```
import { useState, useRef, useEffect } from "react";

const SYSTEM_PROMPT = `你是一位深度研读李常受（Witness Lee）信息的神学分析助手，专门识别和

你的分析框架如下：

【黄金路径的定义】
黄金路径是信息内在的神学骨架：从神的经纶出发，经过基督、灵、生命，抵达教会与新耶路撒冷的有机脉络。

【分析方法】
1．找「神圣动词」：成为、进入、居住、充满、浸透、调和、终极完成、经历、享受、得着、活出、表达
2．找「三一神的移动」：父的计划→子的成就→灵的应用，识别哪个位格是重心
3．找「客观-主观」转折点：客观事实如何转化为主观经历
4．找「终末归宿词」：新耶路撒冷、荣耀、终极完成、神人合一等
5．建立「路径句」：「神要藉着〔某途径〕，使〔某对象〕经历〔某实际〕，成为〔某目标〕」

【输出格式要求】
请严格按照以下JSON格式输出，不要输出任何其他内容，不要有markdown代码块标记：

{
  "title": "信息标题",
  "pathSentence": "一句话路径句（50字以内）",
  "trinitarianMovement": [
    { "person": "父", "role": "角色描述", "content": "具体内容", "emphasis": "high/m
    { "person": "子", "role": "角色描述", "content": "具体内容", "emphasis": "high/m
    { "person": "灵", "role": "角色描述", "content": "具体内容", "emphasis": "high/m
  ],
  "pivotPoints": [
    { "objective": "客观事实", "subjective": "主观经历" }
  ],
  "pivotMechanism": "转折机制说明",
  "pathNodes": [
    { "id": 1, "label": "节点名称", "subtitle": "人物/位格/关键词", "detail": "展开说明
  ],
  "verbChain": ["动词1", "动词2", "动词3"],
  "eschatologicalWords": [
    { "term": "终末词", "reference": "经文" }
  ],
  "uniqueness": "本篇路径的独特性分析（100字以内）",
  "theologicalTension": "最深的神学张力（80字以内）"
}`;

function GoldenPathAnalyzer() {
  const [input, setInput] = useState("");
```

```
const [result, setResult] = useState(null);
const [loading, setLoading] = useState(false);
const [error, setError] = useState(null);
const [activeTab, setActiveTab] = useState("path");
const [animateIn, setAnimateIn] = useState(false);
const textareaRef = useRef(null);

useEffect(() => {
  if (result) {
    setTimeout(() => setAnimateIn(true), 50);
  } else {
    setAnimateIn(false);
  }
}, [result]);

const analyze = async () => {
  if (!input.trim()) return;
  setLoading(true);
  setError(null);
  setResult(null);
  setAnimateIn(false);

  try {
    const response = await fetch("https://api.anthropic.com/v1/messages", {
      method: "POST",
      headers: { "Content-Type": "application/json" },
      body: JSON.stringify({
        model: "claude-sonnet-4-20250514",
        max_tokens: 1000,
        system: SYSTEM_PROMPT,
        messages: [{ role: "user", content: `请分析以下信息纲目的黄金路径：\n\n${inp
      })
    });

    const data = await response.json();
    const text = data.content?.map(i => i.text || "").join("") || "";
    const clean = text.replace(/```json|```/g, "").trim();
    const parsed = JSON.parse(clean);
    setResult(parsed);
  } catch (err) {
    setError("分析时发生错误，请检查输入内容后重试。");
  } finally {
    setLoading(false);
  }
};

const emphasisColor = (level) => {
```

```
  if (level === "high") return "#d4af37";
  if (level === "medium") return "#a0896e";
  return "#6b7280";
};

const tabs = [
  { id: "path", label: "路径图" },
  { id: "trinity", label: "三一神移动" },
  { id: "pivot", label: "转折节点" },
  { id: "verbs", label: "动词链" },
  { id: "insight", label: "神学洞见" }
];

return (
  <div style={{
    minHeight: "100vh",
    background: "linear-gradient(135deg, #0a0e1a 0%, #111827 50%, #0f1520 100%)
    fontFamily: "'Georgia', 'Times New Roman', serif",
    color: "#e8dcc8",
    padding: "0",
    position: "relative",
    overflow: "hidden"
  }}>
    {/* Background texture */}
    <div style={{
      position: "fixed", inset: 0, opacity: 0.03,
      backgroundImage: `repeating-linear-gradient(0deg, transparent, transparen
        repeating-linear-gradient(90deg, transparent, transparent 40px, #d4af37
      pointerEvents: "none"
    }} />

    {/* Header */}
    <div style={{
      borderBottom: "1px solid rgba(212,175,55,0.2)",
      padding: "28px 40px 24px",
      background: "rgba(0,0,0,0.3)",
      backdropFilter: "blur(8px)",
      position: "relative"
    }}>
      <div style={{ display: "flex", alignItems: "baseline", gap: "16px" }}>
        <span style={{ fontSize: "22px", color: "#d4af37", letterSpacing: "4px"
        <div>
          <h1 style={{
            margin: 0, fontSize: "26px", fontWeight: "normal",
            color: "#f0e6cc", letterSpacing: "2px",
            textShadow: "0 0 40px rgba(212,175,55,0.3)"
          }}>
```

```
          黄金路径分析器
        </h1>
        <p style={{ margin: "4px 0 0", fontSize: "12px", color: "#8a7a5a", le
          LEE WITNESS · GOLDEN PATH · THEOLOGICAL ANALYZER
        </p>
      </div>
    </div>
  </div>

  <div style={{ maxWidth: "900px", margin: "0 auto", padding: "32px 24px" }}>

    {/* Input Area */}
    <div style={{
      background: "rgba(212,175,55,0.04)",
      border: "1px solid rgba(212,175,55,0.15)",
      borderRadius: "8px",
      padding: "24px",
      marginBottom: "28px"
    }}>
      <label style={{
        display: "block", fontSize: "11px", letterSpacing: "3px",
        color: "#8a7a5a", marginBottom: "12px"
      }}>
        输入信息纲目
      </label>
      <textarea
        ref={textareaRef}
        value={input}
        onChange={e => setInput(e.target.value)}
        placeholder="将李常受信息纲目粘贴于此……"
        style={{
          width: "100%", minHeight: "180px", background: "transparent",
          border: "none", outline: "none", color: "#e8dcc8",
          fontFamily: "inherit", fontSize: "14px", lineHeight: "1.8",
          resize: "vertical", boxSizing: "border-box",
          "::placeholder": { color: "#4a4030" }
        }}
      />
      <div style={{ display: "flex", justifyContent: "flex-end", marginTop: "
        <button
          onClick={analyze}
          disabled={loading || !input.trim()}
          style={{
            background: loading ? "rgba(212,175,55,0.1)" : "linear-gradient(1
            border: "none", color: loading ? "#8a7a5a" : "#0a0e1a",
            padding: "10px 32px", borderRadius: "4px",
            fontSize: "12px", letterSpacing: "3px",
```

```
        cursor: loading ? "not-allowed" : "pointer",
        fontFamily: "inherit", fontWeight: "bold",
        transition: "all 0.3s ease",
        boxShadow: loading ? "none" : "0 4px 20px rgba(212,175,55,0.3)"
      }}
    >
      {loading ? "分析中……" : "提炼路径"}
    </button>
  </div>
</div>

{error && (
  <div style={{
    background: "rgba(180,60,60,0.1)", border: "1px solid rgba(180,60,60,
    borderRadius: "6px", padding: "16px", marginBottom: "24px",
    color: "#e88080", fontSize: "14px"
  }}>
    {error}
  </div>
)}

{loading && (
  <div style={{ textAlign: "center", padding: "60px 0" }}>
    <div style={{
      width: "40px", height: "40px", border: "2px solid rgba(212,175,55,0
      borderTop: "2px solid #d4af37", borderRadius: "50%",
      margin: "0 auto 20px", animation: "spin 1s linear infinite"
    }} />
    <p style={{ color: "#8a7a5a", fontSize: "13px", letterSpacing: "2px"
    <style>{`@keyframes spin { to { transform: rotate(360deg); } }`}</sty
  </div>
)}

{result && (
  <div style={{
    opacity: animateIn ? 1 : 0,
    transform: animateIn ? "translateY(0)" : "translateY(16px)",
    transition: "all 0.6s ease"
  }}>
    {/* Title & Path Sentence */}
    <div style={{
      background: "linear-gradient(135deg, rgba(212,175,55,0.08), rgba(21
      border: "1px solid rgba(212,175,55,0.25)",
      borderLeft: "3px solid #d4af37",
      borderRadius: "8px", padding: "24px 28px", marginBottom: "24px"
    }}>
      <div style={{ fontSize: "11px", letterSpacing: "3px", color: "#8a7a
```

```
    路径句
  </div>
  <h2 style={{ margin: "0 0 12px", fontSize: "17px", color: "#f0e6cc"
    {result.title}
  </h2>
  <p style={{ margin: 0, fontSize: "15px", color: "#d4af37", lineHeig
    「{result.pathSentence}」
  </p>
</div>

{/* Tabs */}
<div style={{ display: "flex", gap: "4px", marginBottom: "20px", flex
  {tabs.map(tab => (
    <button
      key={tab.id}
      onClick={() => setActiveTab(tab.id)}
      style={{
        background: activeTab === tab.id ? "rgba(212,175,55,0.15)" :
        border: `1px solid ${activeTab === tab.id ? "rgba(212,175,55,
        color: activeTab === tab.id ? "#d4af37" : "#6b7280",
        padding: "7px 18px", borderRadius: "4px",
        fontSize: "12px", letterSpacing: "1px",
        cursor: "pointer", fontFamily: "inherit",
        transition: "all 0.2s ease"
      }}
    >
      {tab.label}
    </button>
  ))}
</div>

{/* Tab Content */}
<div style={{
  background: "rgba(0,0,0,0.2)", border: "1px solid rgba(212,175,55,0
  borderRadius: "8px", padding: "28px", minHeight: "200px"
}}>

  {/* 路径图 */}
  {activeTab === "path" && (
    <div>
      <div style={{ fontSize: "11px", letterSpacing: "3px", color: "#
        路径节点
      </div>
      <div style={{ position: "relative" }}>
        {result.pathNodes?.map((node, i) => (
          <div key={i} style={{ display: "flex", gap: "20px", marginB
            {/* Connector */}
```

```
            <div style={{ display: "flex", flexDirection: "column", a
              <div style={{
                width: "12px", height: "12px", borderRadius: "50%",
                background: i === 0 ? "#d4af37" : i === result.pathNo
                border: "2px solid rgba(212,175,55,0.6)",
                boxShadow: (i === 0 || i === result.pathNodes.length
                flexShrink: 0, marginTop: "4px"
              }} />
              {i < result.pathNodes.length - 1 && (
                <div style={{
                  width: "1px", flex: 1, minHeight: "40px",
                  background: "linear-gradient(to bottom, rgba(212,17
                  margin: "4px 0"
                }} />
              )}
            </div>
            {/* Node Content */}
            <div style={{ paddingBottom: i < result.pathNodes.length
              <div style={{ display: "flex", alignItems: "baseline",
                <span style={{ color: "#f0e6cc", fontSize: "15px" }}>
                {node.subtitle && (
                  <span style={{
                    fontSize: "11px", color: "#8a7a5a",
                    background: "rgba(212,175,55,0.08)",
                    padding: "2px 8px", borderRadius: "3px", letterSp
                  }}>
                    {node.subtitle}
                  </span>
                )}
              </div>
              <p style={{ margin: "0 0 16px", fontSize: "13px", color
                {node.detail}
              </p>
            </div>
          </div>
        ))}
      </div>
    </div>
  )}

  {/* 三一神移动 */}
  {activeTab === "trinity" && (
    <div>
      <div style={{ fontSize: "11px", letterSpacing: "3px", color: "#
        三一神的移动轨迹
      </div>
      <div style={{ display: "grid", gap: "16px" }}>
```

```
        {result.trinitarianMovement?.map((item, i) => (
          <div key={i} style={{
            background: "rgba(212,175,55,0.04)",
            border: `1px solid ${emphasisColor(item.emphasis)}40`,
            borderLeft: `3px solid ${emphasisColor(item.emphasis)}`,
            borderRadius: "6px", padding: "18px 20px"
          }}>
            <div style={{ display: "flex", alignItems: "center", gap:
              <span style={{
                fontSize: "18px", color: emphasisColor(item.emphasis)
                fontWeight: "bold", minWidth: "20px"
              }}>
                {item.person === "父" ? "✦" : item.person === "子" ? "
              </span>
              <span style={{ color: emphasisColor(item.emphasis), fon
                {item.person}
              </span>
              <span style={{ fontSize: "12px", color: "#6b7280" }}>—
              {item.emphasis === "high" && (
                <span style={{
                  marginLeft: "auto", fontSize: "10px", letterSpacing
                  color: "#d4af37", background: "rgba(212,175,55,0.1)
                  padding: "2px 8px", borderRadius: "3px"
                }}>主导</span>
              )}
            </div>
            <p style={{ margin: 0, fontSize: "14px", color: "#c8b898"
              {item.content}
            </p>
          </div>
        ))}
      </div>
    </div>
  )}

  {/* 转折节点 */}
  {activeTab === "pivot" && (
    <div>
      <div style={{ fontSize: "11px", letterSpacing: "3px", color: "#
        客观 → 主观转折
      </div>
      <div style={{ display: "grid", gap: "12px", marginBottom: "24px
        {result.pivotPoints?.map((point, i) => (
          <div key={i} style={{
            display: "grid", gridTemplateColumns: "1fr auto 1fr", gap
            alignItems: "center"
          }}>
```

```
            <div style={{
              background: "rgba(60,100,160,0.1)", border: "1px solid
              borderRadius: "6px", padding: "12px 16px",
              fontSize: "13px", color: "#9aaac8", lineHeight: "1.6"
            }}>
              {point.objective}
            </div>
            <div style={{ color: "#d4af37", fontSize: "18px", textAli
            <div style={{
              background: "rgba(212,175,55,0.06)", border: "1px solid
              borderRadius: "6px", padding: "12px 16px",
              fontSize: "13px", color: "#c8b070", lineHeight: "1.6"
            }}>
              {point.subjective}
            </div>
          </div>
        ))}
      </div>
      {result.pivotMechanism && (
        <div style={{
          background: "rgba(212,175,55,0.06)", border: "1px solid rgb
          borderRadius: "6px", padding: "16px 20px"
        }}>
          <div style={{ fontSize: "11px", letterSpacing: "2px", color
            转折机制
          </div>
          <p style={{ margin: 0, fontSize: "14px", color: "#c8b898",
            {result.pivotMechanism}
          </p>
        </div>
      )}
    </div>
  )}

  {/* 动词链 */}
  {activeTab === "verbs" && (
    <div>
      <div style={{ fontSize: "11px", letterSpacing: "3px", color: "#
        黄金动词链
      </div>
      <div style={{ display: "flex", flexWrap: "wrap", gap: "8px", al
        {result.verbChain?.map((verb, i) => (
          <div key={i} style={{ display: "flex", alignItems: "center"
            <div style={{
              background: `rgba(212,175,55,${0.05 + (i / result.verbC
              border: "1px solid rgba(212,175,55,0.3)",
              borderRadius: "4px", padding: "8px 16px",
```

```
              fontSize: "14px", color: "#e0c878",
              letterSpacing: "1px"
            }}>
              {verb}
            </div>
            {i < result.verbChain.length - 1 && (
              <span style={{ color: "rgba(212,175,55,0.4)", fontSize:
            )}
          </div>
        ))}
      </div>
      <div style={{ fontSize: "11px", letterSpacing: "3px", color: "#
        终末归宿词
      </div>
      <div style={{ display: "flex", flexWrap: "wrap", gap: "8px" }}>
        {result.eschatologicalWords?.map((item, i) => (
          <div key={i} style={{
            background: "rgba(212,175,55,0.08)", border: "1px solid r
            borderRadius: "4px", padding: "6px 14px", fontSize: "13px
          }}>
            <span style={{ color: "#d4af37" }}>{item.term}</span>
            {item.reference && (
              <span style={{ color: "#6b7280", marginLeft: "8px", fon
                {item.reference}
              </span>
            )}
          </div>
        ))}
      </div>
    </div>
  )}

  {/* 神学洞见 */}
  {activeTab === "insight" && (
    <div>
      <div style={{ fontSize: "11px", letterSpacing: "3px", color: "#
        路径独特性
      </div>
      <p style={{
        fontSize: "15px", lineHeight: "1.9", color: "#c8b898",
        borderLeft: "2px solid rgba(212,175,55,0.3)",
        paddingLeft: "20px", margin: "0 0 28px"
      }}>
        {result.uniqueness}
      </p>
      <div style={{ fontSize: "11px", letterSpacing: "3px", color: "#
        最深的神学张力
```

```
                  </div>
                  <div style={{
                    background: "linear-gradient(135deg, rgba(212,175,55,0.08), r
                    border: "1px solid rgba(212,175,55,0.2)",
                    borderRadius: "8px", padding: "20px 24px"
                  }}>
                    <p style={{ margin: 0, fontSize: "15px", lineHeight: "1.9", c
                      「{result.theologicalTension}」
                    </p>
                  </div>
                </div>
              )}
            </div>
          </div>
        )}
      </div>
    </div>
  );
}

export default GoldenPathAnalyzer;
```