Genifer
– an artificial general intelligence

YKY (甄景贤 general.intelligence@Gmail.com)

with input from:
Abram Demski
Ben Goertzel
Martin Magnusson
William Taysom
Russell Wallace
Pei Wang

Preface, executive summary, to-do list

1. This book is a perpetual draft.
2. My personal reason for developing AGI is to achieve life extension.
3. The source code of Genifer is hosted on Google Code, including some very easy tutorial slides. Also feel free to contact me!

— YKY

Executive summary:

Inference: Genifer descended from classical logic-based A.I. Its 3 modes of inference are deduction, abduction (explaining), and induction (learning). This is common to NARS, OpenCog, Cyc.

Logic: Genifer is based on an **algebra of concept composition**, which replaces predicate logic as the internal structure of propositions.

KB: Genifer's KB stores logic formulas, similar to classical A.I. systems such as Cyc, and NARS. OpenCog is an exception in that it stores its knowledge as a hypergraph called AtomSpace.

Uncertainty: Genifer uses fuzzy-probabilistic logic, the probabilistic part is an exact algorithm for belief propagation in Bayesian networks. The fuzzy-probabilistic calculus is created by YKY based on the Beta distribution.

Bootstrapping: Genifer will be written in its own language, which is a **logical-functional** programming language based on Genifer's logic and an existing functional programming language such as Clojure or Haskell.

To-do:

Ch 1 (Introduction) Explain the new ideas that I learned about the relationship between propositional logic and topological logic.

Ch 2 (Architecture) Explain AIXI, algorithmic complexity, Solomonoff induction, etc. Explain distributive architecture. New idea that bootstrapping is possible.

Ch 3 (KR) — ok —

Ch 4 (Logic) New logic of concept composition. Ideas about equational unification and concepts. Explain background notions, eg paradoxes.

Ch 5 (Z) Add new idea on the "Java-girl paradox", which is in draft paper.

Ch 8 (Inference) Copy and paste Bayesian inference and factor graph stuff from the Lisp code to here.

Ch 9 (Pattern recognition) Matrix technique on similarity.

Ch 11 (Learning) A lot of new material is in the slides.

Ch 12 (NL) New idea of semantic parsing. New diagrams from GUI.

Ch 13 (Memory) Explain hierarchical clustering idea, ontology.

Ch 14 (Planning) May need re-think.

Ch 18 (Implementation) Bootstrap Genifer in its own language.

Appendix A Recommend more books for AGI sub-areas. Especially math books.

Contents

0 导论

0.1 鸡与蛋问题，罐头中的罐头刀

Part I

技巧

1 机器学习基础

1.1 归纳偏见与「没有免费午餐」

1.2 结构主义、后结构主义

2 逻辑

2.1 人脑思考的三大模式

2.1.1 Deduction

2.1.2 Abduction

2.1.3 Induction

2.2 命题逻辑

2.3 谓词逻辑 / 一阶逻辑

2.4 逻辑推理算法

2.4.1 同一化算法 (unification)

2.4.2 消解算法 (resolution)

2.5 二阶逻辑 / 高阶逻辑

2.6 λ-演算，组合逻辑

2.7 代数逻辑、逻辑的几何化

2.8 范畴论、范畴逻辑

2.9 量子逻辑

2.10 项重写系统

2.11 图重写系统，超图

3 不确定性

3.1 模糊性

3.2 机率

3.2.1 Bayesian networks

3.3 可信度

3.4 推理算法

3.4.1 MCMC (Markov chain Monte Carlo)

4 神经网络

4.1 神经科学

4.1.1 大脑结构

4.1.2 神经元

4.1.3 神经化学

4.2 神经网络的数学

4.2.1 非线性分析

4.2.2 拓扑度理论

4.2.3 同调论 / 奇点理论

4.2.4 调和分析

4.3 深度学习

5 进化算法

5.1 自然进化历史

5.2 进化算子及其算子谱

6 强化学习

6.1 控制论，微分几何

6.2 最优化

Part II

功能组别

7 模式识别

7.1 视觉

8　信念修正，真理维修

9 归纳学习

9.1 基於逻辑的归纳学习

10 自然语言

10.1 语法

10.2 语义

10.2.1 Abduction-as-interpretation

10.2.2 Montague 语法

10.2.3 Categorial 语法

11 计划

11.1 自动程式生成

Part III

系统

12 认知系统架构

13 记忆系统

13.1 工作记忆

13.2 事件记忆

14 实践

14.1 道德问题

14.2 商业化

Symbols

$\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$	classical number systems	
$\mathbb{Hyp}$	hypothesis space	§??
$\mathbb{Prop}$	(ground) proposition space	

General logic:

$\exists, \forall$	classical existential and universal quantifiers	
$\wedge, \vee, \neg$	classical binary logic AND, OR, NOT	
$\rightarrow$	(classical) implication	§??
$\vdash$	entailment, syntactic	
$\models$	entailment, semantic	
$=$	equality (logic predicate)	§??
$\approx$	similarity = fuzzy equality (logic predicate)	§??
$\subseteq$	inclusion ("is-a" relation)	§??
$\sim$	association (logic predicate)	
$a \circ b$	composition of concepts	§??
(a, b)	pairing or union	§??
$\lambda x.Mx$	lambda abstraction	
$M : \tau$	(type theory) expression M is of type τ	
$t \overset{R}{\Rightarrow} t'$	t rewrites to t' under rewriting system R	
$t \overset{R}{\rightsquigarrow} t'$	t narrows to t' under rewriting system R	§??
$A \bowtie B$	unify(A,B)	§??
[s_1]: formula	KB stores statement s_1	

Fuzzy and probabilistic logic:

$\#x.Q(x)$	probabilistic quantifier ("for some")	§??
$\rightarrow\!\!\!\!\triangleright$	probabilistic implication (= Bayesian network link)	§??
$\overset{Z}{\wedge}, \overset{Z}{\vee}$	fuzzy AND and OR	§??
$\overset{P}{\wedge}, \overset{P}{\vee}$	probabilistic AND and OR	§??
$\odot$	a (fuzzy or probabilistic) operator that combines AND and OR	§??
$\Gamma(\cdot)$	fuzzy modifier	§??
ξ	point of neutrality (fuzzy logic)	§??
w	total number of support for a hypothesis	§??
w^+, w^-	positive and negative support for a hypothesis	§??

Categories of truth values:

$\mathcal{B}$	binary logic
$\mathcal{P}$	(binary) probabilistic logic
$\mathcal{Z}$	pure fuzzy logic
$\mathcal{P}(\mathcal{B})$	binary-probabilistic logic
$\mathcal{P}(\mathcal{Z})$	fuzzy-probabilistic logic

Miscellaneous:

"text"	texts in English / natural language
`source code`	source code

formula	logic formulas
To do: ...	things to do

Bibliography

致谢

In addition to the people listed on the title page, I'd like to thank the AGI mailing-list participants for years of discussions.