## 2. Detailed comparison with TPTU-v2

We provide a detailed comparison between OctoTools and TPTU-v2 (Kong et al., 2023) in Table 2.

| Features | OctoTools | TPTU-v2 | Justification for TPTU-v2 |
|---|---|---|---|
| **Training-Free** | Yes | No | Requires additional fine-tuning for task planning and API calling, rather than being training-free. |
| **Dynamic Planning** | Yes | No | Relies on static plans with limited iterative updates, whereas OctoTools adapts on-the-fly. |
| **Self-Refinement** | Yes | No | Lacks a mechanism to correct or refine reasoning mid-execution. |
| **Toolset Optimization** | Yes | Yes | Employs an "API Retriever" to select relevant APIs, similar to our validation-based selection. |
| **Extensible Tools** | Yes | Limited | Focuses on a fixed set of APIs rather than a standardized plug-and-play interface. |
| **Runnable System** | Yes | No | Not released as a publicly accessible framework, though validated in real-world settings. |
| **Comprehensive Evaluations** | Yes, 16 tasks | No, 1 task only | Evaluated on a single task domain, whereas OctoTools spans 16 diverse benchmarks. |
| **In-Depth Study** | Yes | Limited | Offers less extensive analysis and ablations compared to OctoTools' multi-faceted evaluations. |

*Table 2.* Detailed comparison between OctoTools and TPTU-v2.