Presented by 杨欣然 郑棐巍

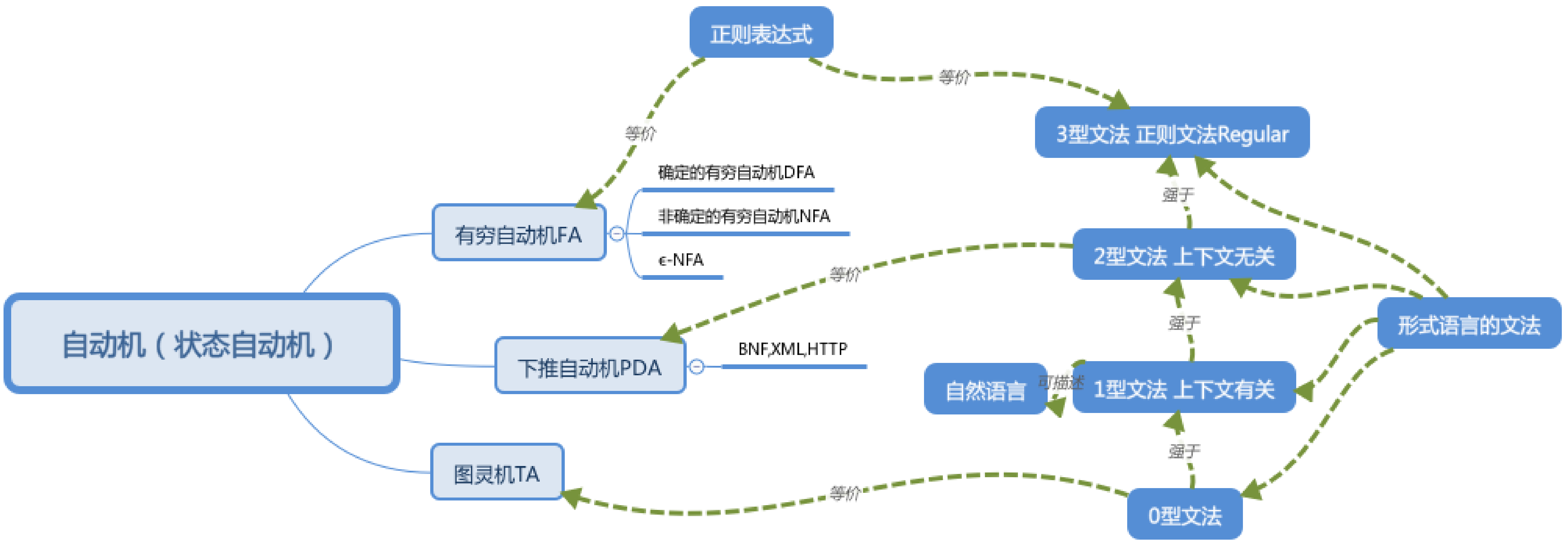
Turing machine

In honor of Alan Turing,
Godel and Cantor

Simulation

- http://morphett.info/turing/turing.html
- Rules
- Design a Palindrome detector
- Show examples of Binary addition and Binary multiplication
- Universal Turing machine
- Not consider the space and efficiency

- read the most left one
- goto the right side
- judge the most right one
- goto the left side ;loop



乔姆斯基四类文法

Q1:语言与可计算性有什么关系?

Q2: 0型文法代表着文法构造能力的极限，它能否表达所有语言?

Definition

- 可计算性理论（Computability theory）研究在不同的计算模型下哪些算法问题能够被解决。
- 可计算：问题能够被图灵机解决
- 解决：图灵机、文法、lambda、马尔科夫转移链......
- 判定问题是无穷多个同类个别问题的总称
- 判定问题怎么刻画绝大部分问题？

转化

- 考虑排序问题，输入一串数（符号），希望输出一串数（符号）
- 考虑一般问题：可以用英文、数字表示输入
- 把自然数关联到公理和规则
- 求解一个函数
- 哥德尔

哥德尔编码

- 在计算机上一切都是数字。比如，字母 “N” 是78，字母 ″u″ 是 117
- Number 0 exists
- 78117109981011143248321011201051151161115
- 对余下的所有初级公理和规则进行同样的编码，接着我们可以开始对这个形式系统的第一个推论进行编码，再对推论的推论进行编码，等等等等。最后，任何公理或者推论序列都变成一个数字。

哥德尔宣称

- 我来告诉你一个特别的方法，怎样把任意证明变成一个极大的数字，并且怎样准确地把这些数字转回到原来的证明。只要你有任何证明，10页纸甚至1000页纸长的证明，都仅仅是一个唯一的完全表示你的证明的数字。并且这一点，对于所有可能的证明都成立。从现在开始，我们就只处理和证明关于数字的定理和关于数字的函数。这就简单得多了。

可计算函数

- 所有问题就成了计算对应函数值（自然数）的问题
- 图灵机能计算该函数就可以解决该问题
- 改变输入方式，穷举自然数，判定是否为解来求解函数（集合无穷也是可以的）
- 把这一列输入的字符串作为语言
- 可计算语言：一个语言S可以被图灵机接受

图灵可判定语言

- Turing machine decidable language： S
- 如果判断在S中则接受，否则停机
- 可计算语言
- 哪些问题不可计算呢？

停机问题

- A={<M, ω>｜M描述一台图灵机，且M描述的机器接受ω}
- 假设存在H
- Post correspondence problem
- Mortality (computability theory)
- Entscheidungsproblem

查 · 论 · 编

自动机理论：形式语言和形式文法			
乔姆斯基层级	文法	语言	极小自动机
类型 0	无限制	递归可枚举	图灵机
—	(无公用名)	递归	判定器
类型 1	上下文有关	上下文有关	线性有界
—	附标	附标	嵌套堆栈
—	Linear context-free rewriting systems etc.	Mildly context-sensitive	Thread automata
—	树-邻接	适度上下文有关	嵌入下推
类型 2	上下文无关	上下文无关	非确定下推
—	确定上下文无关	确定上下文无关	确定下推
—	Visibly pushdown	Visibly pushdown	Visibly pushdown
类型 3	正则	正则	有限
—	—	Star-free	Counter-free (with aperiodic finit

每个语言范畴都是其直接上面的范畴的真子集 每个语言范畴内的语言都可以用同一行的文法和自动机表示

图灵可识别语言

- Turing machine recognizable language
- 对应的是0型语言（半判定语言）
- 如果判断在S中则接受
- A={<M, ω> | M描述一台图灵机，且M描述的机器接受ω}
- 难道还有不可识别语言？

不可识别语言

- 可计算函数是定义在自然数上的
- 通过某种方法可以“遍历”自然数、整数甚至有理数
- 可数集：直观上可以知道下一个是什么
- 根据集合论的定义，可数集是可以和N建立一一关系的集合 （回忆：N的幂集也是不可数的）

全部语言的集合是不可数的

- 不可数及不能与N建立对应关系
- 可以证明：R不可数（其实与N的幂集等势）（利用康托对角线法）
- 可以证明：无限的二进制序列的集合与R可以建立对应关系（直观：无限与无限产生了更高的无限）
- 可以证明：无限的二进制数的集合可以与所有语言的集合建立对应关系
- 易证：图灵机的集合是可数的

结论

- 存在图灵机不可识别语言
- 要例子?
- 难!
- 以实数为例，存在实数是不可定义的。因为定义用的是有限语言，有限语言与N可以建立对应关系。所以，不可识别语言大都不可定义。

查 · 论 · 编

自动机理论：形式语言和形式文法			
乔姆斯基层级	**文法**	**语言**	**极小自动机**
类型 0	无限制	递归可枚举	图灵机
—	(无公用名)	**递归**	判定器
类型 1	上下文有关	上下文有关	线性有界
—	附标	附标	嵌套堆栈
—	Linear context-free rewriting systems etc.	Mildly context-sensitive	Thread automata
—	树-邻接	适度上下文有关	嵌入下推
类型 2	上下文无关	上下文无关	非确定下推
—	确定上下文无关	确定上下文无关	确定下推
—	Visibly pushdown	Visibly pushdown	Visibly pushdown
类型 3	正则	正则	有限
—	—	Star-free	Counter-free (with aperiodic finit

每个语言范畴都是其直接上面的范畴的真子集 每个语言范畴内的语言都可以用同一行的文法和自动机表示

- 递归集　　递归集最初是对于元素都是自然数的集合定义的，它们是有算法确定每个自然数是否为其元素的集合。
- 递归可枚举集　　如果对于集合A可以编一个程序P，输入域中任意元素x，若x∈A，则P的执行将终止并输出“是”，否则P 的执行不终止，就称A为递归可枚举集。A为递归可枚举集的充分必要条件是可以编一个程序枚举A的元素，即打印A的元素，使得对于 A中任意元素，只要时间足够长总会在打印纸上出现。

有关的拓展话题

- 比图灵机更强的计算模型 https://www.zhihu.com/question/21579465
- 谕示机 ——由图灵提出
- Blum - Shub - Smale machine
- …
- 都是理论上的（而且我看不懂）

有关的拓展话题

- 0型语言
- 图灵机的等价模型
- 不可计算问题的证明
- 哥德尔不完备定理
- 连续统假设
- 集合论
- 康托的对角线法
- P 与 NP （都是可判定问题）

以及

- 学好集合论
- 各种悖论

引用及鸣谢

- University of Melbourne, School of Mathematics and Statistics http://morphett.info/turing/turing.html Turing machine simulator
- Wikipedia
- Alan Turing *ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM*
- *silverbullettt 《关于图灵机的三个问题》* http://blog.sina.com.cn/s/blog_4dff87120100ylfv.html
- *刘未鹏 《康托尔、哥德尔、图灵——永恒的金色对角线》*
- *陈有祺 《形式语言与自动机》*
- *http://skibinsky.com/godel-turing-and-cantor-the-math/*
- www.zhihu.com
- 顾森讲数学
- 《算法图灵机及可计算性理论》ppt
- 宋方敏 《20世纪最伟大的智者之一Alan Turing》
- https://yq.aliyun.com/articles/72782
- CSDN
- 百度百科
- 张健 《逻辑公式的可满足性判定—方法 工具及应用》
- 以及其它未列出的参考
- 赫兆宽 杨跃 《集合论 对无穷概念的探索》