Analysis

Refer to the Conceptual Map I made on excalidraw to understand the summary better.

Paper 1: REACT: SYNERGIZING REASONING AND ACTING IN LANGUAGE MODELS

It tells us about ReAct(Reasoning + Action) , Agent receives an observation O(t) at time set t and performs an action following a policy with context.

Augment the agent's action space to Â= A u L, where L is the space of language. An action in the language space, which we will refer to as a thought or a reasoning trace, does not affect the external environment, thus leading to no observation feedback.

The above said "thought" aims to compose useful info by reasoning over current context and updating the context to support future reasoning or Acting.

Paper 2: Toolformer: Language Models Can Teach Themselves to Use Tools

Their main objective was to equip an Language Model with the ability to use different tools via help of API Calls.The tools involved were Question-Answering,Wiki Search,Calculator, Calendar and Machine Translation System.

Their methodology consists of the following steps:

1. Sampling API Calls
2. Executing API Calls
3. Filtering API Calls
4. Fine-tuning the model on the Filtered API Calls.

What they basically did was when they were executing user's query they would send out an API call according to the query and complete the result.

Eg: Joe Biden Was Born in[QA(Where was Joe Biden born)]Scranton,[QA(In Which state in Scranton)] Pennsylvania.

Paper 3: REST MEETS REACT: SELF-IMPROVEMENT FOR MULTI-STEP REASONING LLM AGENT

What they did is implement a Search Agent which **decides** if a query needs additional information and if the answer is yes it uses the search tool and then goes back into the decision step.
After Answer generation it does 2 self-checks(Relevance Self-Check and Grounding Self-Check)

The Paper also discusses about Iterative Self Improvement(ReSt) and Auto Eval by LLM's.

Paper 4: Chain of Tools: Large Language Model is an Automatic Multi-tool Learner

The Most Interesting paper out of the 5 from my pov.

So what they are doing basically is that . They set up a document protocol which contains meta-info about the tools i.e argument requirements, tool description and output schema. Then toolset along with its document protocol is fed into LLM that then generates a python file ,chaining the tools for usage according to their task.

Attributal Reflection: When the generated Python program runs into an error then the error trace and faulty code snippet is fed into LLM to identify the tool causing the error then the result is reevaluated.

BlackBox Probing: Involve instruct the LLM to formulate a question q targeting the functionality of a tool t and generate a program utilizing the tool t to solve the formulated question , the tool use instance is then documented in the protocol structure and can be later used by the LLM for query related to that tool, This makes the LLM a multi-tool learner.

Paper 5: Chain of Tools: Large Language Model is an Automatic Multi-tool Learner

LATS consists of a series of operations – selection, expansion, evaluation, simulation, backpropagation, and reflection – performed in succession until the task is successfully completed or a computational limit is reached after sampling k trajectories.

Creates a tree of trajectory by sampling actions and uses Monte Carlo Tree search to steer the search algorithm towards most promising regions of the tree.

Decision making is refined using back propagation and self reflection.