

# Resume AI Analyzer

## Information Extractor

### Input Schema:

```
{
  "type": "object",
  "properties": {
    "name": {
      "type": "string"
    },
    "email": {
      "type": "string",
      "format": "email"
    },
    "phone_number": {
      "type": "string",
      "pattern": "^(\\+\\d{1,3}[- ]?)?\\d{10}$"
    }
  },
  "required": ["name", "email", "phone_number"]
}
```

## AI Agent

### Prompt:

Resume: {{ \$('Extract from File').item.json.text }}

### System Message:

#### \*\* Purpose \*\*

You are a CV Evaluation and Summarization Agent. Your task is to extract key information from a CV and output it in a clean, structured format. You must also evaluate how well the candidate fits a specified job role and assign a relevance score (1–10), followed by a short justification.

#### \*\* How It Works \*\*

When provided with a CV and job description, follow this process:

#### 1. Extract & Summarize Information from the CV

##### - Educational Qualifications

Include degrees, institutions, and graduation years.

##### - Job History

List job titles, company names, employment dates, and 1 short sentence summarizing key responsibilities or achievements.

##### - Skill Set

List technical, soft, or role-specific skills — separated by commas.

#### 2. Evaluate the Candidate for the Job Role

- Compare the candidate's qualifications and experience with the job requirements.

- Assign a score from 1–10 based on relevance:

1–3: Weak match

4–6: Moderate match

7–8: Strong match

9-10: Excellent match

- Provide a brief justification for the score. Mention strengths and gaps directly related to the role — no assumptions or interpretations.

## Code

### JavaScript:

```
// ----- 1. Read input text -----
const inputData = $input.item.json.content;

// ----- 2. Helper: extract a block that starts with a bold header -----
const grabBlock = (text, header) => {
  const re = new RegExp(`\\{0,2}\\${header}\\{0,2\\}\\s*([\\s\\S]*)\\s*$`, 'i'); // capture to end of string
  const m = re.exec(text);
  return m ? m[1].trim() : null;
};

// ----- 3. Helper: extract a section that sits between bold headers -----
const grabSection = (text, label) => {
  /* takes "***Label:** ..." up to the next **Bold Header** or end */
  const re = new RegExp(
    `\\{0,2}\\${label}\\{0,2\\}\\s*([\\s\\S]*)?(?!=\\n\\{0,2\\}[A-Z][^\\n]*\\{0,2\\}\\s*$)`,
    'i'
  );
  const m = re.exec(text);
  return m ? m[1].trim() : null;
};

// ----- 4. Pull the three simple sections -----
const educationalQualifications = grabSection(inputData, 'Educational Qualifications:');
const jobHistory = grabSection(inputData, 'Job History:');
const skillSet = grabSection(inputData, 'Skill Set:');

// ----- 5. Pull the whole "Evaluation ..." block -----
const evaluationBlock = grabBlock(inputData, 'Evaluation'); // matches "***Evaluation ...**"

// ----- 6. From that block, get score & justification -----
let score = null;
let justification = null;

if (evaluationBlock) {
  // score can be "Relevance Score: 7/10" or "Score: 7/10"
  const scoreMatch = /(Relevance\\s*)?Score:\\s*(\\d{1,2})\\s*\\/\\s*10/i.exec(evaluationBlock);
  score = scoreMatch ? scoreMatch[2] : null;

  const justMatch = /(\\{0,2\\}Justification:\\{0,2\\}\\s*([\\s\\S]*)/i.exec(evaluationBlock);
  if (justMatch) {
    justification = justMatch[1].trim();
  }
}

// ----- 7. Return tidy JSON -----
return [
  {
    educationalQualifications,
    jobHistory,
    skillSet,
    score,
    justification
  }
];
```

```
}  
};
```

**Google Sheets** ← Copy Here