

TOOLMAKER

LLM Agents Making Agent Tools

Georg Wölflein^{1,2,3,†}Dyke Ferber^{3,4,‡}Daniel Truhn⁵Ognjen Arandjelović²Jakob Nikolas Kather^{1,4,6}¹EKFZ for Digital Health, TU Dresden²University of St Andrews³Synagen AI⁴NCT, Heidelberg University Hospital⁵University Hospital Aachen⁶University Hospital Dresden[†]work done while at EKFZ for Digital Health, TU Dresden and University of St Andrews[‡]work done while at EKFZ for Digital Health, TU Dresden and NCT Heidelberg

Motivation

- LLM agents have predefined sets of tools
- Implementing new tools requires manual work & technical expertise
-  Employ agents to autonomously create new tools from research papers with code repositories

Components

Workflow state entails conversation history and environment state

$$s = (\mathcal{H}, \mathcal{E}) \in \mathcal{H} \times \mathcal{E}$$

There are three types of workflow components (each act on state):

$$\underbrace{\mathcal{H} \times \mathcal{E}}_{\text{old state}} \mapsto \underbrace{\mathcal{H} \times \mathcal{E}}_{\text{new state}} \times \mathcal{R}_{\text{return value}}$$

- LLM calls append to conversation history $\mathcal{H} \mapsto \mathcal{H} \times \mathcal{M}$
- Environment interactions mutate environment state and return observation $\mathcal{E} \mapsto \mathcal{E} \times \mathcal{O}$ (\mathcal{O} is the set of observations)
- Agents do both: $\mathcal{H} \times \mathcal{E} \mapsto \mathcal{H} \times \mathcal{E} \times \mathcal{R}$

Problem

Given paper + GitHub repository + task description, generate an LLM-compatible tool



Task

Description: Train a model for biomarker classification using STAMP.

(optional full-text article)

Arguments:

- slide_dir (str): Path to the folder containing the whole slide images.
Example: "/mount/input/TCGA_BRCA"
- clini_table (str): Path to CSV file containing the clinical data.
Example: "/mount/input/clini.xlsx"
- slide_table (str): Path to CSV file containing the slide metadata.
Example: "/mount/input/slides.csv"
- target_column (str): Name of the column in clini_table that contains the target labels.
Example: "pathologic_stage"

Returns:

- trained_model (str): Path to the trained model



Environment definition

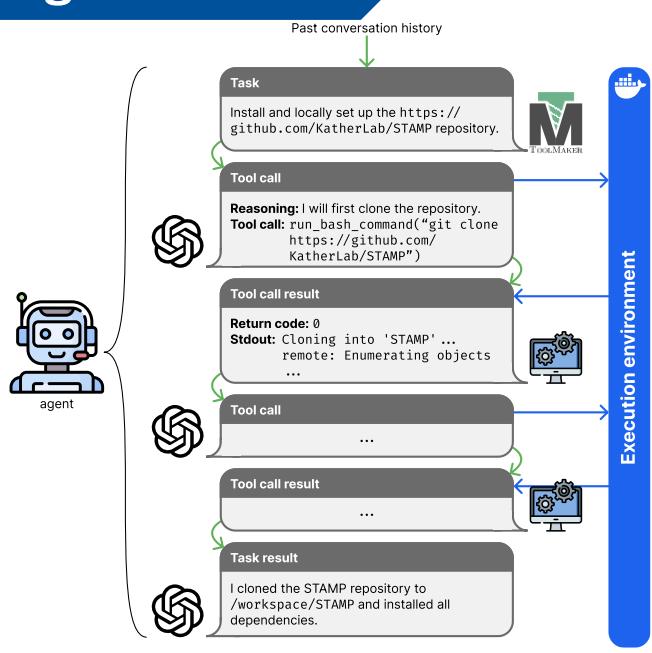
FROM python:3.12

RUN git clone https://github.com/KatherLab/STAMP && cd STAMP && apt update && apt install -y openslide-tools && pip install -e . && stamp init && stamp setup

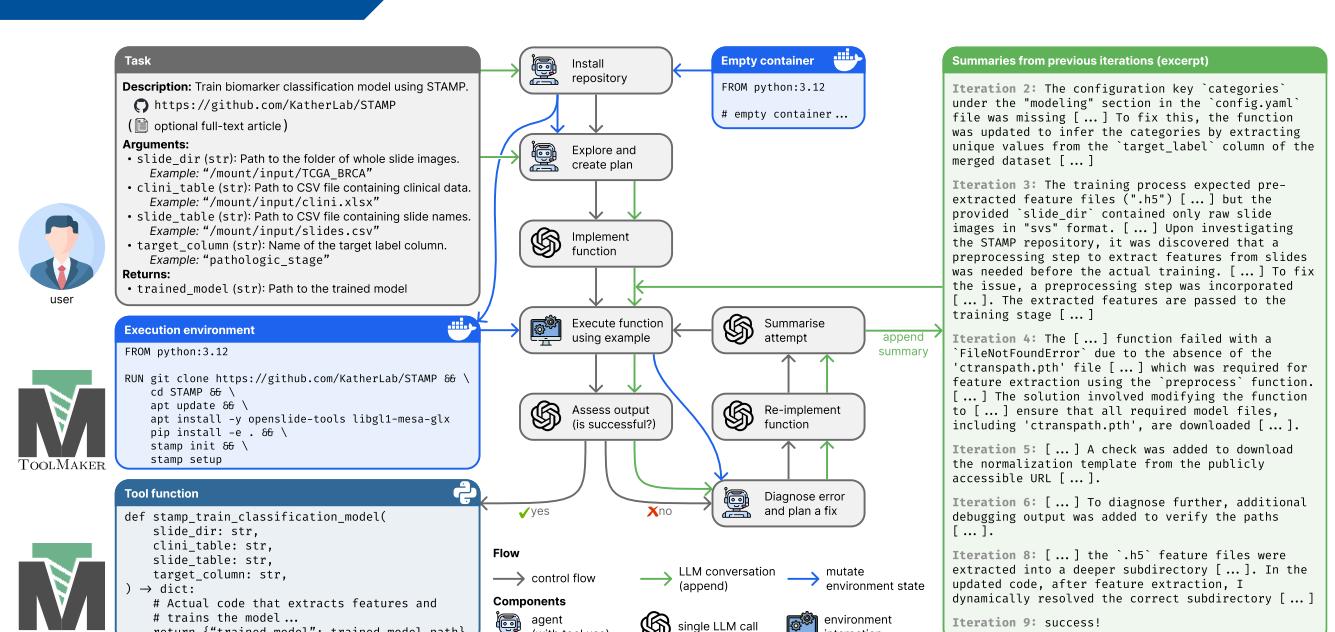
Tool function

```
def stamp_train_classification_model(
    slide_dir: str,
    clini_table: str,
    slide_table: str,
    target_column: str,
) -> dict:
    # Actual code that performs feature extraction and model training...
    return {"trained_model": trained_model_path}
```

Agents



Workflow



Main results

Task	TOOLMAKER (ours)				OpenHands (Wang et al., 2024)						
	Invoc.	Tests	Cost	Actions	Tokens	Invoc.	Tests	Cost	Actions	Tokens	
Pathology	conch_extract_features (Lu et al., 2024b)	3/3	9/9	\$0.35	15 (1 _○)	171,226	3/3	9/9	\$0.08	5	51,701
	musk_extract_features (Xiang et al., 2025)	3/3	6/6	\$1.19	29 (6 _○)	696,386	0/2	X	X	7	97,386
	pathfinder_verify_biomarker (Liang et al., 2023)	0/2	4/6	\$0.61	27 (1 _○)	356,825	0/2	4/6	\$0.08	6	49,414
	stamp_extract_features (El Nahhas et al., 2024)	3/3	12/12	\$1.12	20 (4 _○)	631,138	0/3	3/12	\$0.07	6	42,793
	stamp_train_classification_model (El Nahhas et al., 2024)	3/3	9/9	\$2.27	33 (9 _○)	1,249,521	0/3	0/9	\$0.15	8	87,915
	uni_extract_features (Chen et al., 2024)	3/3	9/9	\$0.61	16 (4 _○)	326,806	X	X	\$0.25	10	177,119
Radiology	medsam_inference (Ma et al., 2024)	3/3	6/6	\$0.96	18 (6 _○)	508,954	X	X	\$0.07	5	41,096
	nnunet_train_model (Isensee et al., 2020)	0/2	0/4	\$2.90	35 (9 _○)	1,792,291	0/2	0/4	\$0.12	8	79,231
Omics	cytopus_db (Kunes et al., 2023)	3/3	12/12	\$0.41	10 (3 _○)	185,912	X	X	\$0.36	8	236,217
	esm_fold_predict (Verkuil et al., 2022; Hie et al., 2022)	2/3	13/15	\$0.66	20 (1 _○)	336,754	X	X	\$0.11	6	69,493
Other	flowmap_overfit_scene (Smith et al., 2024)	2/2	6/6	\$0.70	18 (5 _○)	358,552	X	X	\$0.36	15	250,787
	medsss_generate (Jiang et al., 2025)	3/3	6/6	\$0.53	25 (3 _○)	282,771	3/3	6/6	\$0.15	10	104,505
	modernbert_predict_masked (Warner et al., 2024)	3/3	9/9	\$0.66	20 (4 _○)	356,228	X	X	\$0.13	10	82,930
	retfound_feature_vector (Zhou et al., 2023)	3/3	6/6	\$0.97	31 (5 _○)	561,936	0/3	0/6	\$0.08	4	46,521
	tabPFN_predict (Hollmann et al., 2025)	3/3	9/9	\$0.23	10 (1 _○)	95,257	3/3	9/9	\$0.07	4	36,320

Conclusion

Autonomous tool creation is feasible for complex scientific tasks

