Linguagens

O conceito de linguagem engloba uma variedade de categorias distintas de linguagens: linguagens naturais, linguagens de programação, linguagens matemáticas, etc. Uma definição geral de linguagem deve incluir todos estes tipos de linguagens.

Uma linguagem, no seu sentido mais amplo, pode ser definida como um conjunto de palavras formadas a partir de determinado alfabeto. Esta é a definicção mas abrangete que pode haver do que seja uma linguagem, uma vez que não há restrições na forma das palavras que constituirão a linguagem.

No entanto, linguagens mais interessantes não são constituídas de palavras arbitrárias, mas de palavras que satisfazem um certo conjunto de propriedades. Estas propriedades definem a *sintaxe* da linguagem.

Alfabetos, palavras e linguagens

> Um **alfabeto** é um conjunto de símbolos indivisíveis de qualquer natureza. Um alfabeto é geralmente denotado pela letra grega Σ.

O alfabeto de uma linguagem natural, como Português ou Inglês, é o conjunto de palavras que fazem parte da linguagem. As palavras da linguagem são então os símbolos de seu alfabeto, e são indivisíveis. A palavra *camaleão*, por exemplo, não pode ser dividida em *cama* e *leão*, uma vez que estas são outras palavras da mesma linguagem.

> Uma **palavra** sobre um alfabeto é uma seqüência de símbolos deste alfabeto.

Uma seqüência de palavras em português forma o que chamamos de sentença. As sentenças que podem ser formadas a partir das palavras que pertencem ao Português são consideradas então palavras no sentido definido acima. O alfabeto de uma linguagem de programação é composto das palavras reservadas, variáveis e símbolos da linguagem. Um trecho de código da linguagem é considerado, então, uma palavra da linguagem.

Apesar da possível confusão gerada pelos termos alfabeto e palavra, que são reconhecidos ordinariamente por nós de forma diferente (alfabeto é o conjunto {a, b, ..., z} enquanto uma palavra é formada por uma seqüência de símbolos deste alfabeto), note que a definição que conhecemos é perfeitamente compatível com a noção dada. Obviamente que nem toda

seqüência de símbolos do alfabeto acima constitui uma palavra da Língua Portuguesa. Aliás, o conjunto de todas as palavras válidas da Língua Portuguesa é um conjunto finito e pode ser descrito como tal (como aliás está no dicionário).

Uma palavra foi definida como uma seqüência de símbolos de um alfabeto. Para podermos definir propriedades das palavras, definimos o conjunto de palavras sobre um alfabeto recursivamente. A base de recursão consiste da palavra que não contém nenhum símbolo, chamada **palavra vazia** e denotada por ε.

Seja Σ um alfabeto. Σ^*, o conjunto de todas as palavras formadas a partir de Σ, é definido como:

i. $\varepsilon \in \Sigma^*$
ii. Se $w \in \Sigma^*$ e $a \in \Sigma$, então $wa \in \Sigma^*$.
iii. $w \in \Sigma^*$ somente se pode ser obtido a partir de ε através da aplicação das regras acima um número finito de vezes.

Para todo alfabeto não vazio Σ, Σ^* contém um número infinito de elementos. Se $\Sigma=\{a\}$, Σ^* contém as palavras ε, a, aa, aaa, ...

O comprimento de uma palavra w é o número de símbolos do alfabeto que a palavra contém, ou mais formalmente, o número de aplicações do passo de recursão da definição formal necessários para construir a palavra a partir dos símbolos do alfabeto. O comprimento de uma palavra w é denotado por $|w|$. Se Σ contém n elementos, então existem n^k palavras de comprimento k em Σ^*.

Uma linguagem é composta de palavras formadas sobre um alfabeto. Usualmente, algumas restrições são colocadas nas palavras para que elas façam parte de uma linguagem. A Língua Portuguesa, por exemplo, é formada de seqüências de palavras que pertencem à língua, mas nem toda seqüência de palavras válidas é uma sentença válida em Português, somente aquelas que satisfazem certas condições a respeito da ordem e do tipo de palavras que a constituem. Conseqüentemente, uma linguagem é um subconjunto de todas as palavras possíveis de serem formadas a partir de um alfabeto.

Uma **linguagem** sobre um alfabeto Σ é um subconjunto de Σ^*.

Uma vez que palavras são os elementos de uma linguagem, devemos examinar as propriedades das palavras e as operações que podemos aplicar sobre elas. A concatenação é a principal operação sobre linguagens. A operação de concatenação é uma operação binária sobre palavras de uma linguagem e gera uma terceira palavra a partir das duas palavras iniciais. Podemos definir a operação de concatenação de maneira forma da seguinte forma:

Seja u, v $\in \Sigma^*$. A concatenção de u e v, denotada por uv é uma operação binária em Σ^* definida como se segue:

i. se |v|=0 então v=ε e uv = u.
ii. se |v|=n>0 então v = wa para alguma palavra de comprimento n-1 e a $\in \Sigma$, e uv=(uw)a.

A concatenação de duas palavras é, segundo a definição acima, a justaposição das duas palavras de maneira a formar uma terceira.

Subpalavras podem ser definidas usando a operação de concatenação. Intuitivamente, uma palavra u é uma subpalavra de uma palavra v se ele ocorre "dentro" de v. Formalmente, u é uma **subpalavra** de de v se existem subpalavras x e y tais que v = xuy. Um **prefixo** de v é uma subpalavra u na qual x é a palavra vazia na decomposição de v. Isto é, v=uy. Similarmente, u é um **sufixo** de v se v=xu.

O **reverso** de uma palavra é a palavra escrita de trás para diante. Da mesma forma que a concatenação, o reverso pode ser definido recursivamente no comprimento da palavra. A remoção de um elemento do lado direito da palavra gera uma palavra menor que pode ser usada no passo recursivo da definição.

O reverso da palavra u $\in \Sigma^*$, denotada por u^R, é definido como se segue:

i. se |u|=0 então u=ε e $\varepsilon^R=\varepsilon$.
ii. se |u|=n>0, então u=wa para alguma palavra w $\in \Sigma^*$, com |w|=n-1, e algum símbolo a $\in \Sigma$, e $u^R=aw^R$.

Operações sobre Linguagens

Como uma linguagem é definida como sendo simplesmente um conjunto de palavras formadas sobre símbolos de um alfabeto, todas as operações sobre conjuntos podem ser aplicadas sobre linguagens. Em particular, pode-se aplicar sobre linguagens as operações de união, intersecção, diferença e complemento. O significado destas operações sobre linguagens é bastante intuitiva:

A **união** de duas linguagens L1 e L2 é a linguagem L, que contém todas as palavras que pertencem a qualquer uma das linguagens. Definindo formalmente:

$$L = L1 \cup L2 = \{w \in \Sigma^* \mid w \in L1 \text{ ou } w \in L2\}$$

A **intersecção** de duas linguagens L1 e L2 é a linguagem L, que contém todas as palavras que pertencem simultaneamente às duas linguagens. Definindo formalmente:

$$L = L1 \cap L2 = \{w \in \Sigma^* \mid w \in L1 \text{ e } w \in L2\}$$

A **diferença** de duas linguagens L1 e L2 é a linguagem L, que contém todas as palavras que pertencem à linguagem L1 mas não pertencem à linguagem L2. Definindo formalmente:

$$L = L1 - L2 = \{w \in \Sigma^* \mid w \in L1 \text{ e } w \notin L2\}$$

O **complemento** de uma linguagem L é a linguagem que contém todas as palavras que não pertencem a L. Uma vez que Σ^* é o conjunto de todas as palavras que podem ser formadas sobre um alfabeto Σ, então o complemento da linguagem L, construída sobre Σ é a linguagem Σ^*-L, ou ainda, definindo formalmente:

$$\Sigma^* - L = \{ w \in \Sigma^* \mid w \notin L\}$$

Concatenação

A concatenação de duas linguagens X e Y, denotada por X ° Y, ou simplesmente XY, é definida como

$$XY = \{xy \mid x \in X \text{ e } y \in Y\}$$

A concatenação da linguagem X com ela mesma n vezes é denotada X^n. X^0 é a linguagem $\{\varepsilon\}$.

Operação de Kleene

A operação de Kleene é uma operação unária, da qual resultam as palavras formadas a partir da concatenação sucessiva das palavras da linguagem à qual a operação é aplicada. A linguagem L*, resultante da aplicação da operação de Kleene sobre a linguagem L, é definida como:

$$L^* = \{w \mid w=w_1w_2...w_n,\ n \geq 0,\ w_1,w_2,...,w_n \in L\}$$

Note-se que o uso de Σ^* para denotar o conjunto de todas as palavras de Σ é consistente com a notação de Kleene se considerarmos Σ como uma linguagem finita. Isto é, se fizermos $L = \Sigma$ e aplicarmos a definição de operação de Kleene sobre uma linguagem a Σ, então Σ^* será o conjunto de todas as palavras que podem ser formadas como concatenação das palavras de Σ. Mas as palavras de Σ, se visto como uma linguagem, são os símbolos individuais do alfabeto Σ. Logo, como definido originalmente, Σ^* é o conjunto de todas as palavras que podem ser formadas a partir dos símbolos de Σ.

Representação Finita de Linguagens

Conforme definido anteriormente, uma linguagem é simplesmente um conjunto de palavras constituídas de símbolos de um determinado alfabeto. Colocando de maneira mais formal, uma linguagem é simplesmente um subconjunto de Σ^*. Assim, Σ^*, Σ e $\varnothing$ são linguagens.

Como uma linguagem é um conjunto, pode-se especificar uma linguagem finita listando todas as palavras que fazem parte da linguagem. No entanto, as linguagens de maior interesse são, na sua maioria, infinitas, de modo que é impossível listar todas as suas palavras.

Uma linguagem de programação, por exemplo, se considerarmos as suas palavras como programas sintaticamente válidos, é claramente infinita, uma vez que a partir da especificação de uma linguagem de programação podemos construir um número infinito de programas distintos sintaticamente válidos.

Assim, precisamos de uma maneira de especificar linguagens infinitas de maneira concisa. Como uma linguagem é um conjunto, podemos especificá-las da mesma maneira que especificamos um conjunto:

$$\{w \in \Sigma^* \mid w \text{ tem a propriedade P}\}$$

Esta é uma possível maneira de especificar linguagens infinitas. Afinal, especificamos conjuntos infinitos muito bem através desta notação. Por exemplo, o conjunto dos número pares (que é infinito) pode ser especificado como:

$$\text{Pares} = \{x \in N \mid x \bmod 2 = 0\}$$

Linguagens infinitas podem, então ser especificadas da mesma forma. A linguagem sobre o alfabeto $\Sigma=\{0,1\}^*$, cujas palavras são aquelas que têm um número diferente de 0s e 1s pode ser especificada como:

$$L = \{w \in \{0,1\}^* \mid w \text{ tem um número diferente de 0s e 1s}\}$$

Neste caso, a especificação da linguagem não é ambígua, uma vez que se trata de uma linguagem bastante simples. Linguagens mais complexas, no entanto, podem ser um pouco mais complicadas de expressar desta maneira. Por exemplo, utilizando esta notação, como poderíamos especificar uma palavra (programa sintaticamente correto) da linguagem de programação Pascal?

Outro problema que se apresenta na utilização desta notação é a aplicação de operações entre linguagens. A união do conjunto dos números pares com o conjunto dos números ímpares é o conjunto dos números naturais. Mas qual o resultado da união de duas linguagens?

Dada a linguagem apresentada acima $L = \{w \in \{0,1\}^* \mid w \text{ tem um número diferente de 0s e 1s}\}$, qual seria a linguagem L^*?

Pode-se mostrar que a linguagem $L^*=\{0,1\}^*$, embora este resultado não seja intuitivo da forma de representação da linguagem.

A definição de uma linguagem necessita de uma forma de representação que não seja ambígua. Ou seja, todas as palavras da linguagem devem ser claramente especificadas através desta representação. Além disso, a aplicação de operações sobre linguagens também deve, na medida do possível, não ser complicada demais.

Em suma, o que procuramos é uma linguagem, que possa ser representada de maneira finita, para especificarmos outras linguagens.

Só que aqui já surge um problema: como uma linguagem é um subconjunto qualquer de Σ^* (onde Σ é o alfabeto sobre o qual a linguagem foi construída), então as linguagens que podem ser construídas a partir de um alfabeto Σ são todos os subconjuntos de Σ^*.

Um alfabeto Σ é um conjunto finito, logo o conjunto de palavras que podem ser construídas sobre Σ, Σ^*, é infinitamente contável. Por outro lado, o conjunto de todos os subconjuntos de Σ^* (o conjunto das partes 2^{Σ^*} de Σ^*) é incontável, uma vez que o conjunto das partes de qualquer conjunto infinitamente contável é incontável.

Logo, podemos concluir que temos um número incontável de linguagens distintas para representar, e somente um número contável de especificações finitas para representá-las. Este é um resultado significativo da Teoria da Computação: somente podemos respresentar de maneira finita um subconjunto das linguagens possíveis sobre um alfabeto. Felizmente, grande parte das linguagens nas quais temos interesse podem ser representadas finitamente.

A partir deste ponto serão vistos métodos de representação finita de linguagens, em particular, expressões regulares, que representam linguagens regulares e gramáticas livres de contexto, que representam linguagens livres de contexto.
