



دانشگاه اصفهان

دانشکده مهندسی کامپیوتر

نظریه زبان‌ها و ماشین‌ها

گزارش پروژه اختیاری

تبدیل عبارت‌های منظم به گرامرهای خطی و

ساده کردن گرامرهای منظم

بررسی عضویت رشته‌ها در این گرامر

استاد: دکتر مرجان کائدی

متین اعظمی (4003623003)

بهار 1402

Kleene

تبدیل عبارت‌های منظم به گرامر خطی از راست و چپ می‌تونیم از دو راه مختلف استفاده کنیم:

- **روش غیر مستقیم:** تبدیل عبارت منظم به NFA و سپس تبدیل آن به گرامر خطی از راست یا چپ
- **روش مستقیم:** استفاده از الگوریتم Kleene

این الگوریتم به صورت بازگشتی عمل می‌کند و به صورت پایین به بالا ابتدا حالت پایه و گرامر و می‌سازیم و بعد با اضافه شدن هر عملگر جدید با حفظ خاصیت خطی بودن گرامرها و ترکیب می‌کنیم.

Characters To convert a single character a , we make a simple production.

$$S \rightarrow a$$

where S is the start symbol.

Concatenation To concatenate two regular expressions, add the second start symbol to the end of any "accepting" states from the first grammar.

$$\begin{array}{l} \text{Regexp: } a \\ S_1 \rightarrow a \end{array}$$

$$\begin{array}{l} \text{Regexp: } b \\ S_2 \rightarrow b \end{array}$$

$$\begin{array}{l} \text{Regexp: } ab \\ S_1 \rightarrow aS_2 \\ S_2 \rightarrow b \end{array}$$

Choice To choose between two regular expressions, add a new start symbol that "picks" one of the choices.

$$\begin{array}{l} \text{Regexp: } a \\ S_1 \rightarrow a \end{array}$$

$$\begin{array}{l} \text{Regexp: } b \\ S_2 \rightarrow b \end{array}$$

$$\begin{array}{l} \text{Regexp: } a|b \\ S \rightarrow S_1|S_2 \\ S_1 \rightarrow a \\ S_2 \rightarrow b \end{array}$$

Kleene Star If S is the start symbol, then for every rule of the form $A \rightarrow x$ ("accepting states") add another rule of the form $A \rightarrow xS$. Also add an ϵ rule.

$$\text{Regexp: } a|b$$

$$\begin{array}{l} S \rightarrow S_1|S_2 \\ S_1 \rightarrow a \\ S_2 \rightarrow b \end{array}$$

$$\text{Regexp: } (a|b)^*$$

$$\begin{array}{l} S \rightarrow S_1|S_2|\epsilon \\ S_1 \rightarrow a|aS \\ S_2 \rightarrow b|bS \end{array}$$

ساده کردن گرامرهای منظم

برای ساده کردن گرامرها به ترتیب زیر عمل می‌کنیم:

- حذف متغیرها و قوانین غیرقابل دسترس از S
- حذف قوانین میرا و لامبدا
- حذف قوانین یک‌ه و تکراری

تبدیل گرامر به فرم نرمال چامسکی

بعد از ساده کردن گرامر آن‌را به فرم نرمال چامسکی تبدیل می‌کنیم.

- متناظر کردن الفبای زبان به متغیرهای جدید
- جایگذاری متغیرهای متناظر با هر نماد در قوانین با طول بزرگتر 2
- تبدیل قوانین شامل بزرگتر از 2 متغیر به 2 متغیر با استفاده از متغیرهای میانی

الگوریتم CYK

بعد از تبدیل گرامر به فرم نرمال چامسکی با اعمال الگوریتم CYK روی هر رشته می‌تونیم عضویت آن رشته در عبارت منظم و گرامر را بررسی کنیم.

function CYK($G = (N, \Sigma, S, P), w \in \Sigma^*$): $\{false, true\}$
Precondition: G is in the Chomsky Normal Form

1. $n \leftarrow |w|$
2. **for all** $i = 1, \dots, n$ **do** $N_{i,1} \leftarrow \{A \in N \mid (A, w_{i,1}) \in P\}$
3. **for all** $j = 2, \dots, n$ **do**
4. **for all** $i = 1, \dots, n - j + 1$ **do**
5. $N_{i,j} \leftarrow \emptyset$
6. **for all** $k = 1, \dots, j - 1$ **do**
7. $N_{i,j} \leftarrow N_{i,j} \cup \{A \in N \mid (A, BC) \in P, B \in N_{i,k}, C \in N_{i+k,j-k}\}$
8. **end for**
9. **end for**
10. **end for**
11. **return** $S \in N_{1,n}$

ابزار و زبان‌های استفاده شده

زبان استفاده شده برای پیاده‌سازی پروژه Cpp بوده است و در بستر GitHub قرار داده شده است.

- <https://github.com/InFluX-M/Automata>

منابع

- Formal Languages and Automata Course, Dr.Marjan Kaedi
- <https://courses.engr.illinois.edu/cs421/sp2020/slides/07.2.2-right-linear-grammars.pdf>
- <https://csc.lsu.edu/~busch/courses/theorycomp/>
- Peter Linz, An Introduction to Formal Languages and Automata, Jones & Bartlett Learning, 2017