

Árvores Trie

Projeto de arquivos
Prof. Allan Rodrigo Leite

Um problema

- Objetivo
 - Localizar um dado que corresponde à chave informada
 - A chave e os dados mantidos no conjunto faz parte de um léxico
 - A busca deve considerar uma aproximação de correspondência
- Contexto
 - Conjunto de nomes de pessoas
 - Há nomes com grafias semelhantes
 - Manuel/Manoel
 - Luis/Luiz
 - Podem ocorrer erros na entrada de dados (chave)

Árvores Trie

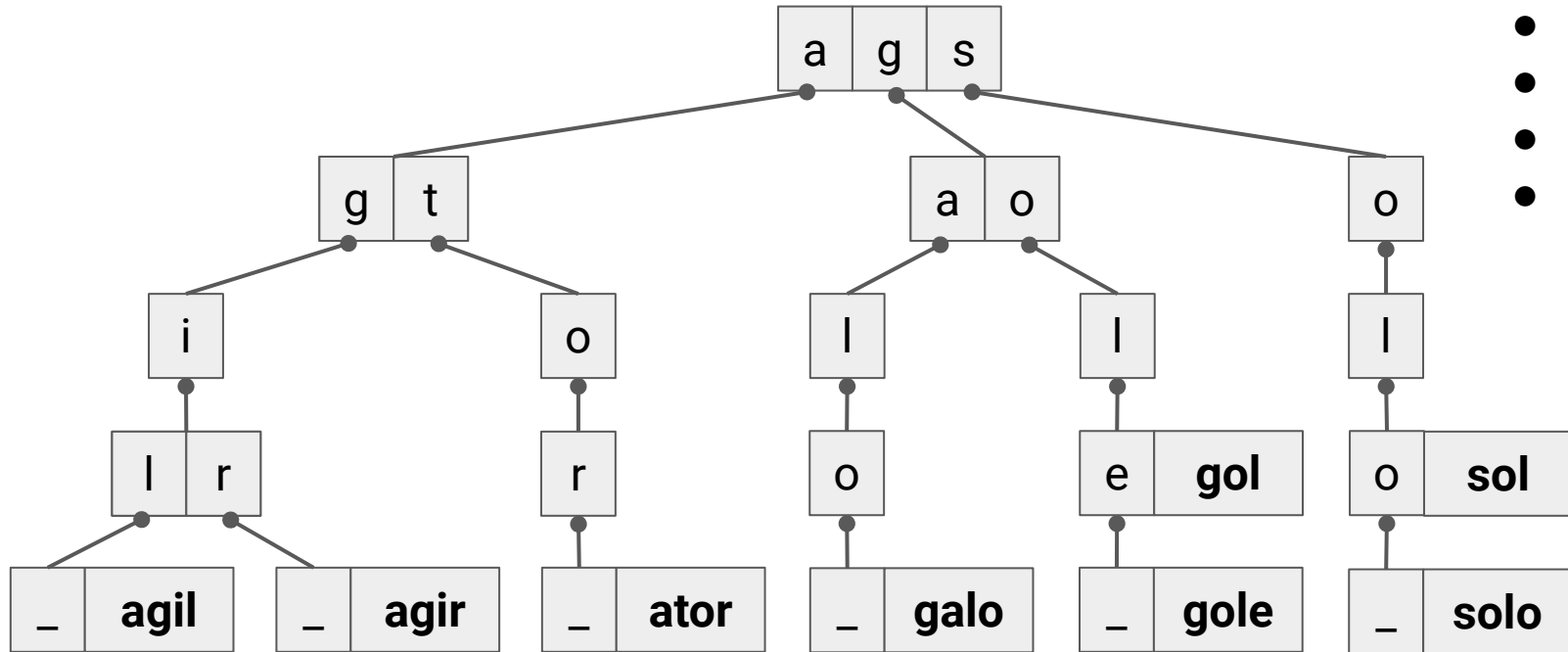
- Introduzida por Edward Fredkin em 1960
 - O nome Trie tem origem em Retrieval (recuperar)
 - No caso do propósito da árvore, recuperação dados
- Estrutura da árvore
 - Cada nó contém informações sobre um ou mais símbolos do léxico
 - O léxico (alfabeto) é finito e pode abranger os símbolos como:
 - $\{ 0, 1, \dots \} \rightarrow$ números
 - $\{ A, B, C, D, \dots \} \rightarrow$ caracteres
 - $\{ \} \rightarrow$ nulo

Árvores Trie

- São adequadas para tratamento lexicográfico
 - Manuseio de dicionário
 - Pesquisas em texto de grande dimensão
 - Construção de índices de documentos baseado em léxico
 - Expressão regular como padrão de pesquisa
- Estratégia
 - Caminho da raiz para qualquer nó representa um prefixo de uma string
 - Todos os descendentes diretos do mesmo pai são agrupados
 - Em um nó folha, o último caractere da palavra sendo procurada deverá apontar para o próprio nó

Árvores Trie

- agir
- agil
- ator
- galo
- gol
- gole
- sol
- solo



Árvores Trie

- Estratégia (cont.)
 - Cada nível da árvore explorado corresponde ao avanço de um símbolo na chave de pesquisa
 - Cada nó pode conter informação sobre um ou mais símbolos do alfabeto utilizado
 - Uma sequência de filhos em um dado nó pode formar qualquer palavra com base neste léxico (alfabeto)
 - Não existe limite para:
 - Tamanho de uma sequência de filhos de um nó
 - Tamanho de uma chave

Aplicações de árvores Trie

- Corretor ortográfico
 - As palavras são comparadas com um dicionário armazenado em arquivo
 - Se não são encontradas, indica-se as opções para correção
- Procedimentos
 - Armazenar dicionário em uma trie
 - Percorrer a trie símbolo por símbolo para localizar a palavra testada
 - Caso seja detectado um erro na chave, o algoritmo sugere correções
 - Indicando as palavras do dicionário formadas a partir de um nó

Aplicações de árvores Trie

- Sugestões possíveis para correção
 - Substituição
 - Avança um caracter na chave e avança um nível na árvore
 - Deleção
 - Avança um nível na árvore
 - Inserção
 - Avança um caracter na chave
 - Transposição
 - Avança um nível na árvore e testa a posição atual da chave
 - Se coincidir, avança um caracter na chave
 - Retrocede um nível na árvore para confirmar a inversão

Aplicações de árvores Trie

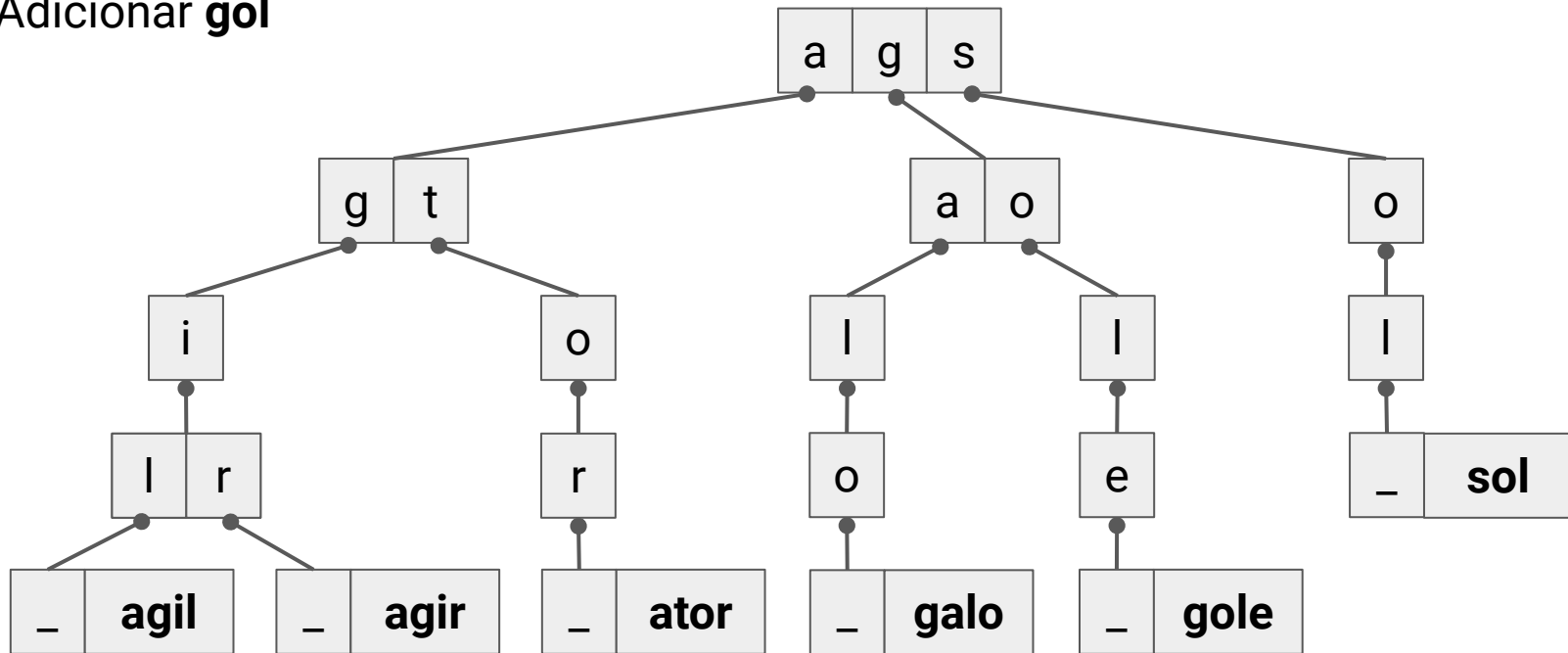
- Auto-preenchimento
 - Armazena palavras mais utilizadas em uma trie
 - A medida que um novo símbolo é adicionado na chave, serão exibidas as opções possíveis de palavras mais utilizadas
- Procedimentos
 - Armazenar dicionário em uma trie
 - Percorrer a trie símbolo por símbolo para localizar a palavra testada
 - Indica as palavras do dicionário formadas a partir de um nó, considerando a frequência de uso delas
 - Requer uma estrutura adicional para manter esta frequência de uso

Operações sobre árvores Trie

- Adição
 - Realizada uma busca pela palavra a ser inserida
 - Se a palavra já existir na trie, nada é feito
 - Caso contrário, é recuperado o nó até o ponto onde acontece a maior subcadeia da palavra a ser inserida
 - O restante dos seus caracteres (prefixo) serão adicionados na trie a partir daquele nó

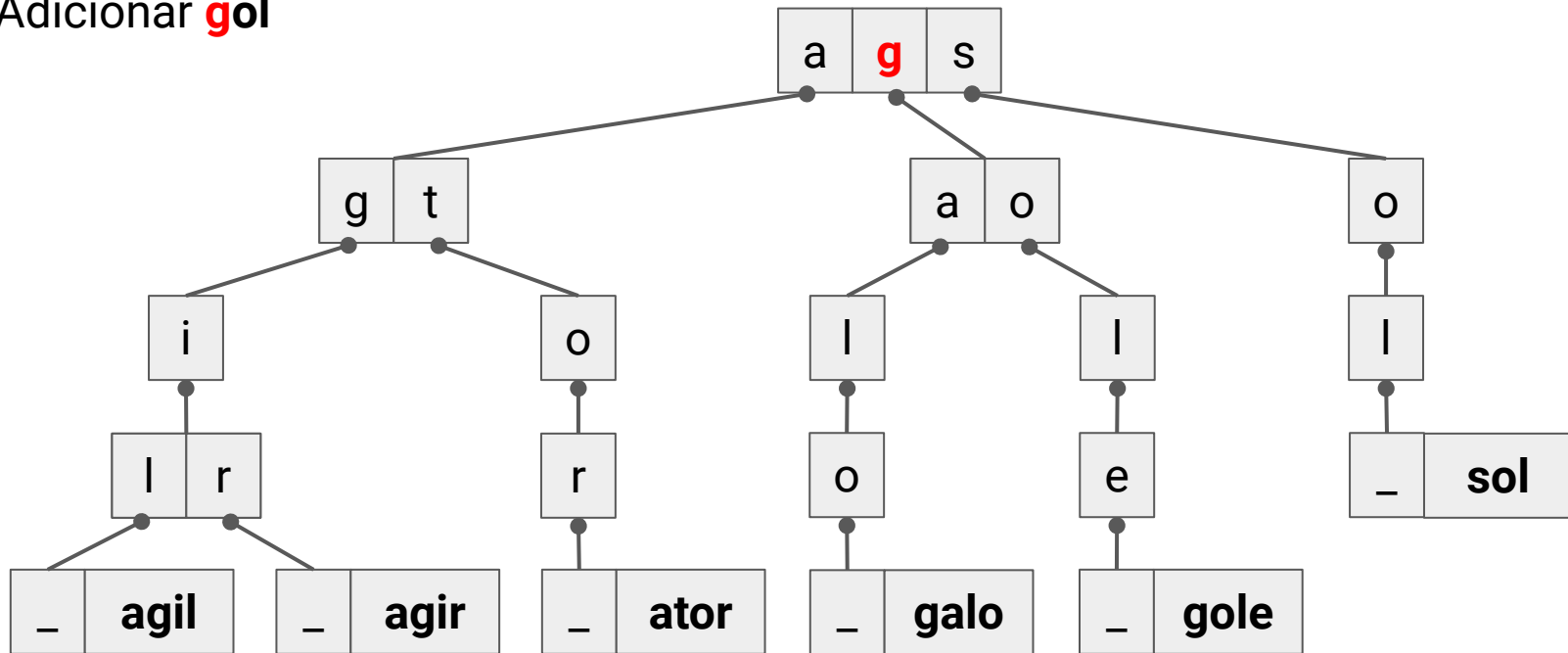
Árvores Trie

Adicionar **gol**



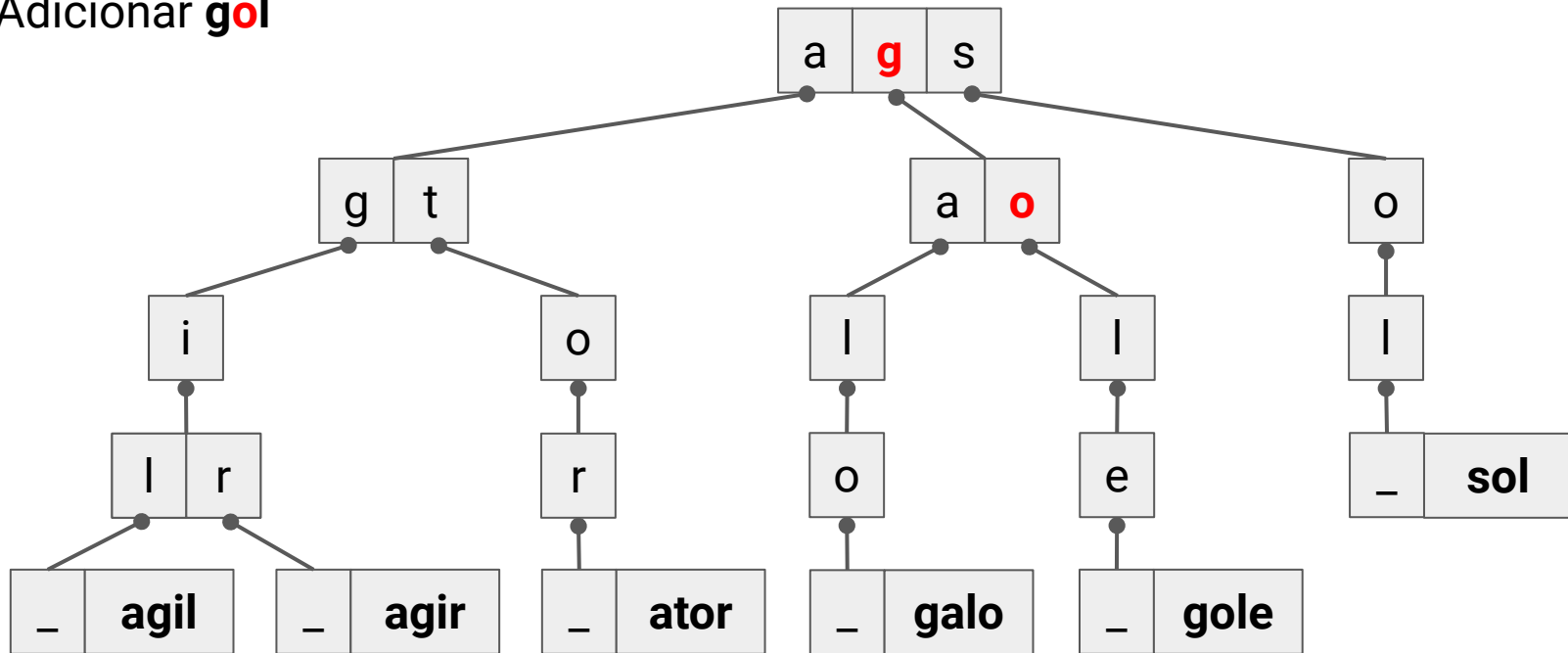
Árvores Trie

Adicionar **gol**



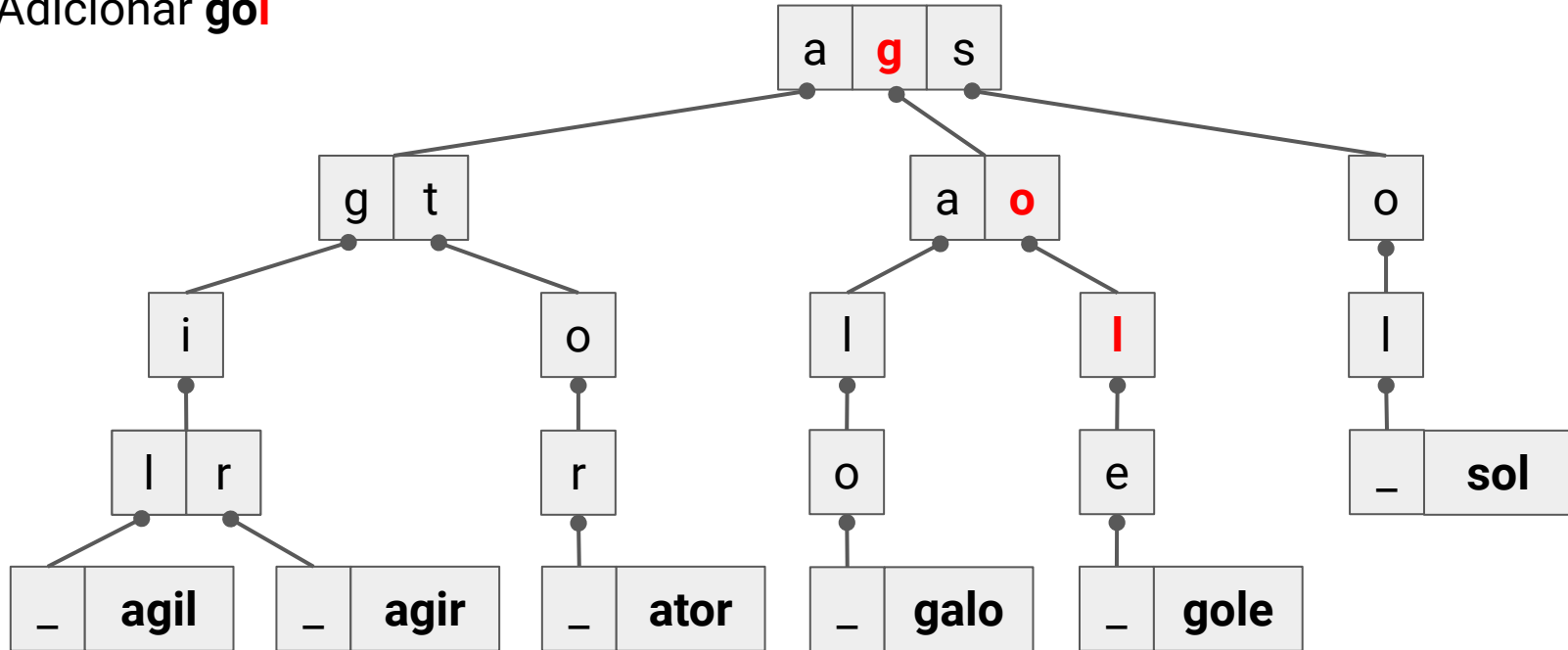
Árvores Trie

Adicionar **gol**



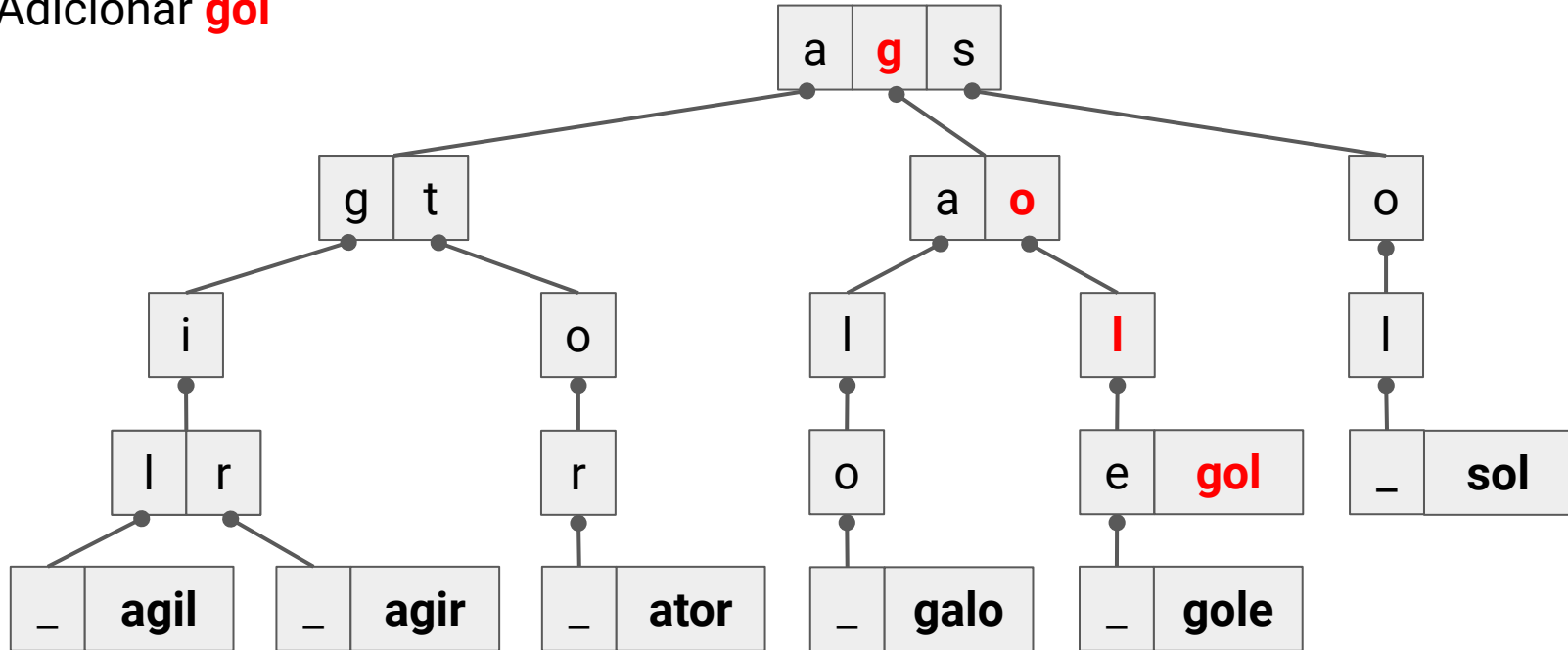
Árvores Trie

Adicionar **go**



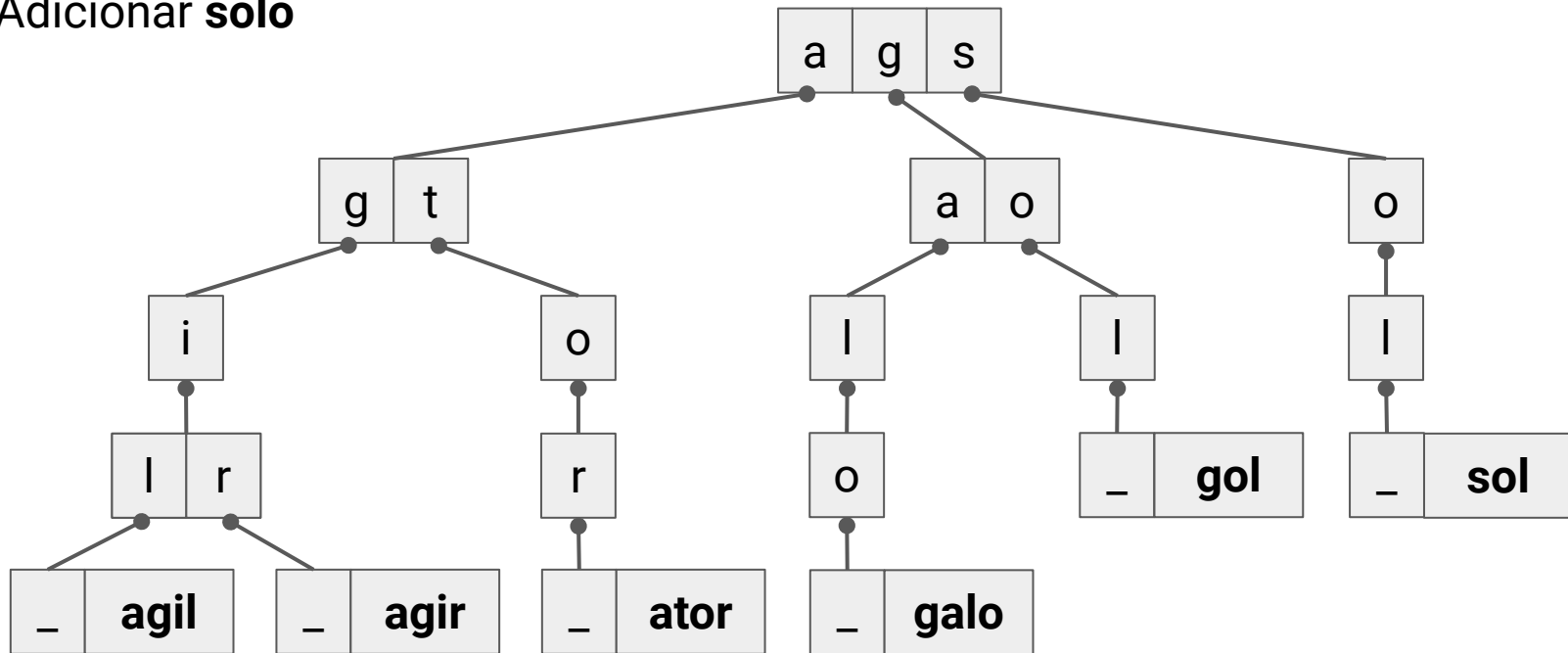
Árvores Trie

Adicionar **gol**



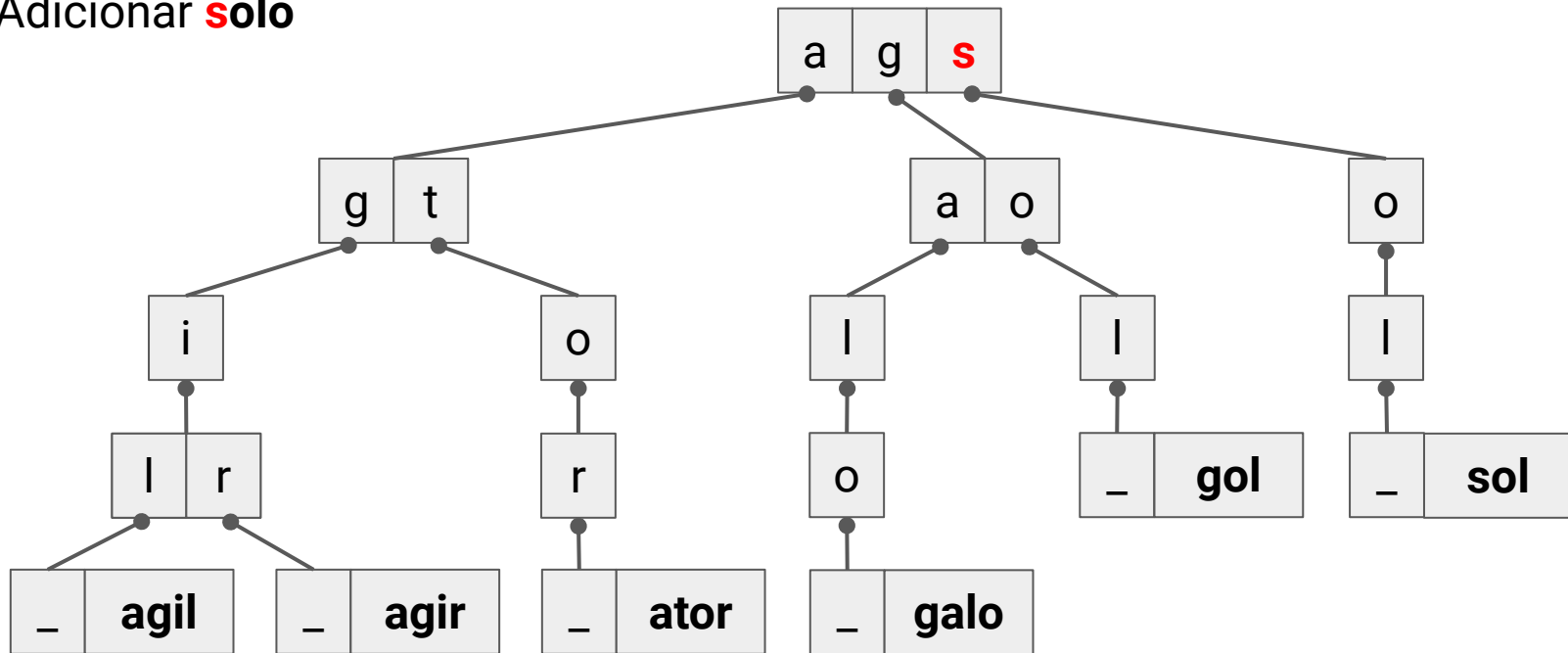
Árvores Trie

Adicionar **solo**



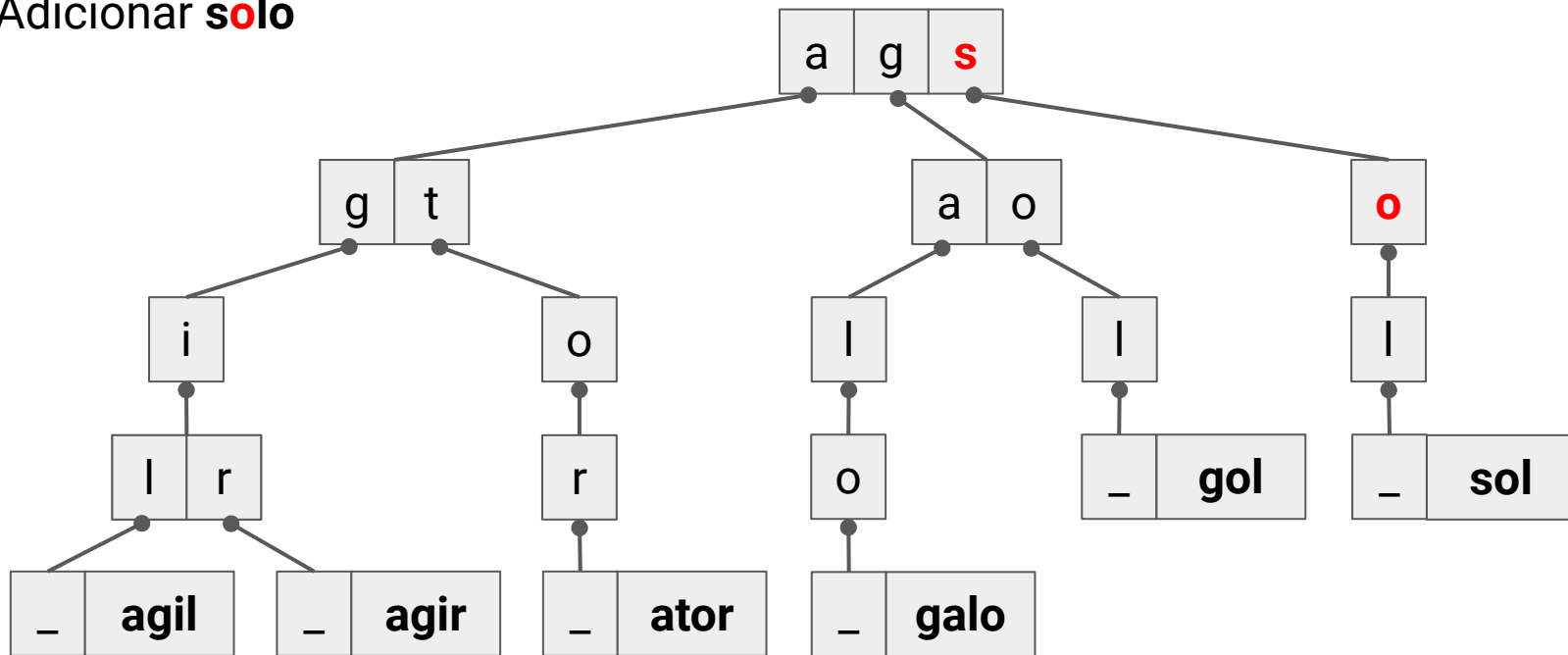
Árvores Trie

Adicionar **solo**



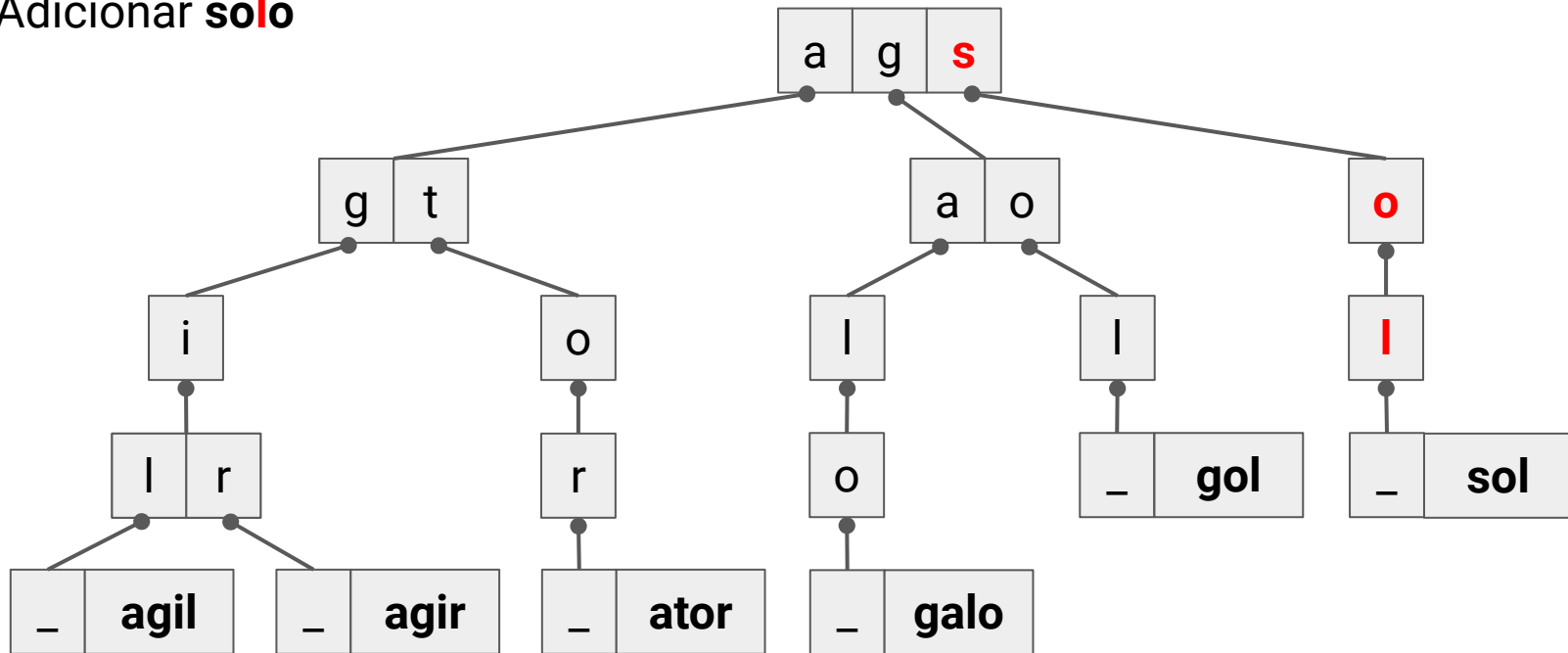
Árvores Trie

Adicionar **solo**



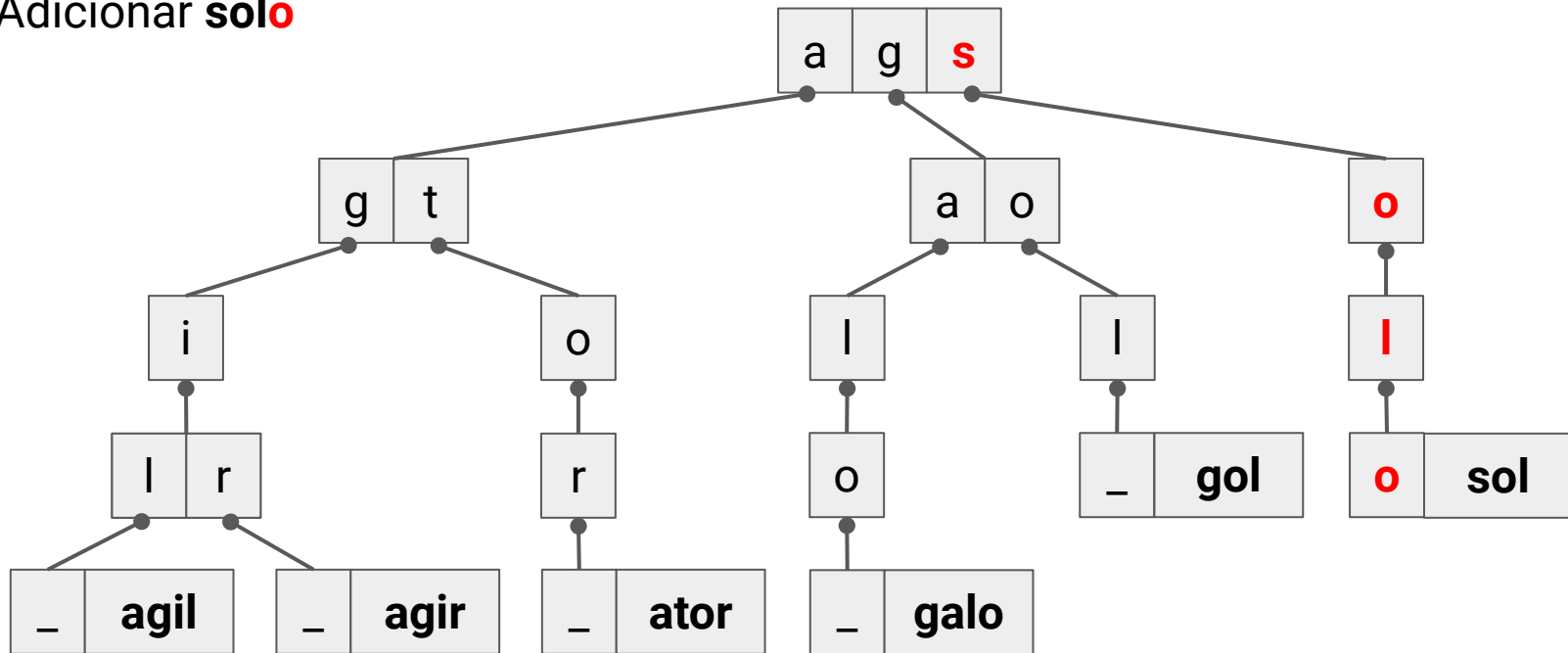
Árvores Trie

Adicionar **so**lo



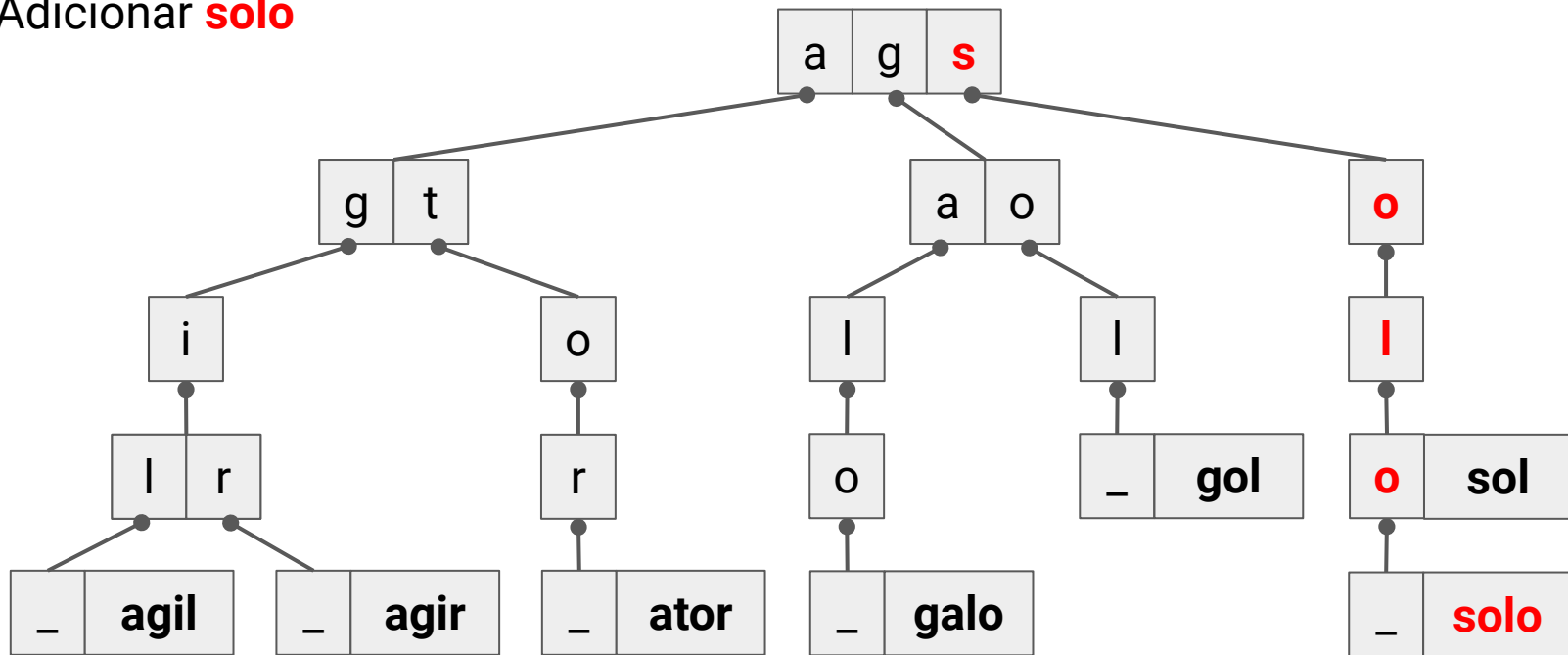
Árvores Trie

Adicionar **sol**



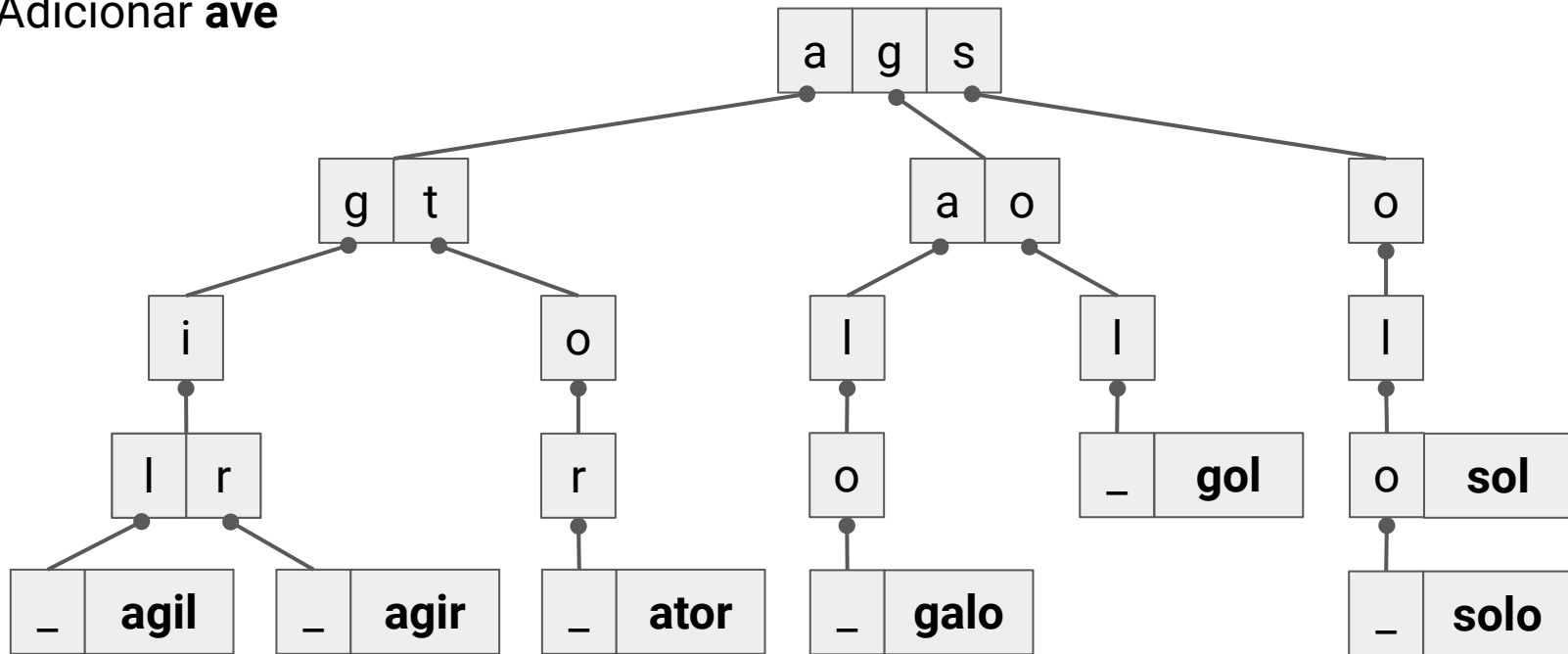
Árvores Trie

Adicionar **solo**



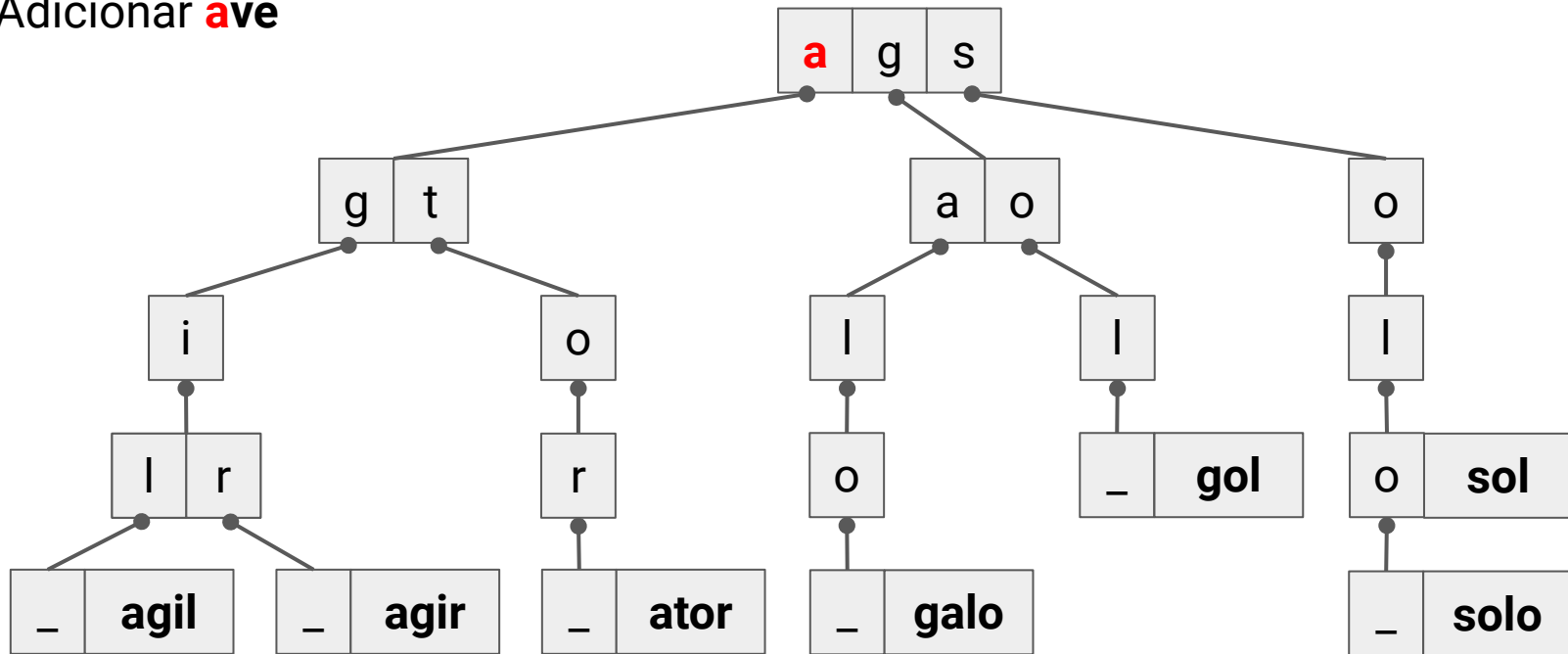
Árvores Trie

Adicionar **ave**



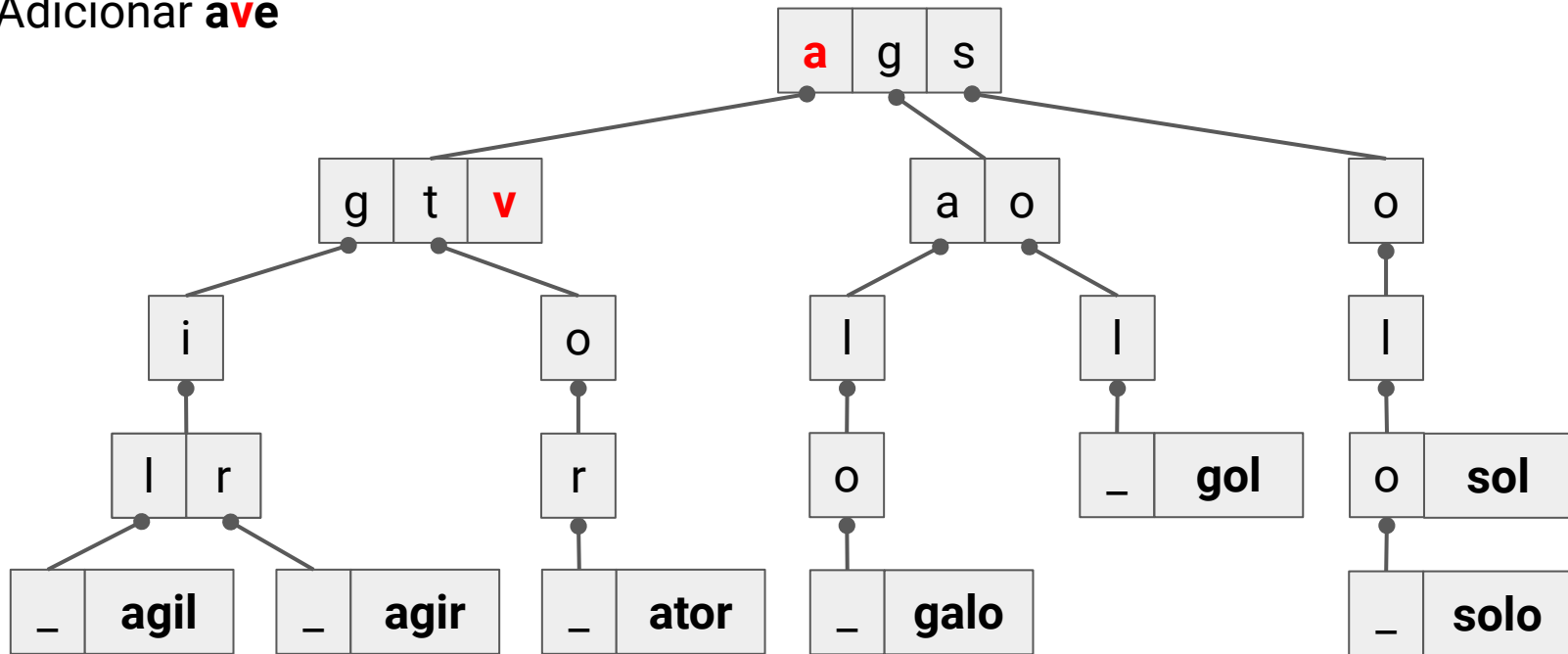
Árvores Trie

Adicionar **ave**



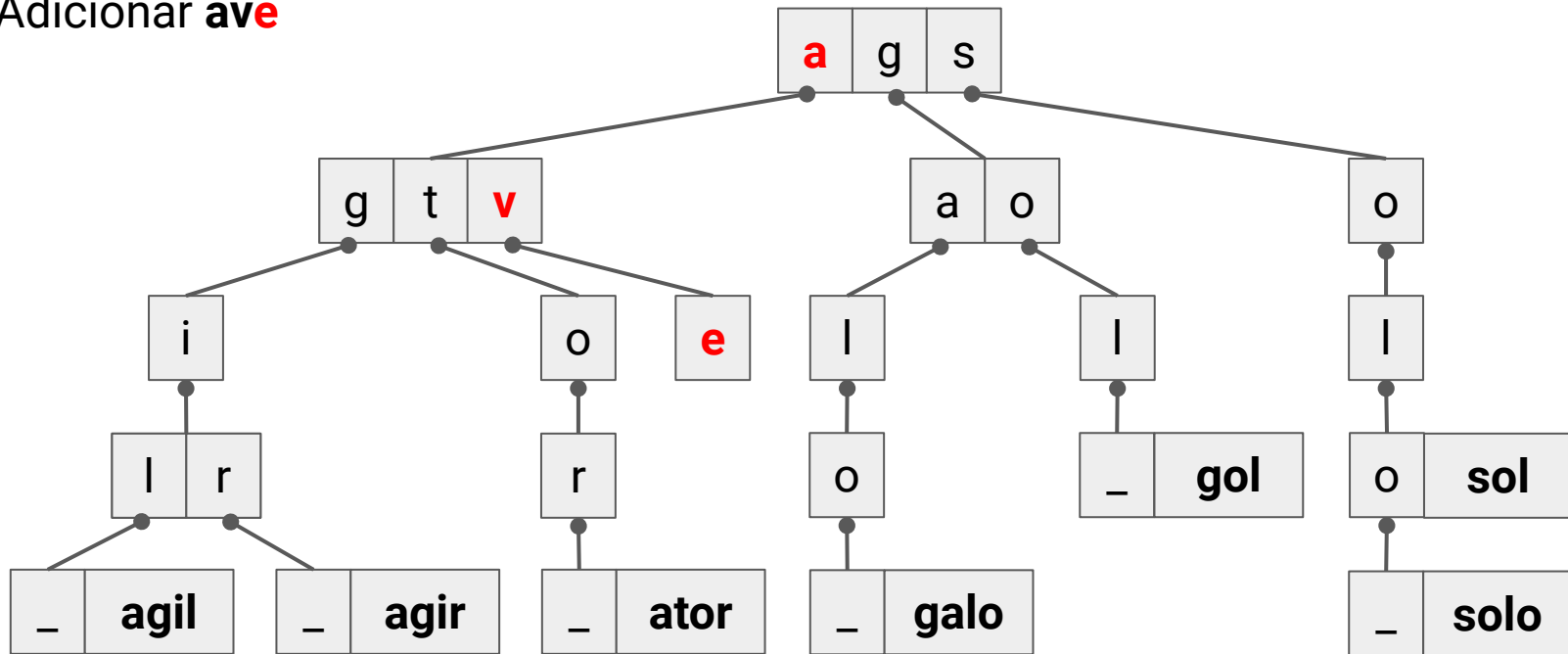
Árvores Trie

Adicionar **ave**



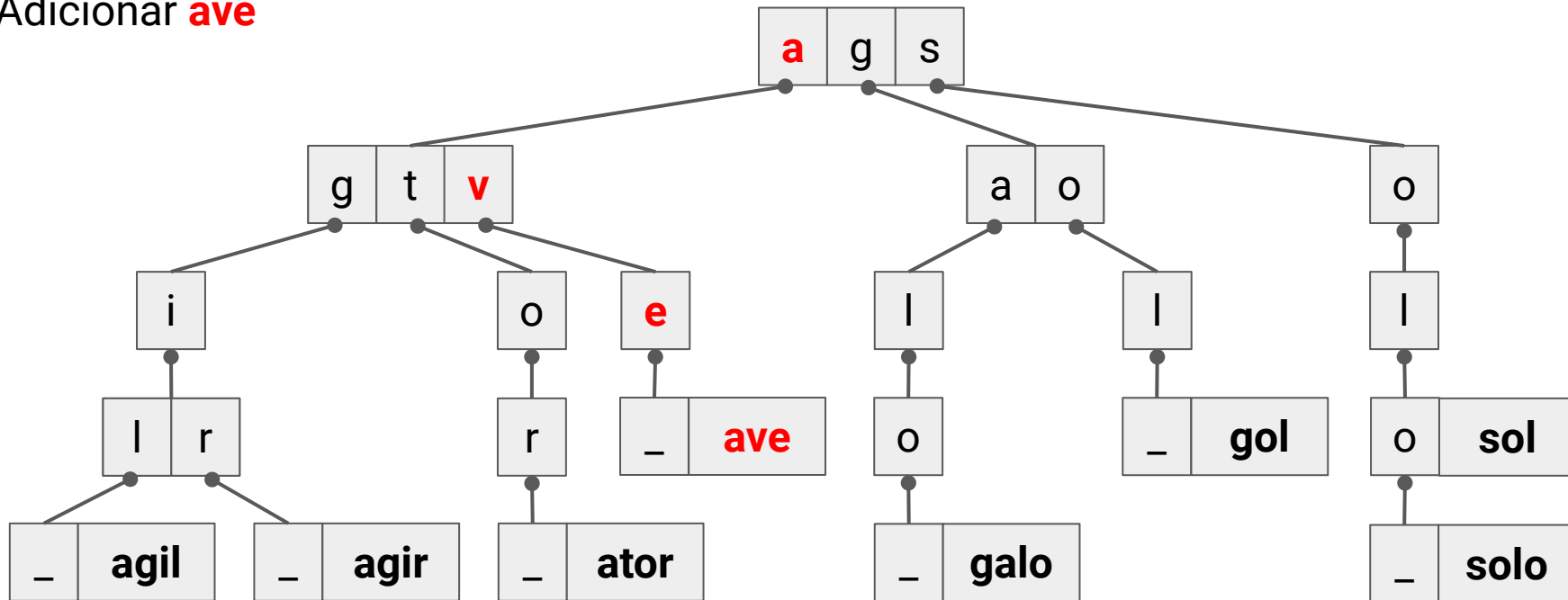
Árvores Trie

Adicionar **ave**



Árvores Trie

Adicionar **ave**

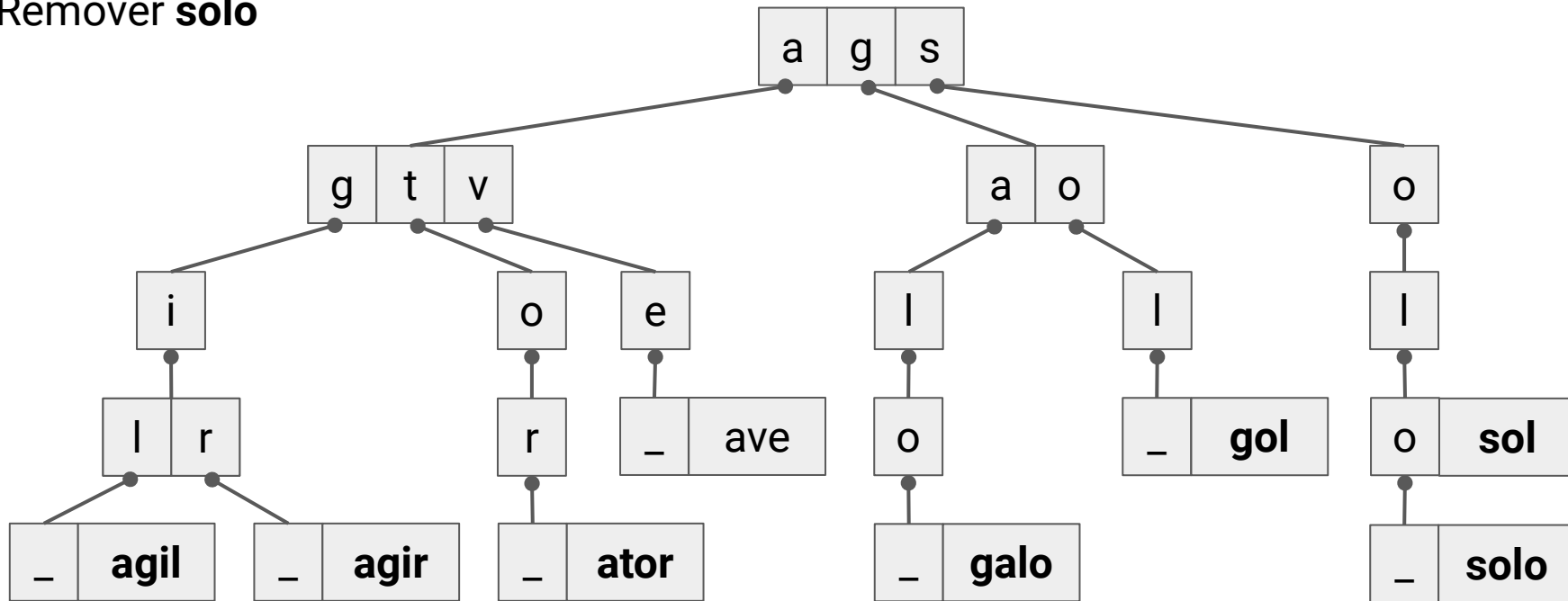


Operações sobre árvores Trie

- Remoção
 - Realizada uma busca pela palavra a ser excluída
 - Ao encontrar o nó que representa o final da palavra a ser removida
 - Remove-se os nós que possuem apenas um filho no caminho ascendente
 - A remoção é concluída quando se encontra um nó com mais de um filho

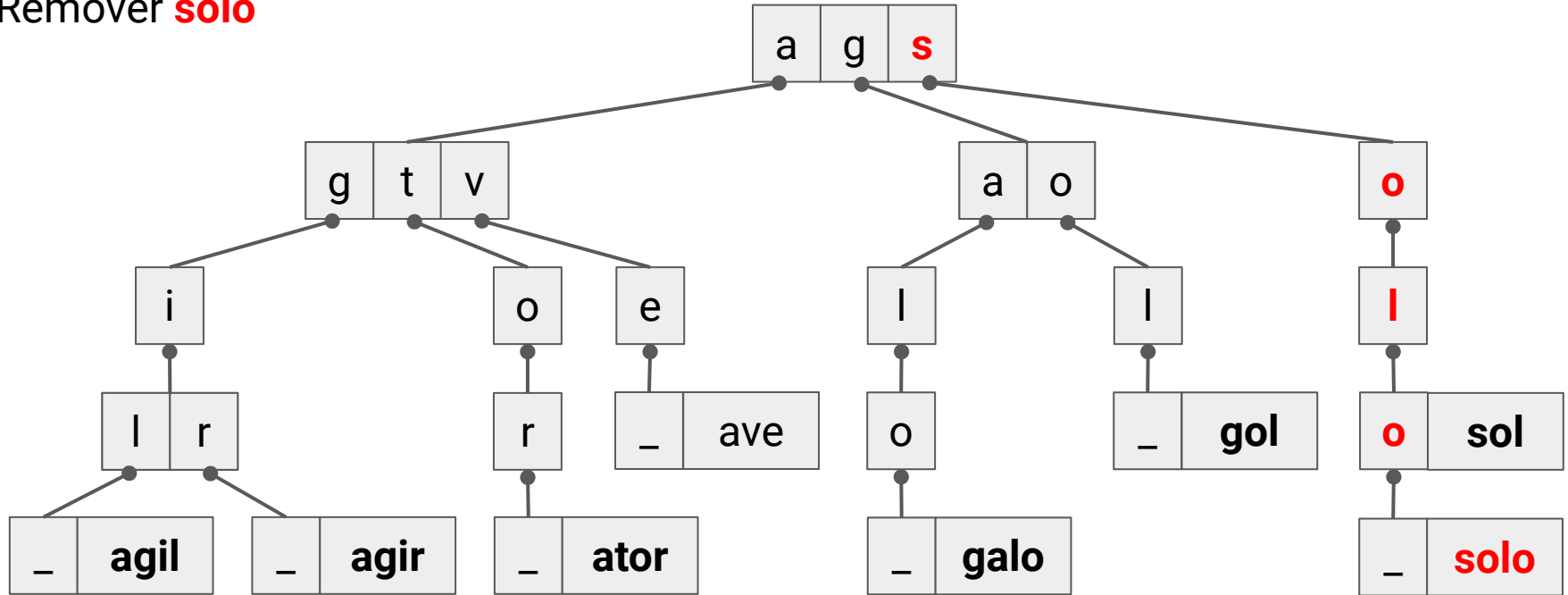
Árvores Trie

Remover **solo**



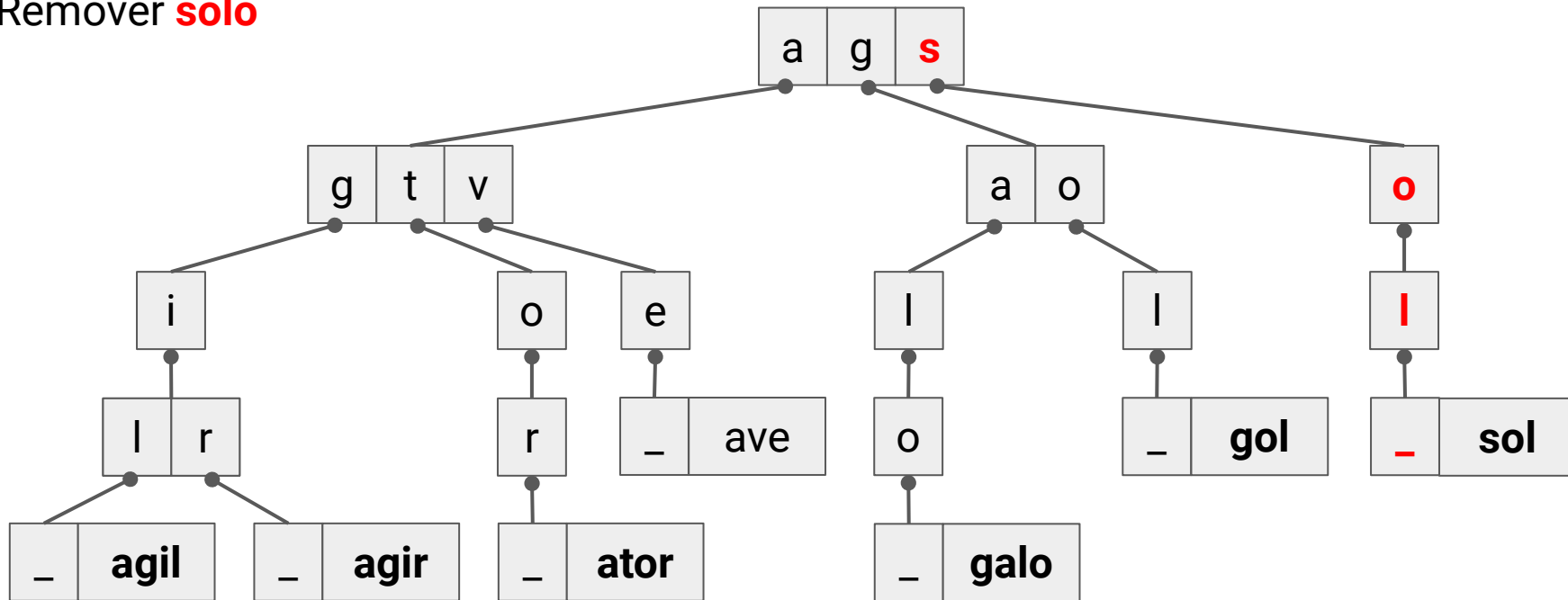
Árvores Trie

Remover **solo**



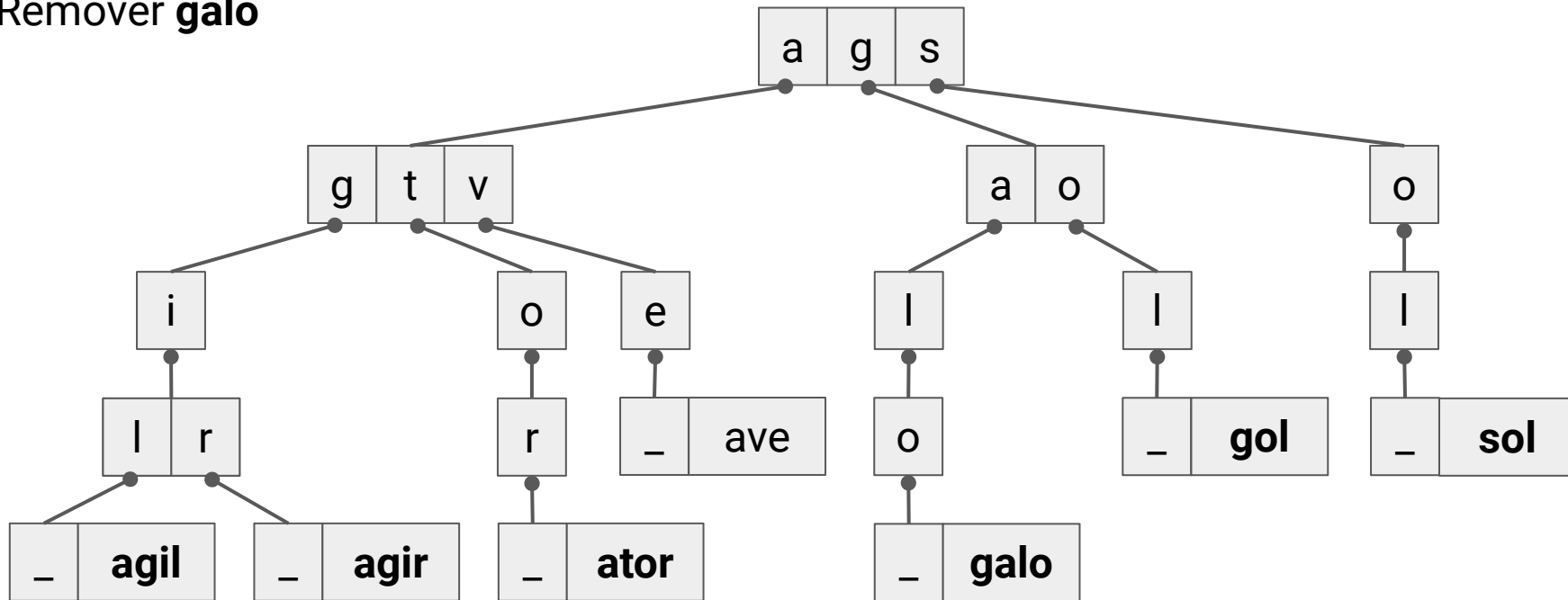
Árvores Trie

Remover **solo**



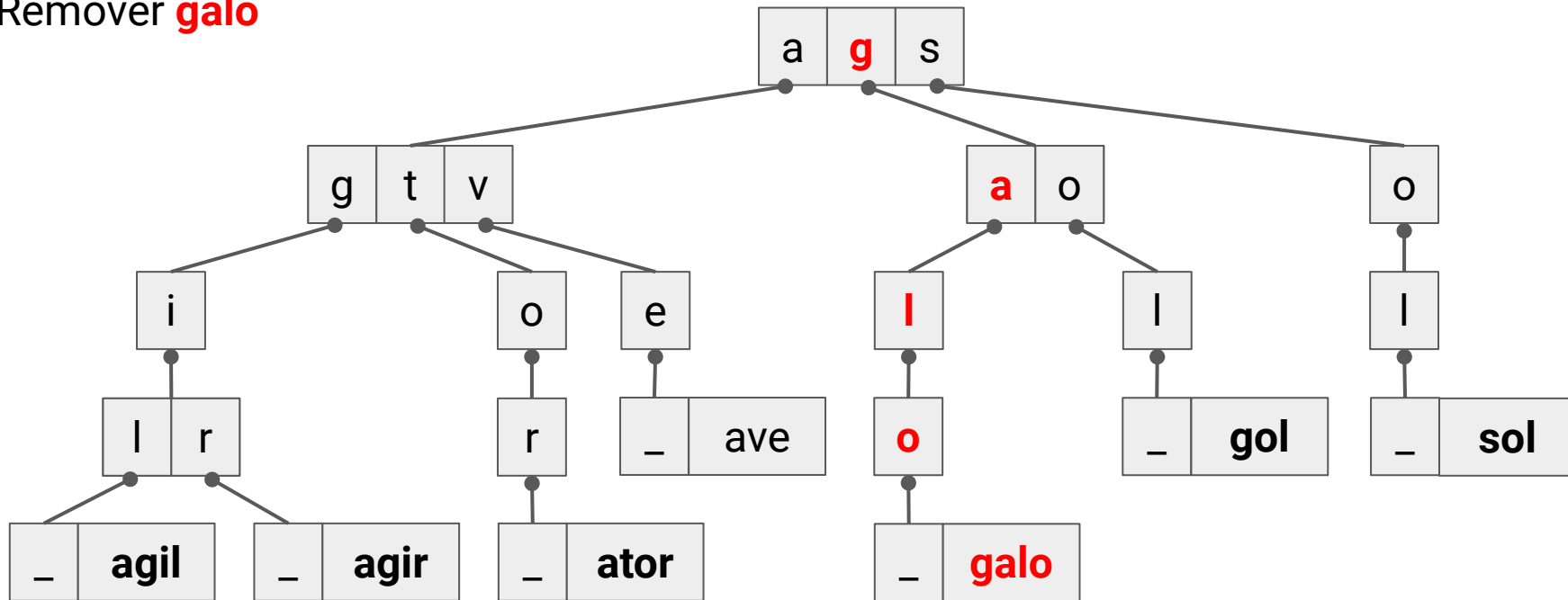
Árvores Trie

Remover **galo**



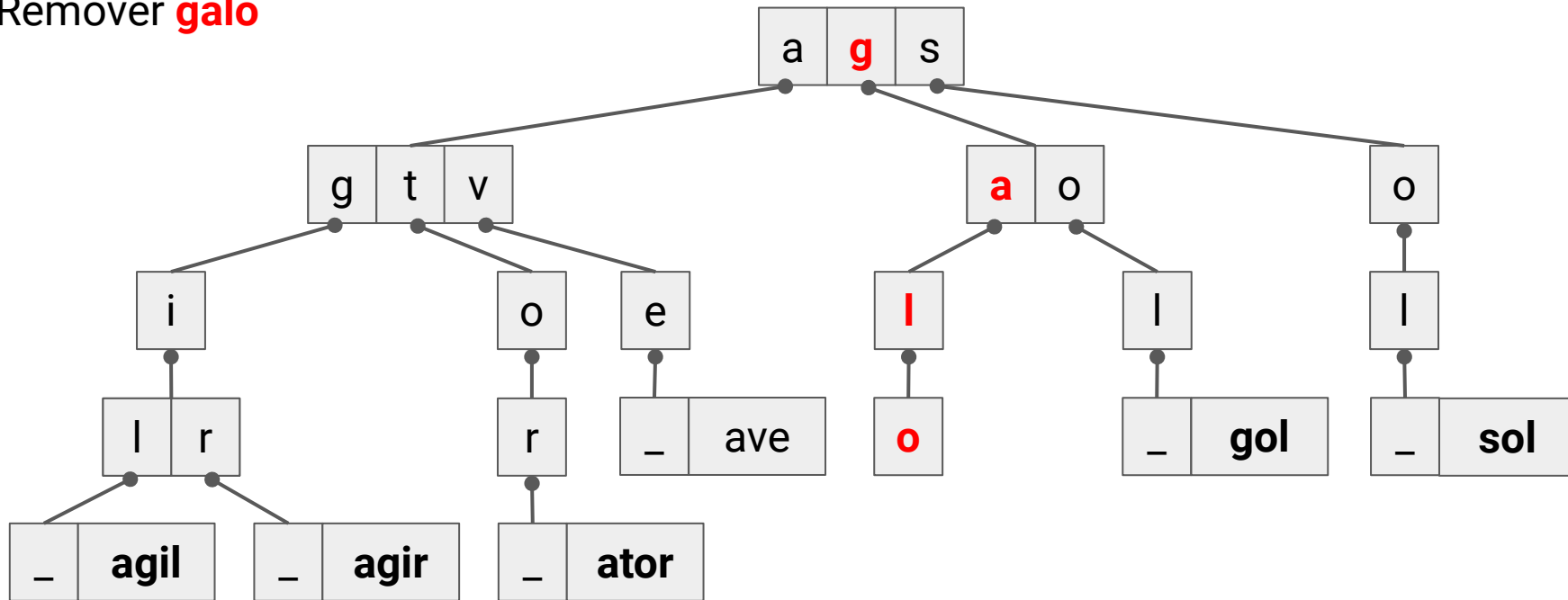
Árvores Trie

Remover **galo**



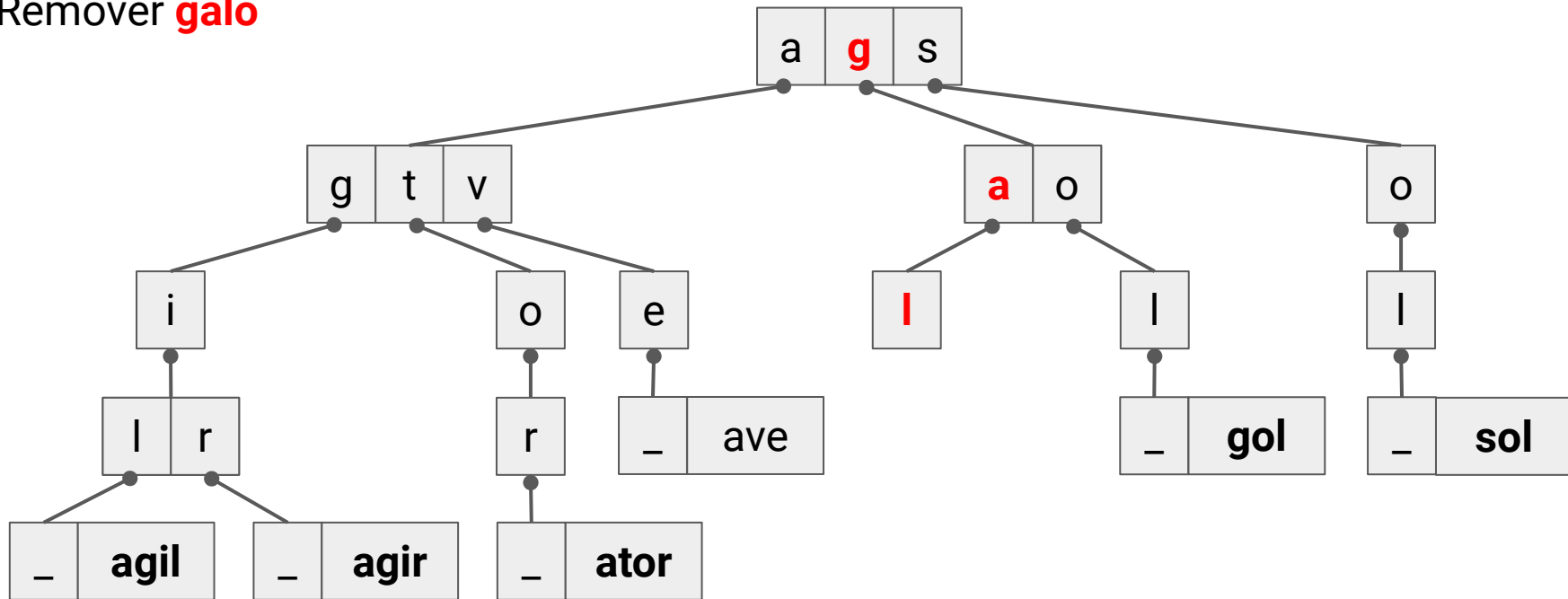
Árvores Trie

Remover galo



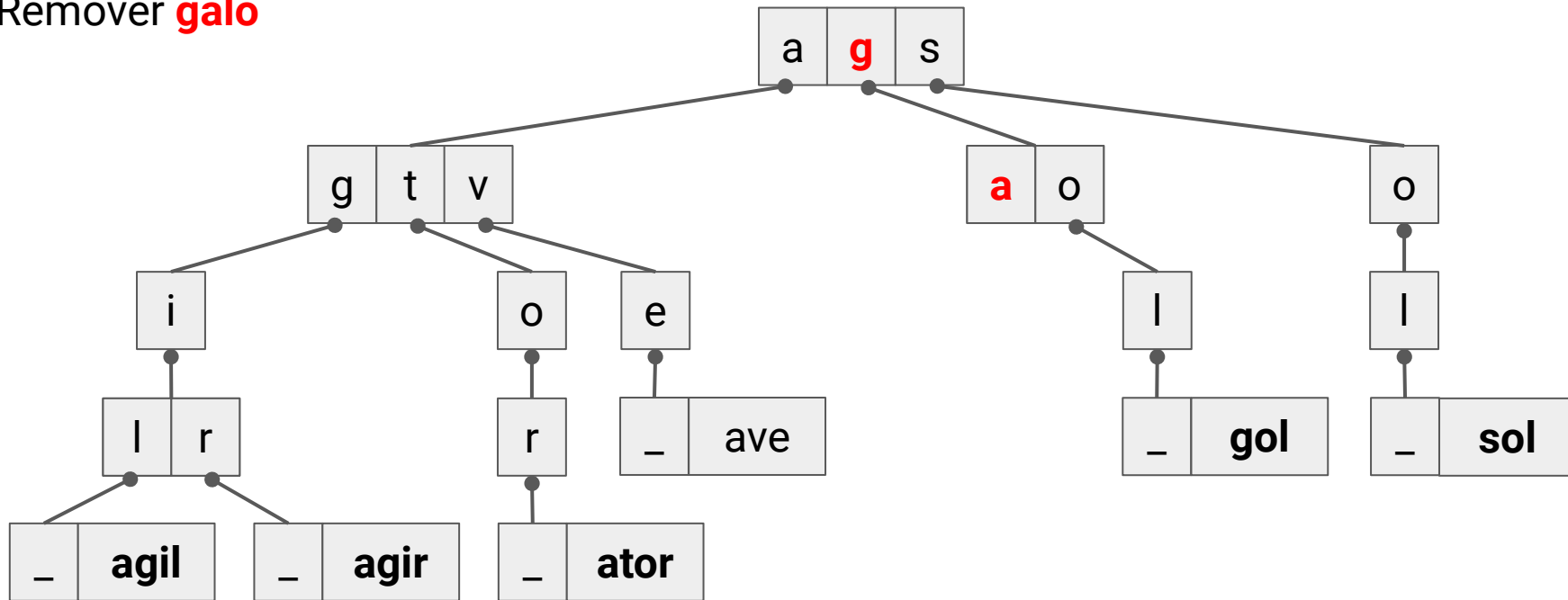
Árvores Trie

Remover galo



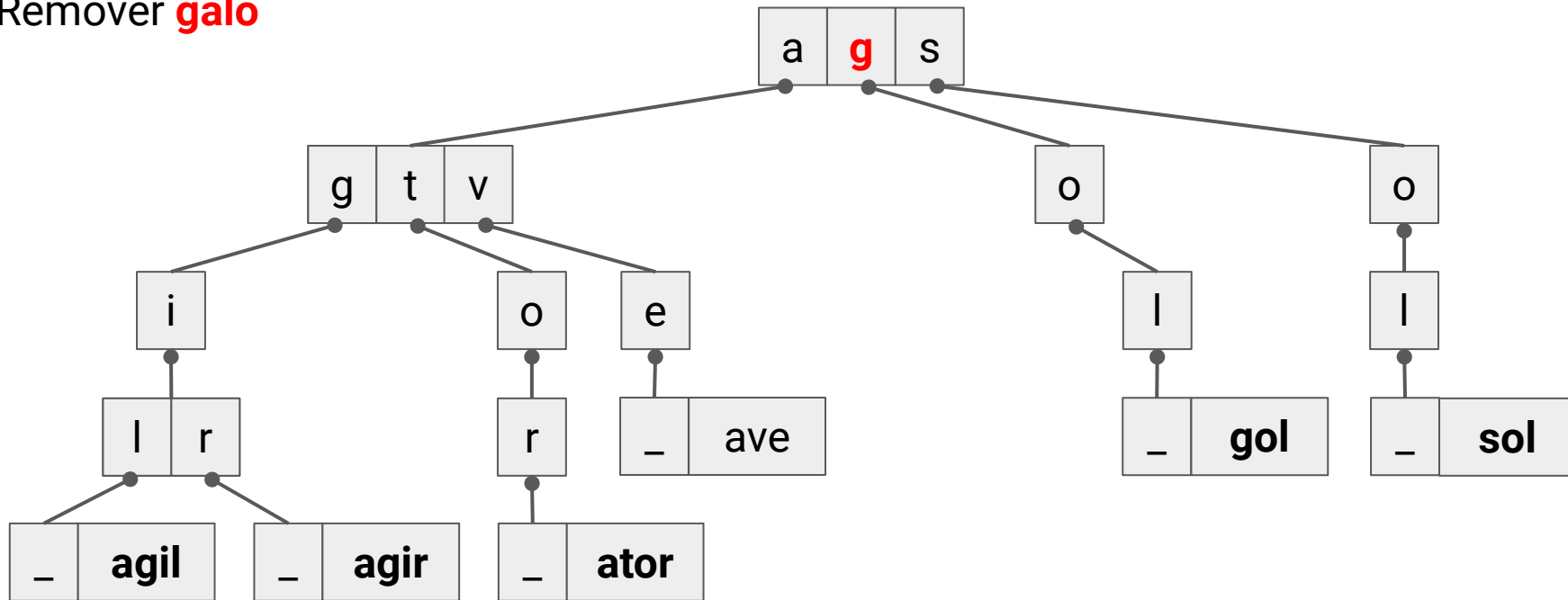
Árvores Trie

Remover galo



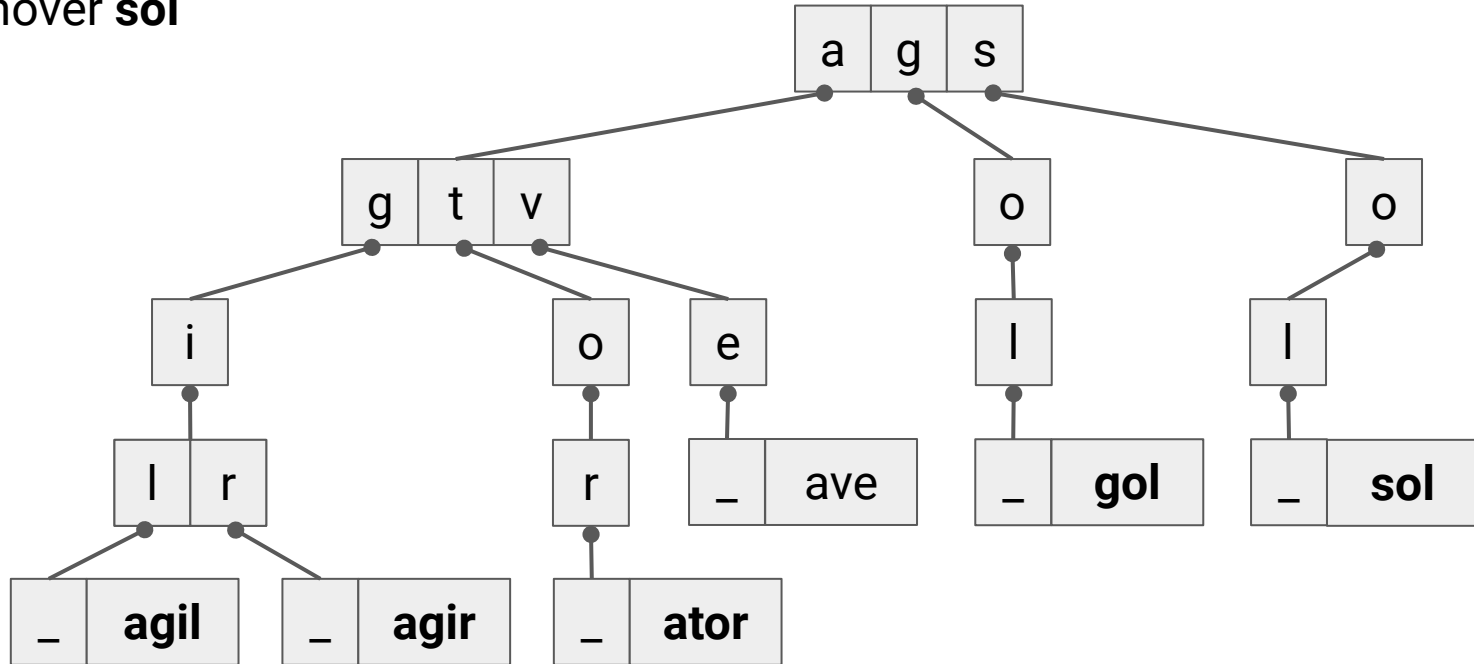
Árvores Trie

Remover galo



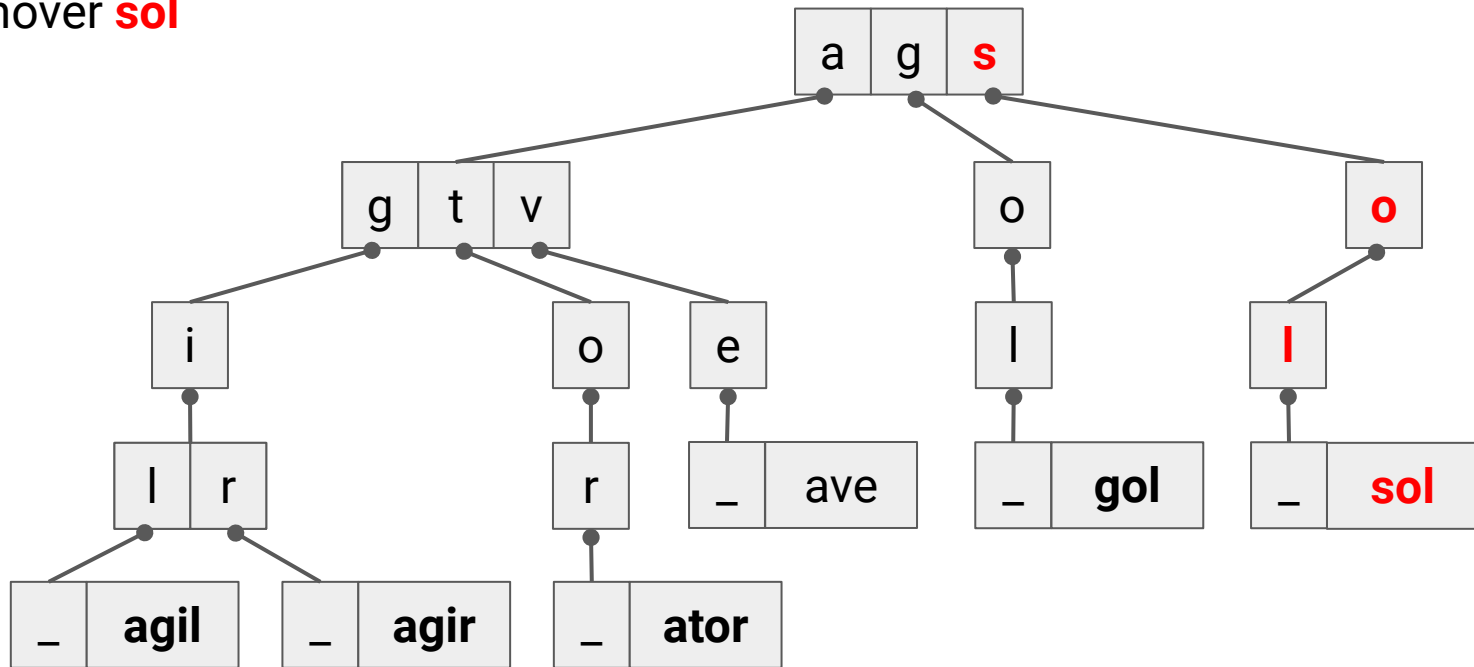
Árvores Trie

Remover **sol**



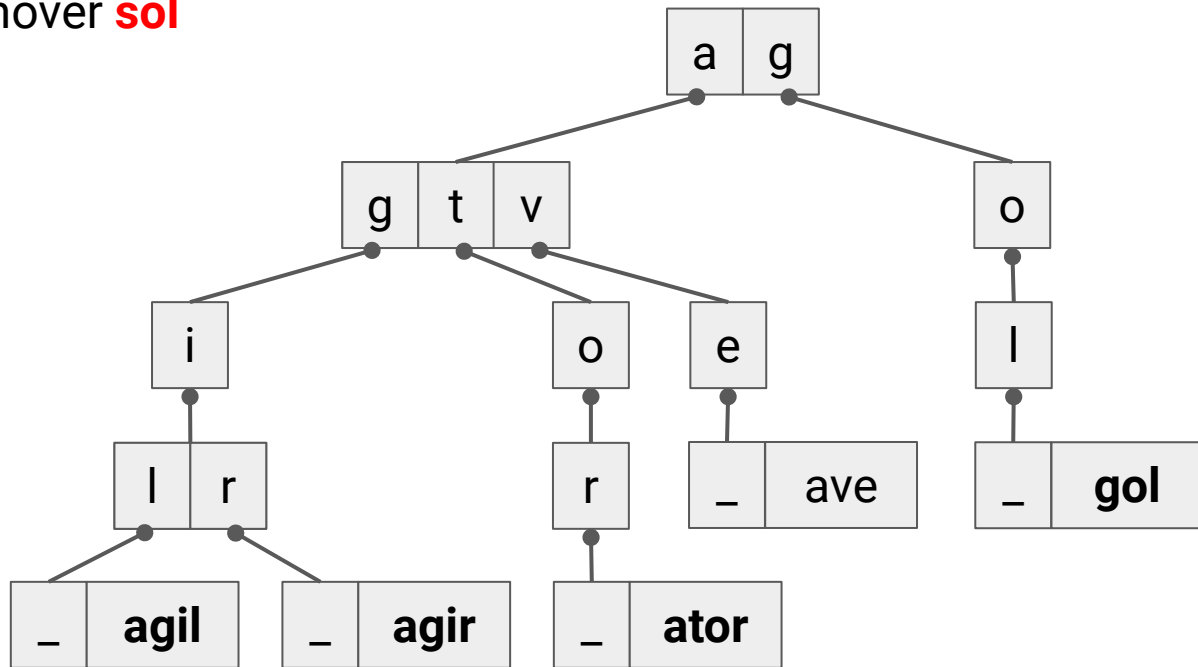
Árvores Trie

Remover **sol**



Árvores Trie

Remover **sol**



Árvores Trie

Projeto de arquivos
Prof. Allan Rodrigo Leite