

Updated Workflow and Plan for the New Competition

Introduction

Based on your feedback, we've updated our plan to include setting up a baseline Kaggle submission in Phase 1. We'll ensure that we align with the competition requirements from the outset. Additionally, we've reviewed the prior plan to incorporate any important elements that were missing, while retaining the structure of the previous plan.

Our goal remains to leverage state-of-the-art (SOTA) methods using the Agent Zero framework to develop a competitive solution for the AI Mathematical Olympiad Progress Prize 2, while enhancing our learning and collaboration.

Updated Workflow

Phase 1: Initial Understanding and Baseline Setup (Weeks 1-2)

Objective: Deepen our understanding of the Agent Zero framework, set up our development environment, and establish a baseline Kaggle submission that aligns with competition requirements.

Tasks:

1. Understand the Agent Zero Framework:
 - Watch Tutorials and Presentations:
 - Agent Zero Framework Introduction
 - Agent Zero YouTube Channel
 - Agent Zero with GPU Acceleration
 - Explore the GitHub Repository:
 - Clone the repository: `git clone https://github.com/frdel/agent-zero.git`
 - Review the documentation and codebase to understand its capabilities and features.
2. Set Up Development Environment:
 - Install necessary dependencies and ensure compatibility with your system.
 - Run sample agents to familiarize yourself with the framework.
3. Set Up Baseline Kaggle Submission:
 - Prepare the Data Pipeline:
 - Obtain and clean the competition dataset as per Kaggle's guidelines.
 - Ensure data processing aligns with competition restrictions (e.g., data privacy, allowed libraries).
 - Integrate Data with Agent Zero:
 - Modify Agent Zero to accept the competition dataset.
 - Ensure that the agent can perform training and testing on the dataset.
 - Generate Initial Output:
 - Run the agent to produce a basic solution output in the required submission format.
 - Validate that the output meets all competition submission requirements.
4. Submit Baseline Solution:
 - Make an initial submission to Kaggle to establish a benchmark score.
 - Record feedback and any error messages for future improvements.
4. Identify Unique Features of Agent Zero:
 - Determine what sets Agent Zero apart from other frameworks.
 - Explore its ability to integrate various tools and handle tasks requiring GPU acceleration.

5. Create a Freedcamp Roadmap:
 - Set up a project management board to track tasks, milestones, and deadlines.

Phase 2: Literature Review and Strategy Planning (Weeks 3-4)

Objective: Review previous top solutions and current SOTA methods to inform our strategy.

Tasks:

1. Review Previous Top Solutions:
 - Analyze Winning Approaches:
 - Numina Team's Solution
 - Other top teams' discussions on Kaggle:
 - Team Fireworks
 - Third Place Solution
 - Extract Key Techniques:
 - Identify methodologies, models, and strategies that led to their success.
 - Note any challenges they faced and how they overcame them.
2. Study SOTA Techniques in LLMs for Math:
 - Google DeepMind's Work:
 - AlphaGeometry
 - DeepMind's AI Solves IMO Problems
 - OpenAI's o1 Model:
 - Improving Mathematical Reasoning with Process Supervision
 - Other Relevant Papers:
 - ReasonEval: Evaluating Mathematical Reasoning Beyond Accuracy
3. Explore Advanced Techniques:
 - Monte Carlo Tree Search (MCTS):
 - Research how MCTS can be implemented within Agent Zero.
 - Teaching Language Models to Reason with Search
 - Memory Layers for Agent Zero:
 - Investigate tools for long-term memory integration:
 - MEMO
 - GetZep
 - AutoKaggle Framework:
 - AutoKaggle: LLM-Powered Multi-Agent Framework
 - Explore how multi-agent collaboration can enhance problem-solving.
4. Find and Review Relevant Articles:
 - Use resources like TopBots to find trending techniques.
 - Utilize tools like Notebook LM to consume articles efficiently.

Phase 3: Data Collection and Analysis (Weeks 5-6)

Objective: Assess existing datasets and identify gaps to enhance our model's training.

Tasks:

1. Evaluate Current Data:
 - Review the data collected from previous efforts.
 - Identify areas where data is lacking, such as problems involving images or specific mathematical domains.
2. Explore the OMNI-MATH Dataset:
 - OMNI-MATH Dataset on Hugging Face

- Understand the dataset's structure, domains, and difficulty levels.
 - Assess how it can be leveraged for our project.
3. Consider Data Synthesis Techniques:
 - Read the Data Synthesis Survey for methods to generate synthetic data.
 - Use models like OpenAI's o1 to create reasoning datasets.
 - Explore CoT-Based Reward Modeling to enhance data quality.
 4. Plan for Data Integration:
 - Determine how to incorporate new data into our training pipeline.
 - Ensure data quality and relevance to competition problems.
 - Address any data privacy or licensing issues.

Phase 4: Implementation of Techniques (Weeks 7-12)

Objective: Implement and experiment with selected techniques within the Agent Zero framework.

Tasks:

1. Implement MCTS in Agent Zero:
 - Develop agents that utilize MCTS for enhanced problem-solving capabilities.
 - Ensure compatibility with Agent Zero's architecture.
 - Test the integration thoroughly to ensure stability.
2. Incorporate Memory Layers:
 - Implement long-term memory functionalities using tools like MEMO or GetZep.
 - Allow the agent to learn from past interactions and improve over time.
 - Evaluate the impact on problem-solving efficiency and accuracy.
3. Integrate SOTA Techniques:
 - Chain-of-Thought (CoT) Prompting:
 - Experiment with CoT prompting to enhance reasoning steps.
 - Implement Self-Consistency methods to improve reliability.
 - Tool Integration:
 - Equip the agent with external tools for symbolic computations (e.g., SymPy).
 - Ensure secure and efficient integration.
 - Reinforcement Learning (RL):
 - Explore RL methods to train the agent for theorem proving.
 - Reference: Deep Reinforcement Learning for Symbolic Mathematics
4. Develop a Reasoning Dataset:
 - Create a dataset using OpenAI's o1 model for reasoning examples.
 - Consider distilling this dataset into Agent Zero for improved performance.
 - Implement Curriculum Learning to organize training data by difficulty.

Phase 5: Testing, Evaluation, and Refinement (Weeks 13-20)

Objective: Continuously test, evaluate, and refine our agent to improve performance in the competition.

Tasks:

1. Conduct Extensive Testing:
 - Run the agent on a wide range of problems, including those from the competition dataset.
 - Utilize cross-validation and hold-out sets to assess generalization.
2. Analyze Results:
 - Use evaluation metrics to identify strengths and weaknesses.

- Perform ablation studies to understand the impact of each technique.
 - Pay special attention to problem types where the agent underperforms.
3. Iterate on Techniques:
 - Adjust parameters and try different combinations of techniques.
 - Optimize hyperparameters using tools like Optuna or Hyperopt.
 - Re-implement or refine methods as necessary based on performance data.
 4. Prepare for Submissions:
 - Ensure that the agent meets all competition requirements.
 - Test submission procedures to avoid technical issues.
 - Prepare documentation and reports as required by the competition.

Ongoing Tasks

Collaboration Framework and Mentor Engagement

- Adopt an Agile Framework:
 - Use Scrum with Kanban elements for flexibility and visualization.
 - Hold regular sprint planning, stand-ups, and retrospectives.
 - Assign roles and responsibilities based on team members' strengths.
- Engage with Mentors Regularly:
 - Schedule periodic meetings to review progress and seek guidance.
 - Share updates and challenges openly for valuable feedback.
 - Incorporate mentors' suggestions into the project plan.

Project Management

- Use Freedcamp for Task Management:
 - Keep track of tasks, deadlines, and responsibilities.
 - Update the roadmap as the project evolves.
 - Utilize Gantt charts or calendars for visual timelines.

Documentation

- Maintain Thorough Records:
 - Document all experiments, code changes, and results.
 - Keep meeting notes and decisions for future reference.
 - Ensure version control for code and documents using GitHub.

Communication

- Regular Team Meetings:
 - Discuss progress, roadblocks, and next steps.
 - Use tools like Slack or Microsoft Teams for ongoing communication.
 - Foster an environment of open and transparent communication.

Additional Resources

- Agent Zero Resources:
 - Agent Zero GitHub Repository
 - Agent Zero YouTube Channel
- Previous Competition Solutions:
 - Numina Team's Winning Approach
 - Kaggle Discussions on AIM0 1 Solutions
- Relevant Research Papers and Resources:

- Neuro-Symbolic Methods:
- Neural Theorem Provers
- Reinforcement Learning for Theorem Proving:
- Deep Reinforcement Learning for Symbolic Mathematics
- Curriculum Learning with Synthetic Data:
- Measuring Mathematical Problem Solving with the MATH Dataset
- AlphaCode and AlphaProof:
- AlphaCode
- AlphaProof
- COT-Based Reward Modeling Discussion:
- Kaggle Discussion on Reward Modeling

Final Thoughts

By incorporating the baseline Kaggle setup in Phase 1 and ensuring that we haven't missed any important elements from the prior plan, we have a comprehensive strategy that leverages both proven methodologies and innovative techniques. Our structured approach allows us to systematically explore, implement, and refine various strategies within the Agent Zero framework.

Consistent collaboration, thorough documentation, and effective use of our resources will be key over the next 5 months. Let's make the most of this opportunity to learn, innovate, and potentially achieve a strong result in the competition.

Let's get started and make progress together!