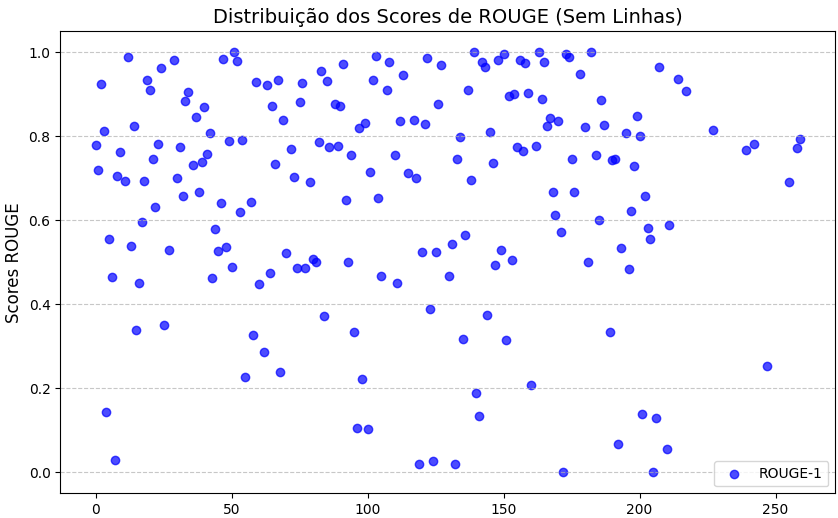
### **Análise dos Resultados Individuais**

#### **1. ROUGE-1 (Unigramas)**

O ROUGE-1 mede a similaridade baseada em unigramas (tokens individuais), capturando a correspondência lexical entre o código humano e o código gerado:

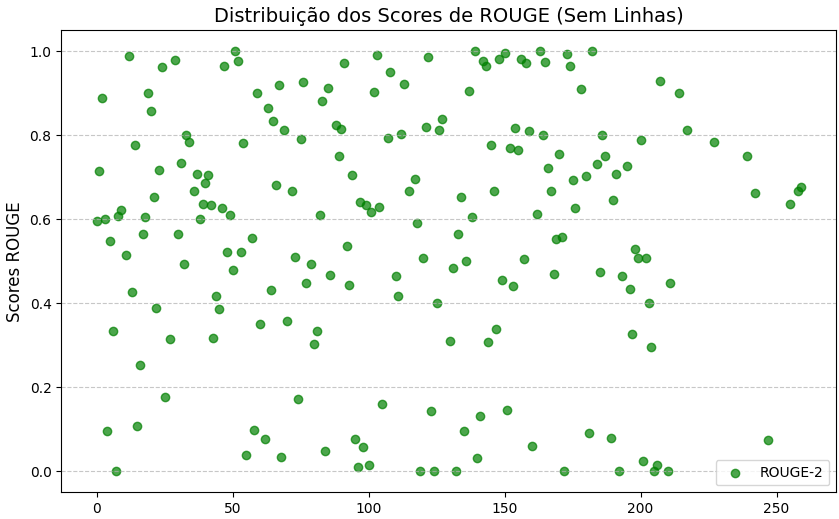


* **Distribuição:**
  + A maioria dos exemplos apresenta scores entre **0.6 e 1.0**, indicando que o código gerado muitas vezes utiliza tokens semelhantes aos encontrados no código humano.
  + Alguns casos possuem scores próximos de **0.0**, sugerindo diferenças lexicais significativas.
* **Interpretação:**
  + Scores altos sugerem que a máquina capturou corretamente os tokens mais relevantes, mas não garantem que o código esteja correto semanticamente.
  + Casos de baixo ROUGE-1 indicam que o modelo pode ter gerado códigos que utilizam abordagens ou termos diferentes.

#### 

#### **2. ROUGE-2 (Bigramas)**

O ROUGE-2 avalia a similaridade em termos de bigramas (pares de tokens), sendo uma métrica mais restritiva:

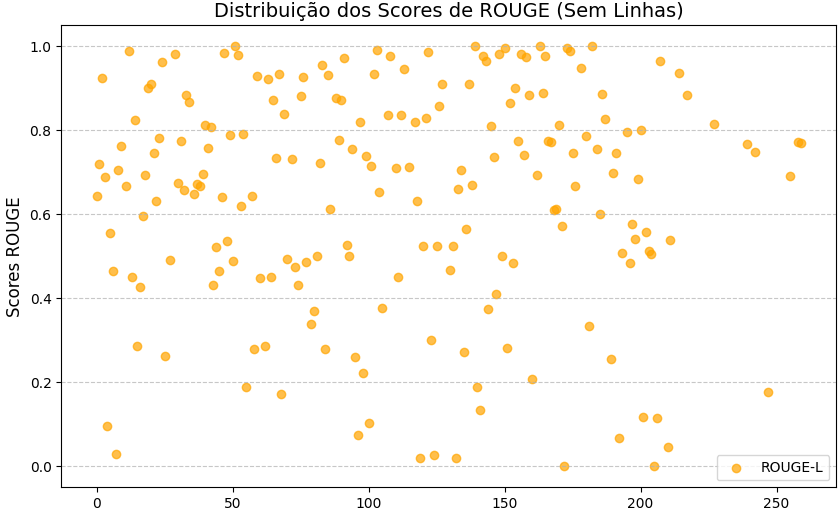


* **Distribuição:**
  + Scores são geralmente mais baixos do que no ROUGE-1, com uma concentração em torno de **0.4 a 0.8**.
  + Alguns exemplos apresentam scores baixos (< **0.2**), indicando pouca correspondência de pares de tokens.
* **Interpretação:**
  + Scores moderados sugerem que o modelo captura sequências curtas de tokens humanos, mas pode não ser consistente em abordagens mais complexas.
  + Scores baixos refletem dificuldades na geração de sequências que preservem relações lexicais.

#### 

#### **3. ROUGE-L (Subsequência Longa Comum)**

O ROUGE-L mede a similaridade com base na maior subsequência comum (LCS), capturando padrões estruturais mais amplos:

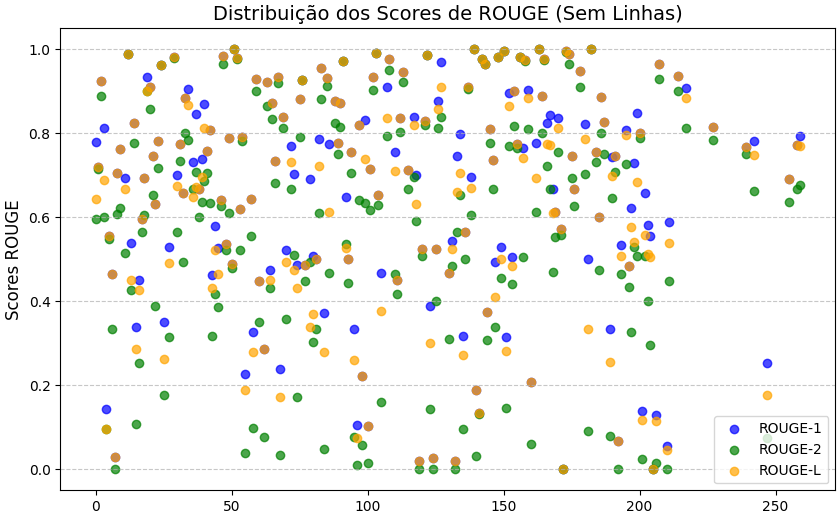


* **Distribuição:**
  + Muitos exemplos apresentam scores entre **0.7 e 1.0**, sugerindo que o código gerado frequentemente reflete a estrutura geral do código humano.
  + Scores baixos (< **0.3**) são menos frequentes, mas indicam divergências significativas na organização estrutural.
* **Interpretação:**
  + Scores altos mostram que o modelo frequentemente preserva a ordem e organização geral do código.
  + Scores baixos podem indicar alterações significativas na abordagem ou estrutura lógica do código gerado.

### 

### **Consolidação dos Resultados**

Ao combinar os resultados de ROUGE-1, ROUGE-2 e ROUGE-L, observamos os seguintes padrões gerais:



1. **Alta Similaridade em ROUGE-L:**
   * A métrica ROUGE-L apresenta consistentemente scores mais altos, sugerindo que o modelo é eficaz em replicar a estrutura geral do código humano.
2. **Diferenças Entre ROUGE-1 e ROUGE-2:**
   * A discrepância entre ROUGE-1 e ROUGE-2 indica que o modelo é mais eficaz em capturar palavras individuais do que relações mais complexas entre tokens.
3. **Casos de Baixa Similaridade em Todas as Métricas:**
   * Alguns exemplos apresentaram scores baixos em todas as métricas, indicando que o código gerado diverge significativamente do código humano. Esses casos precisam ser investigados para identificar se representam falhas do modelo ou soluções alternativas válidas.

### 

### **Críticas e Limitações do Experimento**

1. **Viés na Base de Treinamento:**
   * Se a base de treinamento contém códigos de baixa qualidade, o modelo pode replicar soluções inadequadas, mesmo com scores altos.
2. **Limitação das Métricas ROUGE:**
   * As métricas avaliam apenas a similaridade lexical e estrutural, não garantindo a correção semântica ou funcionalidade do código gerado.
3. **Casos de Baixa Similaridade:**
   * Scores baixos podem não refletir erros, mas sim soluções diferentes para o mesmo problema, o que não é capturado pelas métricas ROUGE.

### **Recomendações**

1. **Complementar as Métricas:**
   * Use outras métricas, como BLEU e Similaridade do Cosseno (CodeBERT), para capturar aspectos semânticos e funcionais do código.
2. **Análise Manual:**
   * Inspecione manualmente exemplos com scores baixos em todas as métricas para avaliar a validade das soluções geradas.
3. **Treinamento com Dados de Alta Qualidade:**
   * Utilize bases de treinamento com códigos revisados e bem estruturados para reduzir o impacto de viés negativo.
4. **Validação Automática:**
   * Integre ferramentas como SonarQube ou Checkmarx para avaliar a qualidade e segurança do código gerado.

### **Conclusão**

O experimento revelou que as métricas ROUGE são úteis para avaliar similaridade lexical e estrutural entre códigos, mas têm limitações significativas na avaliação de funcionalidade e correção semântica.

Embora o modelo mostre bom desempenho em replicar estrutura e tokens relevantes, é essencial complementar a avaliação com métricas semânticas e testes funcionais para garantir a qualidade e segurança do código gerado.