The Boyer-Moore algorithm is an efficient string search algorithm that can outperform other algorithms, especially in scenarios where the pattern is significantly shorter than the text. It achieves this efficiency by using preprocessing to create heuristic tables (the bad character table and the good suffix table) that guide the search process.

Why Boyer-Moore Can Outperform Other Algorithms

1. **Efficient pattern shifting:**
   - **Bad Character Heuristic:** Allows the pattern to skip over sections of the text where the mismatches occur, potentially skipping over large portions of the text.
   - **Good suffix heuristic:** Further optimizes the search by leveraging previously matched suffixes, allowing the pattern to be shifted efficiently.
2. **Reduces Comparisions:** By skipping parts of the text that have been determined to not match, the Boyer-Moore algorithm often performs fewer comparisons than algorithms like the naive approach or even the Rabin-Karp algorithm in practice.
3. **Preprocessing:** The preprocessing of the pattern (i.e., building the bad character table) allows for efficient matching during the search phase, making the algorithm particularly effective for longer patterns.