

Representação de Conhecimento: Lógica de Predicados

Representando Conhecimento Mais Complexo Usando Lógica de Predicados e Quantificadores

Márcio Nicolau

2025-09-18

Table of contents

Introdução	2
Objetivo de Aprendizagem	2
Revisão Rápida: Limitações da Lógica Proposicional	2
Lógica de Predicados de Primeira Ordem (LPO)	3
Sintaxe da Lógica de Predicados	3
Termos	3
Predicados (Símbolos de Predicado)	3
Conectivos Lógicos	3
Quantificadores	4
Fórmulas Bem Formadas (FBFs) em LPO	4
Exemplos de Sentenças em LPO	4
Semântica da Lógica de Predicados	4
Modelo	6
Inferência em Lógica de Predicados	6
Instanciação Universal (Universal Instantiation - UI)	6
Instanciação Existencial (Existential Instantiation - EI)	6
Modus Ponens Generalizado	6
Unificação (Unification)	7
Resolução em LPO	7
Exemplo de Inferência em LPO (Conceitual)	7
Comparação entre Lógica Proposicional e Lógica de Predicados	8
LPO em Python (Bibliotecas)	8
Considerações Finais	12
Verificação de Aprendizagem	12

List of Figures

1	Estrutura Sintática da Lógica de Predicados de Primeira Ordem (LPO)	5
2	Comparativo entre Lógica Proposicional e Lógica de Predicados de Primeira Ordem	9

Introdução

Na aula anterior, exploramos a Lógica Proposicional (LP) como uma ferramenta fundamental para a Representação de Conhecimento (RC). Embora a LP forneça uma base sólida para o raciocínio dedutivo, vimos que suas limitações em expressividade – especialmente a incapacidade de lidar com objetos, suas propriedades, relações e quantificadores (“todos”, “alguns”) – a tornam inadequada para representar a complexidade do mundo real em muitos domínios da Inteligência Artificial.

Nesta aula, avançaremos para a **Lógica de Predicados de Primeira Ordem (LPO)**, frequentemente chamada apenas de Lógica de Predicados. Este formalismo mais poderoso nos permitirá representar conhecimento de uma forma muito mais rica e flexível, introduzindo conceitos como predicados, funções, variáveis e quantificadores, elementos essenciais para modelar o mundo de forma mais granular e expressiva.

Objetivo de Aprendizagem

Ao final desta aula, você será capaz de:

- Explicar as diferenças e vantagens da Lógica de Predicados em relação à Lógica Proposicional.
- Compreender os componentes sintáticos da Lógica de Predicados: termos, predicados, funções, variáveis e quantificadores.
- Construir Fórmulas Bem Formadas (FBFs) em Lógica de Predicados para representar sentenças complexas do mundo real.
- Interpretar a semântica de sentenças em Lógica de Predicados.
- Entender o conceito de unificação e seu papel na inferência em Lógica de Predicados.
- Identificar cenários onde a Lógica de Predicados é mais apropriada para a RC.

Revisão Rápida: Limitações da Lógica Proposicional

Conforme discutido, a Lógica Proposicional representa fatos como unidades atômicas (P, Q, *Chuva*). Ela não pode expressar:

- **Propriedades de objetos:** Não consigo dizer *Cor(Céu, Azul)* ou *Idade(João, 30)*.
- **Relações entre objetos:** Não consigo dizer *Pai(João, Pedro)* ou *Gosta(Maria, Pizza)*.
- **Quantificação:** Não consigo expressar “Todos os pássaros voam” ou “Existe um número primo”.
 - Para “Todos os humanos são mortais”, teríamos que ter *Humano1_Mortal*, *Humano2_Mortal*, ..., o que é inviável e não permite generalização.

A Lógica de Predicados de Primeira Ordem supera essas limitações.

Lógica de Predicados de Primeira Ordem (LPO)

A LPO é um formalismo mais expressivo para representar conhecimento, permitindo uma descrição mais detalhada de objetos, suas propriedades e relações. Ela expande a Lógica Proposicional introduzindo:

- **Objetos:** Entidades no mundo (pessoas, carros, números).
- **Predicados:** Relações ou propriedades sobre objetos.
- **Funções:** Mapeiam um ou mais objetos para outro objeto.
- **Variáveis:** Representam objetos em geral.
- **Quantificadores:** Permitem expressar “para todos” (universal) e “existe” (existencial).

Sintaxe da Lógica de Predicados

A sintaxe da LPO define como construir expressões válidas.

Termos

Um **termo** refere-se a um objeto no mundo. Podem ser:

- **Constantes:** Símbolos que representam objetos específicos (e.g., João, Roma, 2, Quadrado). Geralmente iniciam com letra maiúscula.
- **Variáveis:** Símbolos que representam qualquer objeto dentro de um domínio (e.g., x , y , z). Geralmente iniciam com letra minúscula.
- **Funções:** Um símbolo de função seguido por uma lista de termos entre parênteses. A função mapeia um ou mais termos para outro termo.
 - Exemplos: `pai(João)`, `distancia(Roma, Paris)`, `soma(x, y)`.

Predicados (Símbolos de Predicado)

Um **predicado** representa uma relação entre objetos ou uma propriedade de um objeto. Um símbolo de predicado seguido por uma lista de termos entre parênteses forma uma **fórmula atômica**.

- **Exemplos:**
 - `Pai(João, Maria)`: João é pai de Maria.
 - `Gosta(Maria, Pizza)`: Maria gosta de Pizza.
 - `ÉPar(2)`: 2 é um número par.
 - `Maior(5, 3)`: 5 é maior que 3.

Conectivos Lógicos

Os mesmos conectivos da Lógica Proposicional são usados:

Símbolo	Nome	Significado
$\neg$	Negação	“não”
$\wedge$	Conjunção	“e”
$\vee$	Disjunção	“ou”

Símbolo	Nome	Significado
$\rightarrow$	Implicação (Condicional)	“se...então...”
$\leftrightarrow$	Bicondicional	“se e somente se”

Quantificadores

Essenciais para expressar sentenças sobre coleções de objetos.

- **Quantificador Universal ($\forall$):** “Para todo”, “Para cada”, “Para qualquer”.
 - **Sintaxe:** $\forall x P(x)$
 - **Significado:** Para todos os objetos x no domínio, a propriedade $P(x)$ é verdadeira.
 - *Exemplo:* $\forall x (Pssaro(x) \rightarrow Voa(x))$ (Todos os pássaros voam).
- **Quantificador Existencial ($\exists$):** “Existe um”, “Pelo menos um”, “Algum”.
 - **Sintaxe:** $\exists x P(x)$
 - **Significado:** Existe pelo menos um objeto x no domínio para o qual a propriedade $P(x)$ é verdadeira.
 - *Exemplo:* $\exists x (Homem(x) \wedge Morto(x))$ (Existe algum homem que está morto).

Fórmulas Bem Formadas (FBFs) em LPO

As regras para construir FBFs em LPO são:

1. Uma fórmula atômica é uma FBF.
 2. Se ϕ é uma FBF, então $\neg\phi$ é uma FBF.
 3. Se ϕ e ψ são FBFs, então $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$ e $(\phi \leftrightarrow \psi)$ são FBFs.
 4. Se ϕ é uma FBF e x é uma variável, então $\forall x \phi$ e $\exists x \phi$ são FBFs.
 5. Nada mais é uma FBF.
- **Variáveis Livres e Ligadas:** Uma variável é **ligada** se estiver dentro do escopo de um quantificador. Caso contrário, é **livre**. Em geral, as FBFs que usamos em bases de conhecimento não devem ter variáveis livres.

Exemplos de Sentenças em LPO

- “Todos os humanos são mortais”: $\forall x (Humano(x) \rightarrow Mortal(x))$
- “Sócrates é humano”: $Humano(Scrates)$
- “Existe um aluno que gosta de café”: $\exists x (Aluno(x) \wedge Gosta(x, Caf))$
- “Nenhum pássaro nada” (equivalente a “Para todo pássaro, não é verdade que ele nada”): $\forall x (Pssaro(x) \rightarrow \neg Nada(x))$
- “Todo mundo gosta de alguém”: $\forall x \exists y Gosta(x, y)$
- “Alguém gosta de todo mundo”: $\exists x \forall y Gosta(x, y)$

Semântica da Lógica de Predicados

A semântica da LPO define o significado das FBFs em relação a um modelo do mundo.

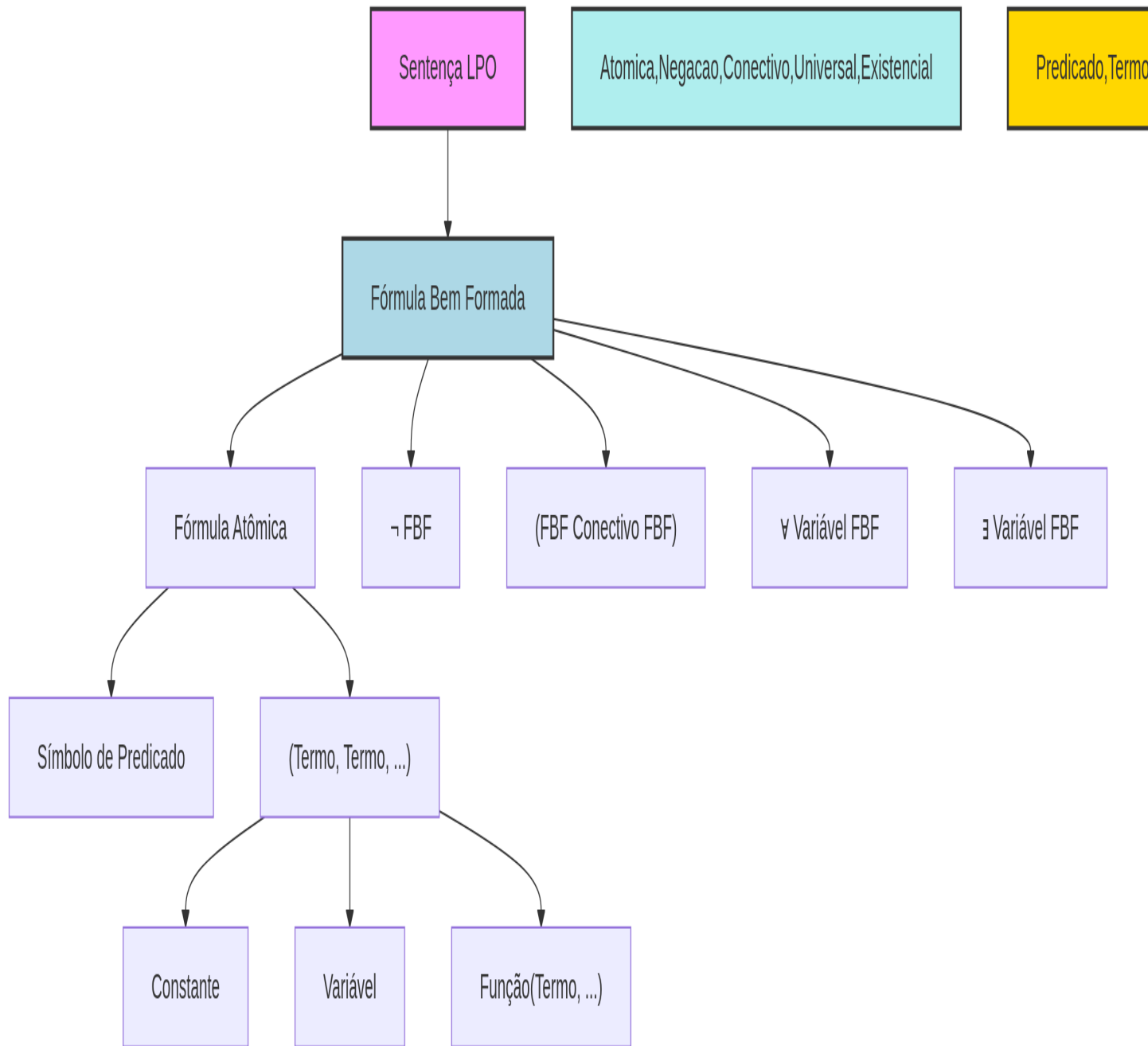


Figure 1: Estrutura Sintática da Lógica de Predicados de Primeira Ordem (LPO)

Modelo

Um **modelo** em LPO é composto por:

1. Um **Domínio (D)**: Um conjunto não vazio de objetos sobre os quais o conhecimento está sendo expresso.
 - *Exemplo*: $D = \{Scrates, Plato, Aristteles\}$.
2. Uma **Interpretação**: Mapeia cada:
 - **Constante** para um objeto em D .
 - **Símbolo de Predicado** de n argumentos para uma relação de n -ários sobre D .
 - **Símbolo de Função** de n argumentos para uma função de n -ários de D^n para D .

O valor de verdade de uma FBF é determinado recursivamente com base na interpretação e nos valores de verdade de suas subfórmulas.

- **Quantificadores**:
 - $\forall x \phi(x)$ é verdadeiro se $\phi(x)$ for verdadeiro para *todo* objeto x no domínio.
 - $\exists x \phi(x)$ é verdadeiro se $\phi(x)$ for verdadeiro para *pelo menos um* objeto x no domínio.

Inferência em Lógica de Predicados

A inferência em LPO é mais complexa do que em LP, devido à presença de variáveis e quantificadores.

Instanciação Universal (Universal Instantiation - UI)

Se temos uma sentença da forma $\forall x \phi(x)$, podemos inferir $\phi(c)$ para qualquer constante c no domínio.

- *Exemplo*:
 - $\forall x (Humano(x) \rightarrow Mortal(x))$
 - Inferimos: $Humano(Scrates) \rightarrow Mortal(Scrates)$

Instanciação Existencial (Existential Instantiation - EI)

Se temos uma sentença da forma $\exists x \phi(x)$, podemos inferir $\phi(k)$, onde k é uma constante **nova** que não aparece em nenhum outro lugar na base de conhecimento (uma **constante de Skolem**).

- *Exemplo*:
 - $\exists x (Pssaro(x) \wedge Voa(x))$
 - Inferimos: $Pssaro(B) \wedge Voa(B)$, onde B é uma nova constante que representa *aquele pássaro específico que voa*.

Modus Ponens Generalizado

Esta é uma extensão do Modus Ponens para LPO. Se temos uma regra universal e um fato que unifica com a premissa da regra, podemos inferir a consequência.

- *Exemplo*:

1. $\forall x (Humano(x) \rightarrow Mortal(x))$
2. $Humano(Scrates)$
3. Usando UI com $x = Scrates$ na primeira, obtemos $Humano(Scrates) \rightarrow Mortal(Scrates)$.
4. Aplicando Modus Ponens com $Humano(Scrates)$ e $Humano(Scrates) \rightarrow Mortal(Scrates)$, inferimos $Mortal(Scrates)$.

Unificação (Unification)

A **unificação** é o processo de encontrar substituições para variáveis que tornam duas expressões lógicas idênticas. É crucial para aplicar regras de inferência como o Modus Ponens Generalizado e a Resolução em LPO.

- **Exemplo:**
 - Expressão 1: $Gosta(Jo\tilde{a}o, x)$
 - Expressão 2: $Gosta(Jo\tilde{a}o, Pizza)$
 - Unificador: $\{x/Pizza\}$ (Substitui x por $Pizza$).
 - Expressão 1: $Gosta(Pai(y), y)$
 - Expressão 2: $Gosta(Pai(Zeca), Zeca)$
 - Unificador: $\{y/Zeca\}$

Resolução em LPO

Assim como em LP, a resolução é um método de prova por refutação em LPO. Envolve a conversão de sentenças para a Forma Normal Clausal (Clausal Form) e a aplicação repetida da regra de resolução, que agora precisa de unificação para combinar literais. É um algoritmo completo e sólido para LPO. ([Russell; Norvig, 2004, p. 289–299](#))

Exemplo de Inferência em LPO (Conceitual)

Considere a seguinte base de conhecimento:

1. $\forall x (Pssaro(x) \rightarrow Voa(x))$ (“Todos os pássaros voam”)
2. $Pssaro(Piupiu)$ (“Piupiu é um pássaro”)
3. $\forall y (Ovo(y) \rightarrow Pssaro(pai(y)))$ (“Todo ovo tem um pai que é pássaro”)
4. $Ovo(OvoDePiupiu)$ (“Existe um ovo de Piupiu”)

Queremos provar que $Voa(Piupiu)$.

- **Passo 1:** Usar Instanciação Universal na sentença 1 com $x = Piupiu$: $Pssaro(Piupiu) \rightarrow Voa(Piupiu)$
- **Passo 2:** Aplicar Modus Ponens generalizado com a sentença 2 ($Pssaro(Piupiu)$) e a conclusão do Passo 1: Inferimos: $Voa(Piupiu)$.

Queremos provar que existe algo que voa (ou seja, $\exists z Voa(z)$).

- **Passo 1:** Já provamos que $Voa(Piupiu)$.

- **Passo 2:** Pela regra de Introdução Existencial, se $\phi(c)$ é verdade para uma constante c , então $\exists x \phi(x)$ é verdade. Inferimos: $\exists z Voa(z)$.

Comparação entre Lógica Proposicional e Lógica de Predicados

Característica	Lógica Proposicional (LP)	Lógica de Predicados (LP)
Elementos Básicos	Proposições atômicas	Objetos, propriedades, predicados
Variáveis	Não possui	Sim (para objetos)
Quantificadores	Não possui	Sim ($\forall, \exists$)
Expressividade	Limitada; apenas fatos “sim ou não”	Rica; pode expressar p
Domínio de Aplicação	Problemas simples, pequenos, sem necessidade de generalização	Problemas complexos, grandes
Complexidade da Inferência	Decidível (tabelas verdade, resolução)	Semidecidível (completo)

LPO em Python (Bibliotecas)

Implementar um motor de inferência completo para Lógica de Predicados do zero em Python é uma tarefa consideravelmente complexa. Envolveria parsers para as FBFs, implementações de unificação e resolução. No entanto, existem bibliotecas que podem ajudar a trabalhar com lógica simbólica em Python.

Uma das bibliotecas mais conhecidas é a **SymPy**, que embora seja focada em matemática simbólica, possui módulos que podem ser usados para lógica. Para sistemas de raciocínio mais robustos, outras ferramentas e linguagens (como Prolog) são mais comuns.

Exemplo (Conceitual - Usando SymPy para Lógica Proposicional Extendida):

Embora SymPy não implemente LPO completa com unificação e resolução por padrão, ela pode representar e manipular expressões lógicas de forma simbólica. Este é um exemplo simplificado que mostra como lidaríamos com algumas expressões.

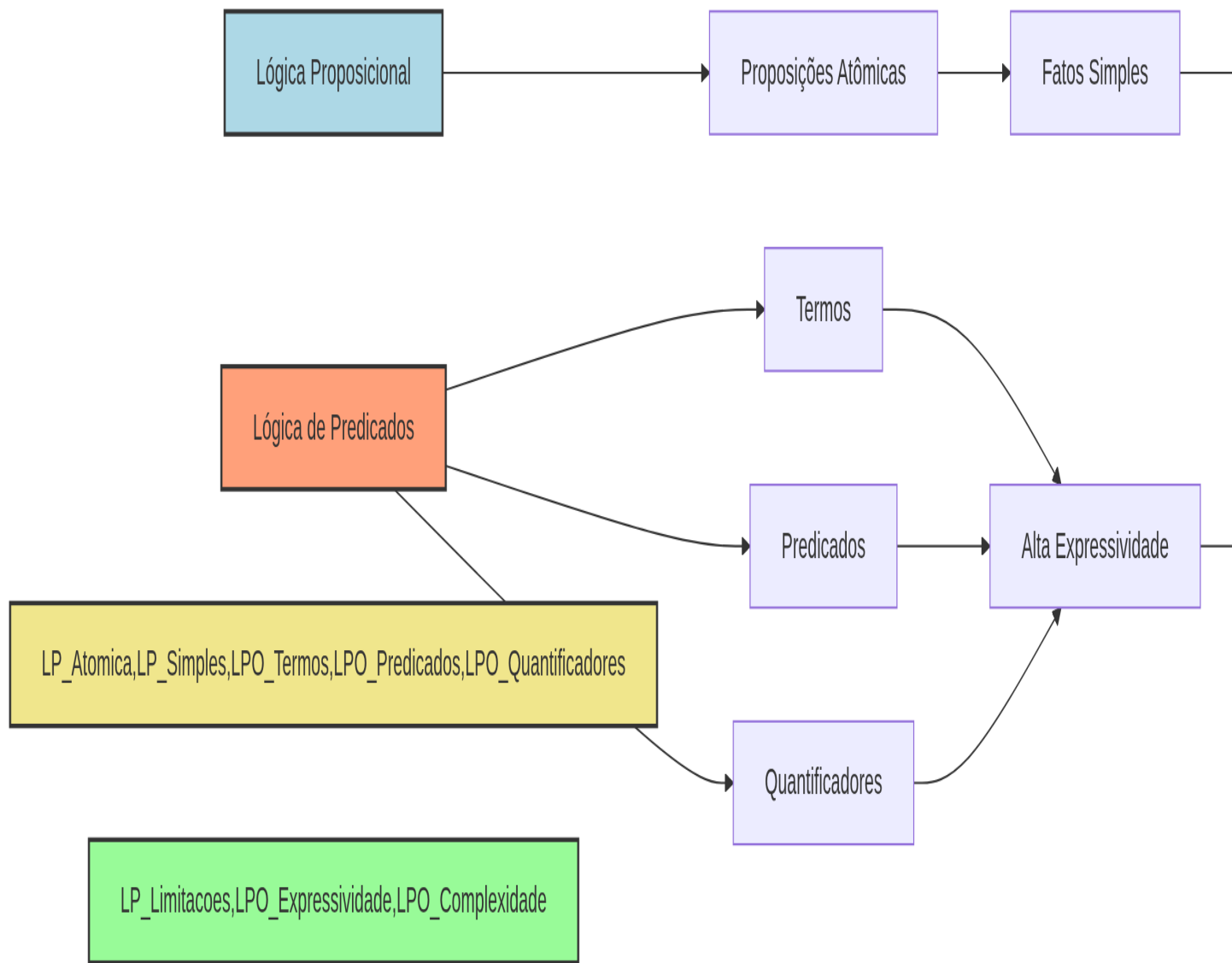


Figure 2: Comparativo entre Lógica Proposicional e Lógica de Predicados de Primeira Ordem


```

from sympy.logic.boolalg import And, Or, Not, Implies, Equivalent
from sympy.abc import x, y, z # Variáveis simbólicas

# Em SymPy, predicados e funções são representados por objetos Function e Symbol
# e quantificadores não são diretamente suportados para inferência de forma nativa
# como em um resolvidor de LPO.
# Este é um exemplo mais ilustrativo das sentenças que queremos representar.

# Definindo símbolos para predicados e funções
# Em um sistema de LPO real, estes teriam aridades definidas
Pássaro = lambda arg: f"Pássaro({arg})"
Voa = lambda arg: f"Voa({arg})"
Humano = lambda arg: f"Humano({arg})"
Mortal = lambda arg: f"Mortal({arg})"
Pai = lambda arg1, arg2: f"Pai({arg1}, {arg2})"

# Definindo constantes
Sócrates = "Sócrates"
Piupiu = "Piupiu"
João = "João"
Pedro = "Pedro"

# Representando sentenças em LPO
# (Note: Isso é uma representação de strings, não um motor de inferência LPO real do SymPy)

# 1. "Todos os humanos são mortais"
# Sem quantificador universal direto em SymPy para inferência como LPO
# Seria uma série de implicações para cada humano conhecido:
# Implies(Humano(Sócrates), Mortal(Sócrates))
# Implies(Humano(Platão), Mortal(Platão))
# ...

# Para expressar a regra universalmente, usamos a notação teórica
sentenca_1 = "x (Humano(x) → Mortal(x))"
print(f"Sentença 1: {sentenca_1}")

# 2. "Sócrates é humano"
sentenca_2 = Humano(Sócrates)
print(f"Sentença 2: {sentenca_2}")

# 3. "João é pai de Pedro"
sentenca_3 = Pai(João, Pedro)
print(f"Sentença 3: {sentenca_3}")

```

Considerações Finais

A Lógica de Predicados de Primeira Ordem é uma ferramenta poderosa para a Representação de Conhecimento na Inteligência Artificial. Sua capacidade de expressar propriedades de objetos, relações e quantificações a torna indispensável para domínios complexos, onde fatos não podem ser simplesmente representados como proposições atômicas. Embora a inferência em LPO seja computacionalmente mais desafiadora, a sua expressividade é um trade-off necessário para modelar cenários ricos do mundo real, formando a base de sistemas de raciocínio lógico mais avançados.

Verificação de Aprendizagem

Responda às seguintes questões e implemente as tarefas propostas para solidificar seu entendimento sobre a Lógica de Predicados.

1. Diferenças LP vs. LPO:

Qual a principal motivação para usar Lógica de Predicados em vez de Lógica Proposicional na representação de conhecimento de um agente inteligente? Cite dois exemplos de sentenças que são facilmente representáveis em LPO, mas difíceis ou impossíveis em LP.

2. Sintaxe da LPO:

Identifique e explique os três tipos de termos em Lógica de Predicados. Para cada tipo, forneça um exemplo concreto.

3. Quantificadores:

- Explique a diferença entre o quantificador universal ($\forall$) e o quantificador existencial ($\exists$).
- Traduza as seguintes sentenças para FBFs em Lógica de Predicados:
 - “Todos os carros têm rodas.”
 - “Existe um professor que é amigo de todos os alunos.”
 - “Nenhum animal é imortal.” (Tente duas formas diferentes de expressar isso usando quantificadores e negação)
 - “Nem todos os alunos gostam de matemática.”

4. Inferência em LPO:

Você tem a seguinte base de conhecimento (KB):

- $\forall x (Homem(x) \rightarrow Mortal(x))$
- $\forall y (Grego(y) \rightarrow Homem(y))$
- $Grego(Plato)$

- Usando as regras de inferência (Instanciação Universal e Modus Ponens Generalizado), mostre os passos para provar que $Mortal(Plato)$ é uma consequência lógica desta KB.
- Explique o conceito de **unificação** e como ele é essencial para a inferência em LPO, usando um exemplo diferente dos apresentados na aula.

5. Limitações e Desafios:

Mesmo com sua maior expressividade, a Lógica de Predicados de Primeira Ordem ainda possui desafios. Pesquise brevemente (fora dos materiais da aula, se necessário) e discuta uma limitação ou desafio que a LPO enfrenta na representação de conhecimento do senso comum ou em domínios mais complexos.

Referências Bibliográficas

RUSSELL, Stuart J.; NORVIG, Peter. **Inteligência Artificial: Um Enfoque Moderno**. 2. ed. Rio de Janeiro: Prentice Hall, 2004.