Expressões regulares

Rodrigo Gabriel Ferreira Soares

DEINFO - UFRPE

2014

Representações finitas de linguagens

- Linguagens finitas podem ser descritas pela enumeração exaustiva de suas cadeias.

Representações finitas de linguagens

- Linguagens finitas podem ser descritas pela enumeração exaustiva de suas cadeias.
- Linguagens infinitas são um desafio.

Representações finitas de linguagens

- Linguagens finitas podem ser descritas pela enumeração exaustiva de suas cadeias.
- Linguagens infinitas são um desafio.
- Como o conjunto Σ^* de cadeias sobre o alfabeto Σ é contavelmente infinito, o número de possíveis representações de linguagens será também contavelmente infinito.

Representações finitas de linguagens

- Linguagens finitas podem ser descritas pela enumeração exaustiva de suas cadeias.
- Linguagens infinitas são um desafio.
- Como o conjunto Σ^* de cadeias sobre o alfabeto Σ é contavelmente infinito, o número de possíveis representações de linguagens será também contavelmente infinito.
- Por outro lado, o conjunto de todas as possíveis linguagens sobre Σ, ou seja 2^{Σ^*}, é incontavelmente infinito, uma vez que o conjunto das partes de qualquer conjunto contavelmente **NÃO É** contavelmente infinito.

Representações finitas de linguagens

- Independentemente do poder dos métodos de representação de linguagens, seremos capazes apenas de representar um conjunto contável delas, caso as representações sejam finitas.

Representações finitas de linguagens

- Independentemente do poder dos métodos de representação de linguagens, seremos capazes apenas de representar um conjunto contável delas, caso as representações sejam finitas.
- Apresentaremos diversas maneiras de se representar linguagens, cada uma mais poderosa que a anterior.

Representações finitas de linguagens

Exemplo

- Seja $L = \{w \in \{0,1\}^* : w$ tem duas ou tres ocorrencias de 1, sendo que a primeira e a segunda nao sao consecutivas$\}$, podemos representar L da seguinte forma:

$$\{0\}^* \circ \{1\} \circ \{0\}^* \circ \{0\} \circ \{1\} \circ \{0\}^* \circ ((\{1\} \circ \{0\}^*) \cup \emptyset^*)$$

- Podemos simplificar essa representação com uma nova notação

$$L = 0^*10^*010^*(10^* \cup \emptyset^*)$$

Representações finitas de linguagens

Exemplo. Uma expressão regular descreve uma linguagem por meio de símbolos isolados e $\emptyset$, combinados, não necessariamente através dos símbolos $\cup$ e $*$, possivelmente agrupados com parênteses.

Representações finitas de linguagens

Exemplo. Uma expressão regular descreve uma linguagem por meio de símbolos isolados e $\emptyset$, combinados, não necessariamente através dos símbolos $\cup$ e $*$, possivelmente agrupados com parênteses. Expressões regulares sobre Σ^* são cadeias sobre o alfabeto $\Sigma \cup \{(,), \emptyset, \cup, *\}$ que podem ser construídas através das seguintes regras

Representações finitas de linguagens

Exemplo. Uma expressão regular descreve uma linguagem por meio de símbolos isolados e $\emptyset$, combinados, não necessariamente através dos símbolos $\cup$ e $*$, possivelmente agrupados com parênteses. Expressões regulares sobre Σ^* são cadeias sobre o alfabeto $\Sigma \cup \{(,), \emptyset, \cup, *\}$ que podem ser construídas através das seguintes regras

1. $\emptyset$ e cada membro de Σ são uma expressão regular

Representações finitas de linguagens

Exemplo. Uma expressão regular descreve uma linguagem por meio de símbolos isolados e $\emptyset$, combinados, não necessariamente através dos símbolos $\cup$ e $*$, possivelmente agrupados com parênteses. Expressões regulares sobre Σ^* são cadeias sobre o alfabeto $\Sigma \cup \{(,),\emptyset,\cup,*\}$ que podem ser construídas através das seguintes regras

1. $\emptyset$ e cada membro de Σ são uma expressão regular
2. Se α e β são expressões regulares, então, $(\alpha\beta)$ será também uma expressão regular.

Representações finitas de linguagens

Exemplo. Uma expressão regular descreve uma linguagem por meio de símbolos isolados e $\emptyset$, combinados, não necessariamente através dos símbolos $\cup$ e $*$, possivelmente agrupados com parênteses. Expressões regulares sobre Σ^* são cadeias sobre o alfabeto $\Sigma \cup \{(,), \emptyset, \cup, *\}$ que podem ser construídas através das seguintes regras

1. $\emptyset$ e cada membro de Σ são uma expressão regular
2. Se α e β são expressões regulares, então, $(\alpha\beta)$ será também uma expressão regular.
3. Se α e β são expressões regulares, então, $(\alpha \cup \beta)$ será também uma expressão regular.

Representações finitas de linguagens

Exemplo. Uma expressão regular descreve uma linguagem por meio de símbolos isolados e $\emptyset$, combinados, não necessariamente através dos símbolos $\cup$ e $*$, possivelmente agrupados com parênteses.
Expressões regulares sobre Σ^* são cadeias sobre o alfabeto $\Sigma \cup \{(,), \emptyset, \cup, *\}$ que podem ser construídas através das seguintes regras

1. $\emptyset$ e cada membro de Σ são uma expressão regular
2. Se α e β são expressões regulares, então, $(\alpha\beta)$ será também uma expressão regular.
3. Se α e β são expressões regulares, então, $(\alpha \cup \beta)$ será também uma expressão regular.
4. Se α é uma expressão regular, então, α^* será também uma expressão regular.

Representações finitas de linguagens

Exemplo. Uma expressão regular descreve uma linguagem por meio de símbolos isolados e $\emptyset$, combinados, não necessariamente através dos símbolos $\cup$ e $*$, possivelmente agrupados com parênteses. Expressões regulares sobre Σ^* são cadeias sobre o alfabeto $\Sigma \cup \{(,), \emptyset, \cup, *\}$ que podem ser construídas através das seguintes regras

1. $\emptyset$ e cada membro de Σ são uma expressão regular
2. Se α e β são expressões regulares, então, $(\alpha\beta)$ será também uma expressão regular.
3. Se α e β são expressões regulares, então, $(\alpha \cup \beta)$ será também uma expressão regular.
4. Se α é uma expressão regular, então, α^* será também uma expressão regular.
5. Nada será expressão regular, a menos que possa ser produzido a partir das regras anteriores.

Representações finitas de linguagens

- Toda expressão regular representa uma linguagem.

Representações finitas de linguagens

- Toda expressão regular representa uma linguagem.
- Formalmente, a função $\mathcal{L}$ representa a relação entre expressões regulares e linguagens. Se α é uma expressão regular, então $\mathcal{L}(\alpha)$ é a linguagem representada por α.

Representações finitas de linguagens

- Toda expressão regular representa uma linguagem.
- Formalmente, a função $\mathcal{L}$ representa a relação entre expressões regulares e linguagens. Se α é uma expressão regular, então $\mathcal{L}(\alpha)$ é a linguagem representada por α.
  1. $\mathcal{L}(\emptyset) = \emptyset$ e $\mathcal{L}(a) = \{a\}$ para cada $a \in \Sigma$

Representações finitas de linguagens

- Toda expressão regular representa uma linguagem.
- Formalmente, a função $\mathcal{L}$ representa a relação entre expressões regulares e linguagens. Se α é uma expressão regular, então $\mathcal{L}(\alpha)$ é a linguagem representada por α.
  1. $\mathcal{L}(\emptyset) = \emptyset$ e $\mathcal{L}(a) = \{a\}$ para cada $a \in \Sigma$
  2. Se α e β são expressões regulares, então $\mathcal{L}((\alpha\beta)) = \mathcal{L}(\alpha)\mathcal{L}(\beta)$

Representações finitas de linguagens

- Toda expressão regular representa uma linguagem.
- Formalmente, a função $\mathcal{L}$ representa a relação entre expressões regulares e linguagens. Se α é uma expressão regular, então $\mathcal{L}(\alpha)$ é a linguagem representada por α.
    1. $\mathcal{L}(\emptyset) = \emptyset$ e $\mathcal{L}(a) = \{a\}$ para cada $a \in \Sigma$
    2. Se α e β são expressões regulares, então $\mathcal{L}((\alpha\beta)) = \mathcal{L}(\alpha)\mathcal{L}(\beta)$
    3. Se α e β são expressões regulares, então $\mathcal{L}((\alpha \cup \beta)) = \mathcal{L}(\alpha) \cup \mathcal{L}(\beta)$

Representações finitas de linguagens

- Toda expressão regular representa uma linguagem.
- Formalmente, a função $\mathcal{L}$ representa a relação entre expressões regulares e linguagens. Se α é uma expressão regular, então $\mathcal{L}(\alpha)$ é a linguagem representada por α.
  1. $\mathcal{L}(\emptyset) = \emptyset$ e $\mathcal{L}(a) = \{a\}$ para cada $a \in \Sigma$
  2. Se α e β são expressões regulares, então $\mathcal{L}((\alpha\beta)) = \mathcal{L}(\alpha)\mathcal{L}(\beta)$
  3. Se α e β são expressões regulares, então $\mathcal{L}((\alpha \cup \beta)) = \mathcal{L}(\alpha) \cup \mathcal{L}(\beta)$
  4. Se α é expressão regular, então $\mathcal{L}((\alpha^*)) = \mathcal{L}(\alpha)^*$

Representações finitas de linguagens

- Toda expressão regular representa uma linguagem.
- Formalmente, a função $\mathcal{L}$ representa a relação entre expressões regulares e linguagens. Se α é uma expressão regular, então $\mathcal{L}(\alpha)$ é a linguagem representada por α.
  1. $\mathcal{L}(\emptyset) = \emptyset$ e $\mathcal{L}(a) = \{a\}$ para cada $a \in \Sigma$
  2. Se α e β são expressões regulares, então $\mathcal{L}((\alpha\beta)) = \mathcal{L}(\alpha)\mathcal{L}(\beta)$
  3. Se α e β são expressões regulares, então $\mathcal{L}((\alpha \cup \beta)) = \mathcal{L}(\alpha) \cup \mathcal{L}(\beta)$
  4. Se α é expressão regular, então $\mathcal{L}((\alpha^*)) = \mathcal{L}(\alpha)^*$
- Toda expressão regular é associada a uma linguagem.

Representações finitas de linguagens

- Exemplo 1. O que significa $\mathcal{L}(((a \cup b)^* a))$?

Representações finitas de linguagens

- Exemplo 1. O que significa $\mathcal{L}(((a \cup b)^* a))$?
- Exemplo 2. Qual a linguagem representada por $(c^*(a \cup (bc^*))^*)$?

Representações finitas de linguagens

- Exemplo 1. O que significa $\mathcal{L}(((a \cup b)^* a))$?
- Exemplo 2. Qual a linguagem representada por $(c^*(a \cup (bc^*))^*)$?
- Toda linguagem que pode ser representada por uma expressão regular pode ser representada por infinitas outras. Exemplo α e $(\alpha \cup \epsilon)$, e $((\alpha \cup \beta) \cup \gamma)$ e $(\alpha \cup (\beta \cup \gamma))$

Representações finitas de linguagens

- Exemplo 1. O que significa $\mathcal{L}(((a \cup b)^* a))$?
- Exemplo 2. Qual a linguagem representada por $(c^*(a \cup (bc^*))^*)$?
- Toda linguagem que pode ser representada por uma expressão regular pode ser representada por infinitas outras. Exemplo α e $(\alpha \cup \epsilon)$, e $((\alpha \cup \beta) \cup \gamma)$ e $(\alpha \cup (\beta \cup \gamma))$
- Podemos agora dizer que a expressão regular $a^* b^*$ é o conjunto de todas as cadeias com um certo número de a's seguidos de um certo número de b's.

Representações finitas de linguagens

- A classe das **linguagens regulares** sobre um alfabeto Σ são todas as linguagens L tais que $L = \mathcal{L}(\alpha)$ para alguma expressão regular α sobre Σ.

Representações finitas de linguagens

- A classe das **linguagens regulares** sobre um alfabeto Σ são todas as linguagens L tais que $L = \mathcal{L}(\alpha)$ para alguma expressão regular α sobre Σ.
- Todas as linguagens regulares podem ser descritas por ERs.

Representações finitas de linguagens

- A classe das **linguagens regulares** sobre um alfabeto Σ são todas as linguagens L tais que $L = \mathcal{L}(\alpha)$ para alguma expressão regular α sobre Σ.
- Todas as linguagens regulares podem ser descritas por ERs.
- A classe das **linguagens regulares** sobre um alfabeto Σ é o fechamento do conjunto de linguagens $\{\{\sigma\} : \sigma \in \Sigma\} \cup \{\emptyset\}$ sob a união, concatenação e estrela de Kleene.

Representações finitas de linguagens

- A classe das **linguagens regulares** sobre um alfabeto Σ são todas as linguagens L tais que $L = \mathcal{L}(\alpha)$ para alguma expressão regular α sobre Σ.
- Todas as linguagens regulares podem ser descritas por ERs.
- A classe das **linguagens regulares** sobre um alfabeto Σ é o fechamento do conjunto de linguagens $\{\{\sigma\} : \sigma \in \Sigma\} \cup \{\emptyset\}$ sob a união, concatenação e estrela de Kleene.
- Não podemos descrever todas as linguagens com ER's. Por exemplo, $\{0^n, 1^n : n \geq 0\}$ não é regular.

Representações finitas de linguagens

- ERs são insuficientes. Podemos voltar para $L = \{w \in \Sigma^* : w \text{ goza da propriedade } P\}$.

Representações finitas de linguagens

- ERs são insuficientes. Podemos voltar para $L = \{w \in \Sigma^* : w \text{ goza da propriedade } P\}$.
- Por enquanto, P é uma propriedade algorítmica. Deve haver um algoritmo para decidir se uma cadeia pertence a uma linguagem.

Representações finitas de linguagens

- ERs são insuficientes. Podemos voltar para $L = \{w \in \Sigma^* : w \text{ goza da propriedade } P\}$.
- Por enquanto, P é uma propriedade algorítmica. Deve haver um algoritmo para decidir se uma cadeia pertence a uma linguagem.
- Para responder 'A cadeia w pertence a L?' temos o **dispositivo reconhecedor dessa linguagem**.

Representações finitas de linguagens

Exemplo $L = \{w \in \{0,1\}* : wnotem111comosubcadeia\}$. O reconhecedor de L que lê um símbolo por vez da esquerda da direita pode ter os seguintes passos:

Representações finitas de linguagens

Exemplo $L = \{w \in \{0,1\}* : wnotem111comosubcadeia\}$. O reconhecedor de L que lê um símbolo por vez da esquerda da direita pode ter os seguintes passos:

1. Mantenha um contador que começa em zero e é zerado sempre que encontrar 0.

Representações finitas de linguagens

Exemplo $L = \{w \in \{0,1\}* : wnotem111comosubcadeia\}$. O reconhecedor de L que lê um símbolo por vez da esquerda da direita pode ter os seguintes passos:

1. Mantenha um contador que começa em zero e é zerado sempre que encontrar 0.
2. Incremente o contador se encontrar um 1.

Representações finitas de linguagens

Exemplo $L = \{w \in \{0,1\}* : wnotem111comosubcadeia\}$. O reconhecedor de L que lê um símbolo por vez da esquerda da direita pode ter os seguintes passos:

1. Mantenha um contador que começa em zero e é zerado sempre que encontrar 0.
2. Incremente o contador se encontrar um 1.
3. Pare em resposta NÃO se o contador for 3.

Representações finitas de linguagens

Exemplo $L = \{w \in \{0,1\}* : wnotem111comosubcadeia\}$. O reconhecedor de L que lê um símbolo por vez da esquerda da direita pode ter os seguintes passos:

1. Mantenha um contador que começa em zero e é zerado sempre que encontrar 0.
2. Incremente o contador se encontrar um 1.
3. Pare em resposta NÃO se o contador for 3.
4. Pare em resposta SIM caso a cadeia termine e o contador for 3.

Representações finitas de linguagens

- Um método alternativo é definir como cada membro da linguagem é gerado. Uma ER, como $L = (\epsilon \cup b \cup bb)(a \cup ab \cup abb)^*$, pode ser entendida como uma forma de se produzir elementos de uma linguagem.

Representações finitas de linguagens

- Um método alternativo é definir como cada membro da linguagem é gerado. Uma ER, como $L = (\epsilon \cup b \cup bb)(a \cup ab \cup abb)^*$, pode ser entendida como uma forma de se produzir elementos de uma linguagem.
- Para gerar um elemento de L
  1. Escreva nada, ou b, ou bb

Representações finitas de linguagens

- Um método alternativo é definir como cada membro da linguagem é gerado. Uma ER, como $L = (\epsilon \cup b \cup bb)(a \cup ab \cup abb)^*$, pode ser entendida como uma forma de se produzir elementos de uma linguagem.
- Para gerar um elemento de L
  1. Escreva nada, ou b, ou bb
  2. Escreva a, ou ab, ou abb e faça isso zero ou mais vezes

Representações finitas de linguagens

- Um método alternativo é definir como cada membro da linguagem é gerado. Uma ER, como $L = (\epsilon \cup b \cup bb)(a \cup ab \cup abb)^*$, pode ser entendida como uma forma de se produzir elementos de uma linguagem.
- Para gerar um elemento de L
  1. Escreva nada, ou b, ou bb
  2. Escreva a, ou ab, ou abb e faça isso zero ou mais vezes
- todos os elementos de L e somente eles podem ser gerados dessa forma.

Representações finitas de linguagens

- Um método alternativo é definir como cada membro da linguagem é gerado. Uma ER, como $L = (\epsilon \cup b \cup bb)(a \cup ab \cup abb)^*$, pode ser entendida como uma forma de se produzir elementos de uma linguagem.
- Para gerar um elemento de L
  1. Escreva nada, ou b, ou bb
  2. Escreva a, ou ab, ou abb e faça isso zero ou mais vezes
- todos os elementos de L e somente eles podem ser gerados dessa forma.
- Tais **geradores de linguagens** não são algoritmos, pois não são explícitos sobre o que fazer.