下任意一个条件时，停止递归深入：

- 当前元素已经是读者可感知的、不可再分的最小语义单元或最小视觉元素。
- 当前元素在整棵“元素树”中的层级已经达到10层，即解构已经深入10层。

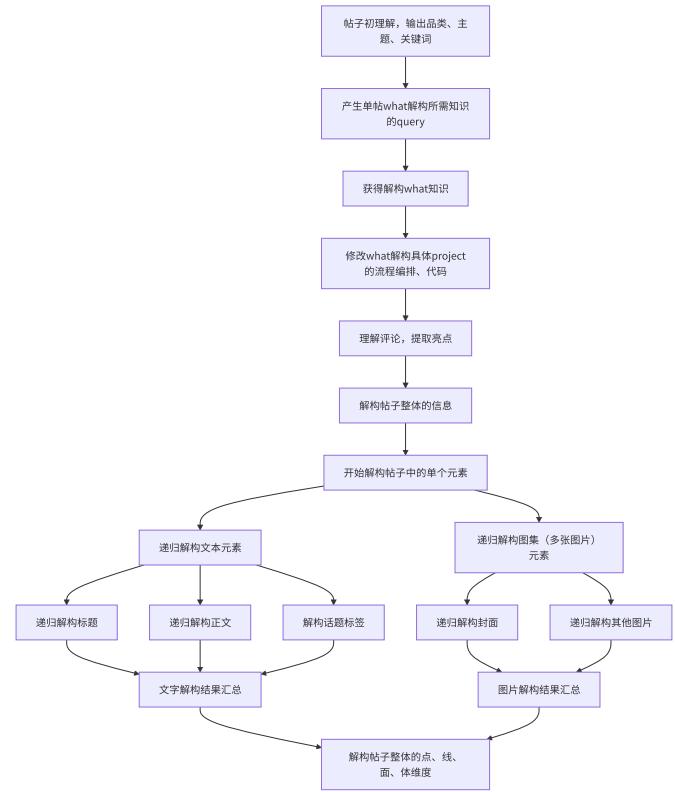
3. what解构的业务流程

3.1 总体解构流程

```

1 flowchart TD
2     A[帖子初理解, 输出品类、主题、关键词] --> B
3     B --> C[获得解构what知识]
4     C --> D[修改what解构具体project的流程编排、代码]
5     D --> E0[理解评论, 提取亮点]
6     E0 --> E1[解构帖子整体的信息]
7     E1 --> E[开始解构帖子中的单个元素]
8     E --> F[递归解构图集(多张图片)元素]
9     F --> F1[递归解构封面]
10    F --> F2[递归解构其他图片]
11    F1 --> F3[图片解构结果汇总]
12    F2 --> F3
13    E --> G[递归解构文本元素]
14    G --> G1[递归解构标题]
15    G --> G2[递归解构正文]
16    G --> G3[解构话题标签]
17    G1 --> G4[文字解构结果汇总]
18    G2 --> G4
19    G3 --> G4
20    G4-->H[解构帖子整体的点、线、面、体维度]
21    F3-->H
22

```



3.2 帖子整体解构环节

帖子的整体解构需要解构出的维度字段，需要通过知识库来获取。向知识库请求知识需要先产生query，具体query产生的方法和原则如下：

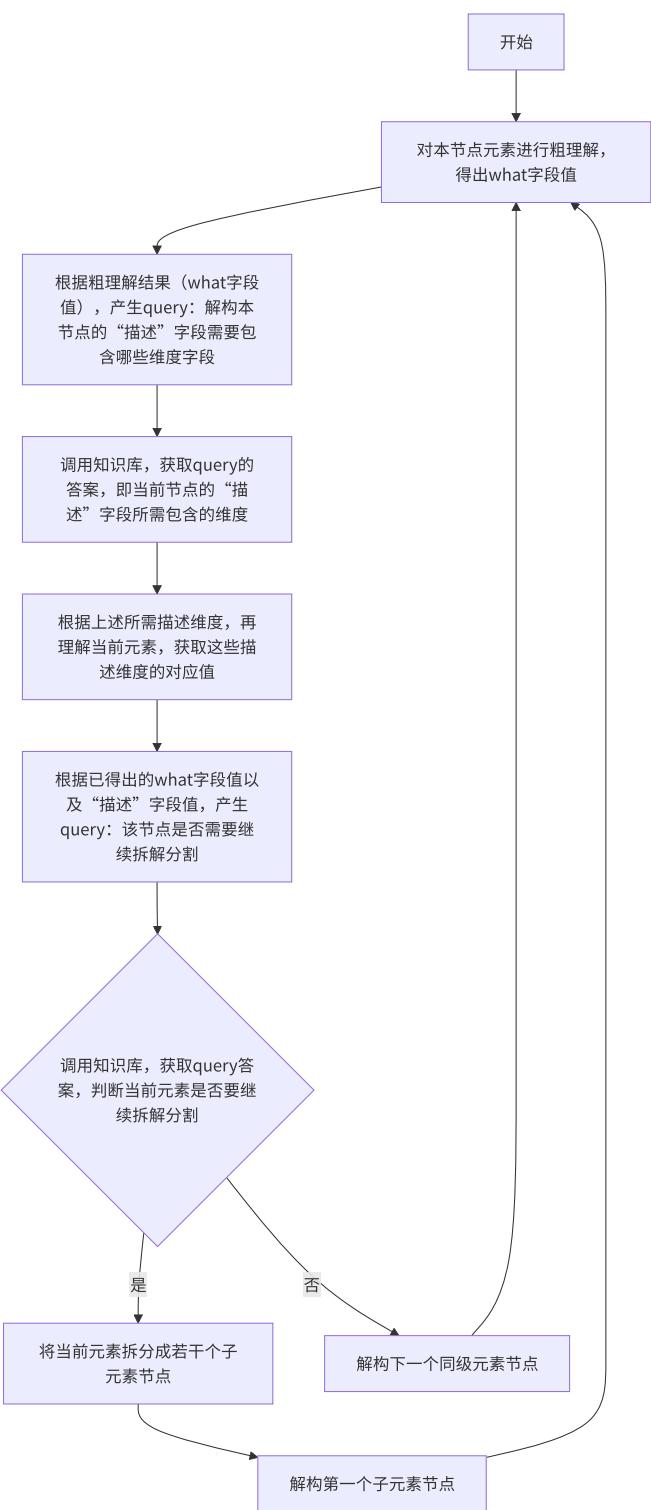
- Query句式：**对于一篇主题为“{帖子主题}”，品类为“{帖子品类}”，关键词包含“{帖子关键词列表}”的多模态社交媒体帖子，从内容创作者视角进行What要素的初步识别和分类，需要使用哪些通用工具？
- 动态填充示例：**对于一篇主题为“夏日穿搭”，品类为“时尚美妆”，关键词包含“OOTD, 小白裙”的多模态社交媒体帖子，从内容创作者视角进行What要素的初步识别和分类，需要使用哪些通用工具？

3.3 对单一节点（封面\其他图片\标题\正文）的递归解构的详细流程

```

1 flowchart
2     A[开始] -->B0[对本节点元素进行粗理解, 得出what字段值]
3     B0--> B1[根据粗理解结果 (what字段值) , 调用知识库, 获取query的答案, 即当前节点的“描述”字段值]
4     B1--> B[根据上述所需描述维度, 再理解当前元素, 获取这些描述维度的对应值]
5     B --> C[根据已得出的what字段值以及“描述”字段值, 产生query: 该节点是否需要继续拆解分割]
6     C --> D0[根据已得出的what字段值以及“描述”字段值, 产生query: 解构本节点的“描述”字段需要包含哪些维度字段]
7     D0 -->D{调用知识库, 获取query答案, 判断当前元素是否需要继续拆解分割}
8     D --是--> E1[将当前元素拆分成若干个子元素节点]
9     E1 --> E2[解构第一个子元素节点]
10    D --否--> E[解构下一个同级元素节点]
11    E-->B0
12    E2-->B0
13
14
15
16

```

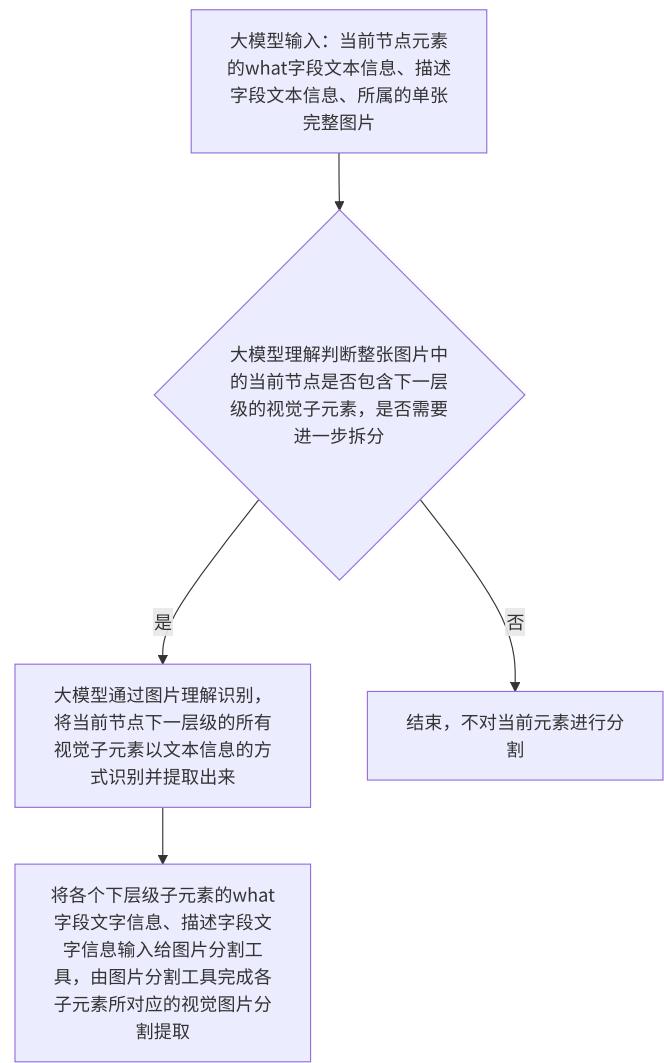


3.3.1 判断图片元素是否要继续分割拆解的具体细化流程

```

1 flowchart
2 A[大模型输入：当前节点元素的what字段文本信息
3 A-->C{大模型理解判断整张图片中的当前节点是否
4 C--是-->D1[大模型通过图片理解识别，将当前节
5 D1-->D2[将各个下层级子元素的what字段文字信息
6 C--否-->E[结束，不对当前元素进行分割]
7
8
9

```



需遵循原则：

- 每个视觉元素节点是否拆解，判断所需的输入信息不是经过图片分割后仅包含当前节点视觉元素的图片，而是当前节点所属的未分割的完整图片，以及当前节点的文字描述信息。也就是说，大模型在判断元素是否需要拆解时，不需要依赖图片分割的结果。

3.3.2 对单一节点解构如何获取知识

- 本环节旨在提出一些问题，这些问题的答案可以帮助对一个“图片/文字”元素进行what解构
- 解构过程会产生多个问题query，所有问题query分为2类：1) 如何进行what解构的方法，也称“内容知识”；2) 解构过程需要用到什么工具，也称“工具知识”。这两类query有各自不同的内容。
- 所有query的完整语句都由“固定的句式” + “动态填充的内容”构成，动态填充的内容是在what解构过程中产生的信息，在固定句式中通过大括号“{}”中间的内容表示
- 获取工具知识的query句式：

a. 针对“**图片整体**”的**分割拆解工具Query**:

- **Query句式**: 对一张描述为“{图片整体内容的简要描述}”的图片，从内容创作者视角进行视觉元素分割，需要使用哪些图片分割/抠图工具？
- **动态填充示例**: 对一张描述为“海边穿小白裙的女生”的图片，从内容创作者视角进行视觉元素分割，需要使用哪些图片分割/抠图工具？

b. 针对“**图片整体**”的**关键点/亮点提取工具Query**:

- **Query句式**: 从内容创作者视角，对一张描述为“{图片整体内容的简要描述}”的图片，进行亮点或关键点的提取，需要使用什么图片分析工具或大模型视觉理解工具？
- **动态填充示例**: 从内容创作者视角，对一张描述为“海边穿小白裙的女生”的图片，进行亮点或关键点的提取，需要使用什么图片分析工具或大模型视觉理解工具？

c. 针对“**图片中的特定视觉元素(XXX)**”的**分割拆解工具Query**:

- **Query句式**: 对一张“{XXX图片整体描述}”的图片中的“{YYY具体视觉元素描述}”的视觉元素，从内容创作者视角进行细致分割拆解，需要使用哪些高精度图片分割/抠图工具或细粒度图像理解大模型？
- **动态填充示例**: 对一张“海边穿小白裙的女生”的图片中的“画面中的小白裙”的视觉元素，从内容创作者视角进行细致分割拆解，需要使用哪些高精度图片分割/抠图工具或细粒度图像理解大模型？

d. 针对“**图片中的特定视觉元素(YYY)**”的**关键点/亮点提取工具Query**:

- **Query句式**: 从内容创作者视角，对一张“{XXX图片整体描述}”的图片中的“{YYY具体视觉元素描述}”的视觉元素，进行亮点或关键点的提取，需要使用什么图像特征提取工具或专业视觉大模型？
- **动态填充示例**: 从内容创作者视角，对一张“海边穿小白裙的女生”的图片中的“女生的笑容”的视觉元素，进行亮点或关键点的提取，需要使用什么图像特征提取工具或专业视觉大模型？

e. 针对“**文本段落整体(XXX)**”的**分割拆解工具Query**:

- **Query句式**: 对一段描述为“{文本段落内容摘要}”的文本段落，从内容创作者视角进行结构化分割拆解，需要使用什么文本切分工具或文本结构化大模型？
- **动态填充示例**: 对一段描述为“小白裙穿搭技巧与产品推荐”的文本段落，从内容创作者视角进行结构化分割拆解，需要使用什么文本切分工具或文本结构化大模型？

f. 针对“**文本段落整体(XXX)**”的**关键点/亮点提取工具Query**:

- **Query句式**: 从内容创作者视角，对一段描述为“{文本段落内容摘要}”的文本段落（包含标题、正文、话题标签），进行亮点或关键点的提取，需要使用什么情感分析工具、语义理解工具或文本摘要大模型？
- **动态填充示例**:

- 从内容创作者视角，对一段描述为“#夏日穿搭 #小白裙”的话题标签集合，进行亮点或关键点的提取，需要使用什么关键词分析工具或社交媒体热点分析大模型？
- 从内容创作者视角，对一段描述为“懒人必备！三明治机搞定一周早餐不重样”的标题，进行亮点或关键点的提取，需要使用什么标题优化分析工具或文本营销效果预测大模型？

g. 以上句式里的{XXX}是动态内容，XXX需要替换为当前待解构节点的what字段值

- 若当前节点是图片，则XXX是这张图片的一句话概括描述
- 若当前节点是文本，则XXX是这段文本的原文

3.4 图片元素/文本元素的递归解构需要遵循的准则

- 元素解构次序遵循“由宏观到微观、由大到小、由整体到局部”的原则。
- 分解解构出的元素必须做到“不重复不缺漏”、“充分必要”的原则，即一个父节点下解构出的所有子节点元素合并后就是完整的父节点元素，且子节点元素都必须在父节点元素中存在（子节点文本必须存在于父节点文本中，子节点视觉元素必须能在父节点视觉元素中可看到）
- 元素解构拆分的维度需按照通过外部知识库获取到解构拆分维度进行拆解。若没有外部知识，可以根据大模型自身知识来判断。

3.5 what解构可能需要的组件/模块

名称	功能定义
外部知识调用	根据产生的query，调用外部知识库获取解构所需的知识
图片理解	根据图片识别其中的元素、主题、风格等等信息，转化为文字语义信息
图片分割/抠图工具	实现将一张图片分割为多个不同的完整的视觉元素，并能将分割出的多个元素另存为独立的图片。
文本切分工具	将一段文本根据知识库提供的知识分割为多段文本/词组/单词等。

4. what解构的全局通用约束准则

- 客观至上: 你的所有解构必须严格基于用户提供的帖子内容。禁止主观臆测、凭空假设、虚构数据。
-