JSPM’s
Rajarshi Shahu College of Engineering, Tathawade, Pune
(An Empower Autonomous Institute under Savitribai Phule Pune University)

Department of Computer Science and Business Systems
Artificial Intelligence Mini Project
AY-2025-26

Title of Project: AI-Based Stock Trading Optimizer using Heuristic Search Algorithms

Student Name:
1. Sushrut Deshpande - RBT23CB002

Branch: T.Y CSBS

Sub: Professional Elective - Artificial Intelligence

Faculty Name: Prof. Kavita Patil

Introduction

This project aims to design an intelligent trading system that automatically identifies the most profitable stock trading parameters using heuristic search algorithms. It leverages AI techniques such as Hill Climbing, Local Beam Search, and A* Search to optimize a moving average crossover strategy for stock trading. The system downloads live market data, simulates trades, and evaluates profitability to find optimal configurations.

Objective of the Project:

- To automate the process of finding optimal short-term and long-term moving average windows for trading strategies.
- To compare the efficiency and accuracy of different heuristic algorithms for financial optimization.
- To develop a backtesting framework that evaluates trading strategies on real stock data.

Key Features:

- Downloads historical stock data from Yahoo Finance using yfinance.
- Implements multiple heuristic algorithms: Hill Climbing, Local Beam Search, and A* Search.
- Automatically finds optimal moving average parameters to maximize trading profit.
- Provides detailed performance comparison across algorithms.
- Modular design for scalability and future extensions.

Mapping with SDG goals : SDG Goal 8: Decent Work and Economic Growth

Scope:

- Can be extended to include real-time trading integration.
- Useful for financial analysts, investors, and algorithmic traders.
- Adaptable for multiple stock tickers and time periods.
- Can serve as a foundation for AI-based portfolio management systems.

Algorithm Used

Chosen Algorithms:

1. Hill Climbing Search
2. Local Beam Search
3. A Search

Reason For Selection: These algorithms are well-suited for parameter optimization problems with large search spaces and unknown objective functions, like financial data modeling.

Advantages:

- Efficient parameter tuning.
- Can escape local minima using advanced search techniques.
- Provide interpretable results with profit-based evaluation.
- Easy to implement and extend for other domains.

Module

1. Data Model Acquisition:
Retrieves historical stock data (e.g., Apple – AAPL) from Yahoo Finance using the yfinance library.
The user specifies a date range, and the data (Open, High, Low, Close, Volume) is stored in a clean Pandas DataFrame for further analysis.

2. Backtesting Module:
Implements the moving average crossover strategy.
Calculates profit by simulating trades when the short-term moving average crosses above or below the long-term one.
Uses an initial capital of $10,000 and outputs the final profit for given parameters.

3. Hill Climbing Search Module:
Starts with random parameter values and iteratively adjusts them to improve profit.
Stops when no better neighboring state is found — representing a local optimum.

4. Local Beam Search:
Runs multiple search paths (beams) in parallel.
Selects and continues only the best few states at each step to explore a broader parameter space and avoid local maxima.

5. A* Search Modul:
Uses a heuristic-driven approach combining path cost and estimated future gain.
Prioritizes the most promising parameter combinations to find the global best profit efficiently.

6. Results & Visualization Module:
Collects outcomes from all algorithms, compares final profits, and presents the best-performing configuration. Helps users analyze the performance of Hill Climbing, Beam Search, and A* methods clearly.

Methodology & Results

Problem Analysis:

- The goal was to maximize the profit of a trading strategy based on historical stock data.
- The parameters (short and long moving average windows) heavily influence strategy performance.
- This is a classic search and optimization problem suitable for heuristic AI methods.

Algorithm Design & Coding:

- The algorithms (Hill Climbing, Local Beam Search, A*) were implemented in Python.
- Each algorithm explores the parameter space differently to find optimal trading windows.
- Common helper functions were developed for data acquisition, portfolio simulation, and evaluation.

Execution & Testing:

- Dataset: Apple Inc. (AAPL)
- Duration: 2020–2023
- Initial Capital: $10,000
- Evaluation Metric: Final portfolio value (profit).
- The algorithms were tested sequentially, and their outputs were logged and compared.

Validation:

- Results were verified by cross-checking profits manually for a few parameter pairs.
- Consistency between different runs confirmed the correctness of the implementation.

Methodology & Results

Algorithm	Optimal Parameters (Short, Long)	Profit ($)	Observation
Hill Climbing	(31, 164)	8,365.55	Reached local maximum quickly but limited exploration.
Local Beam Search	(6, 112)	16,357.79	Improved over Hill Climbing due to multi-state search.
A* Search	(9, 10)	**116,864.41**	Found global optimum using heuristic guidance.

Conclusions:

- Among the three algorithms, A Search* demonstrated the best results in terms of profit and convergence.
- The heuristic-based approach helped explore a wider parameter space effectively.
- The system successfully shows how AI search algorithms can be used in real-world financial applications like stock trading optimization.

References

References:

1. Yahoo Finance API Documentation.
2. Kaggle – Financial Market Datasets and Trading Simulations
3. Python Libraries: yfinance, pandas, numpy, heapq, itertools