2025年秋季学期

智能化软件系统与工程

马 郓

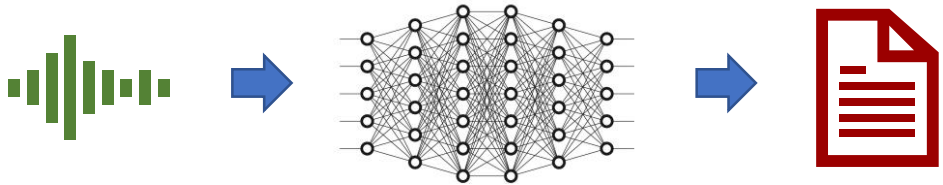
人工智能研究院

北京大学 PEKING UNIVERSITY

案例：语音转文本

案例：语音转文本

- **语音转文本有很多应用场景**
  - 科研人员分析访谈文本
  - 为视频生成字幕
- **传统依赖人工转写，成本很高（数百元/小时）**
- **目前已经有许多语音转文本的机器学习模型**



案例：语音转文本

北京大学 PEKING UNIVERSITY

- **假设你完成了一项针对特定领域语音识别的研究工作**
  - 基于公共访谈数据训练深度神经网络，并在规模较小的人工标注特定领域语料库上进行了迁移学习
  - 得到的模型可检测特定领域的专业术语
  - 在医学、少数民族文化等领域的讲座，以及Rust编程大会的讲座中，均展现出极高的识别准确率；相关成果已在顶级会议上发表

将这个研究成果做成商业化软件产品，销售给科研人员和会议组织者！

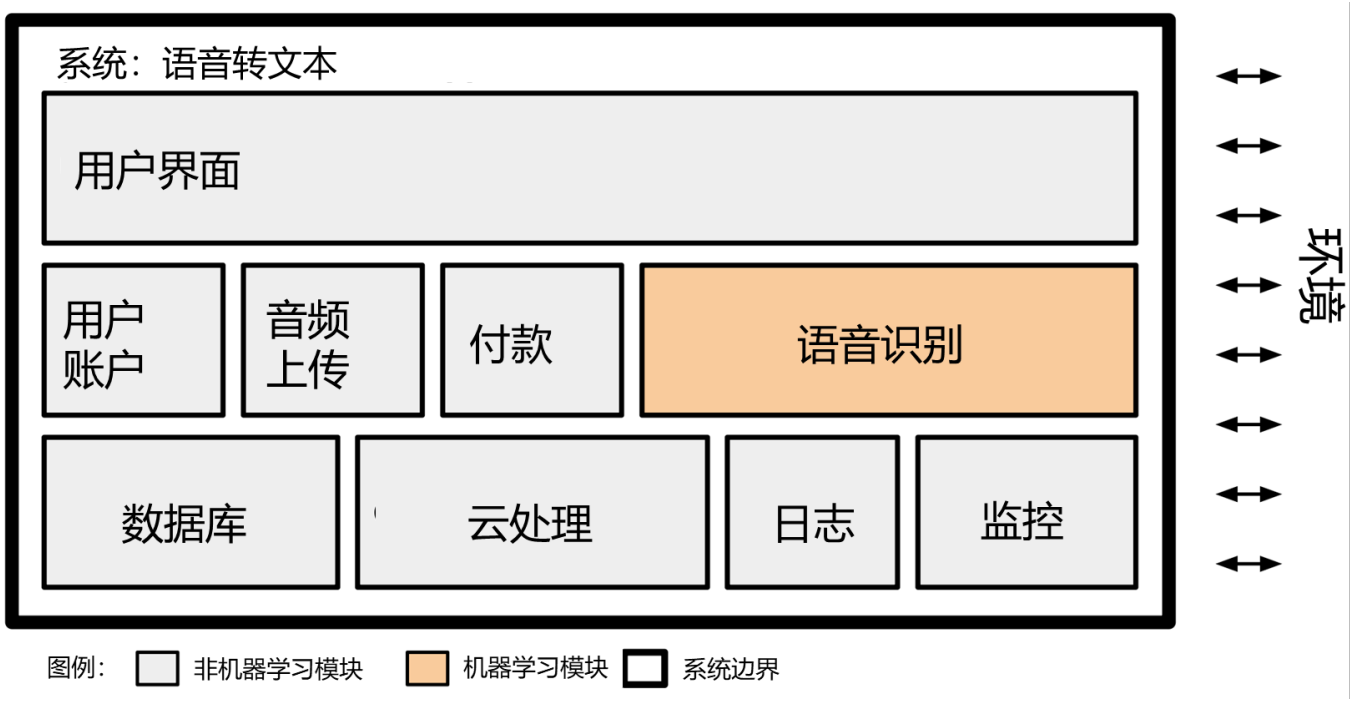
案例：语音转文本

构建商业化产品可能面临的挑战是什么?

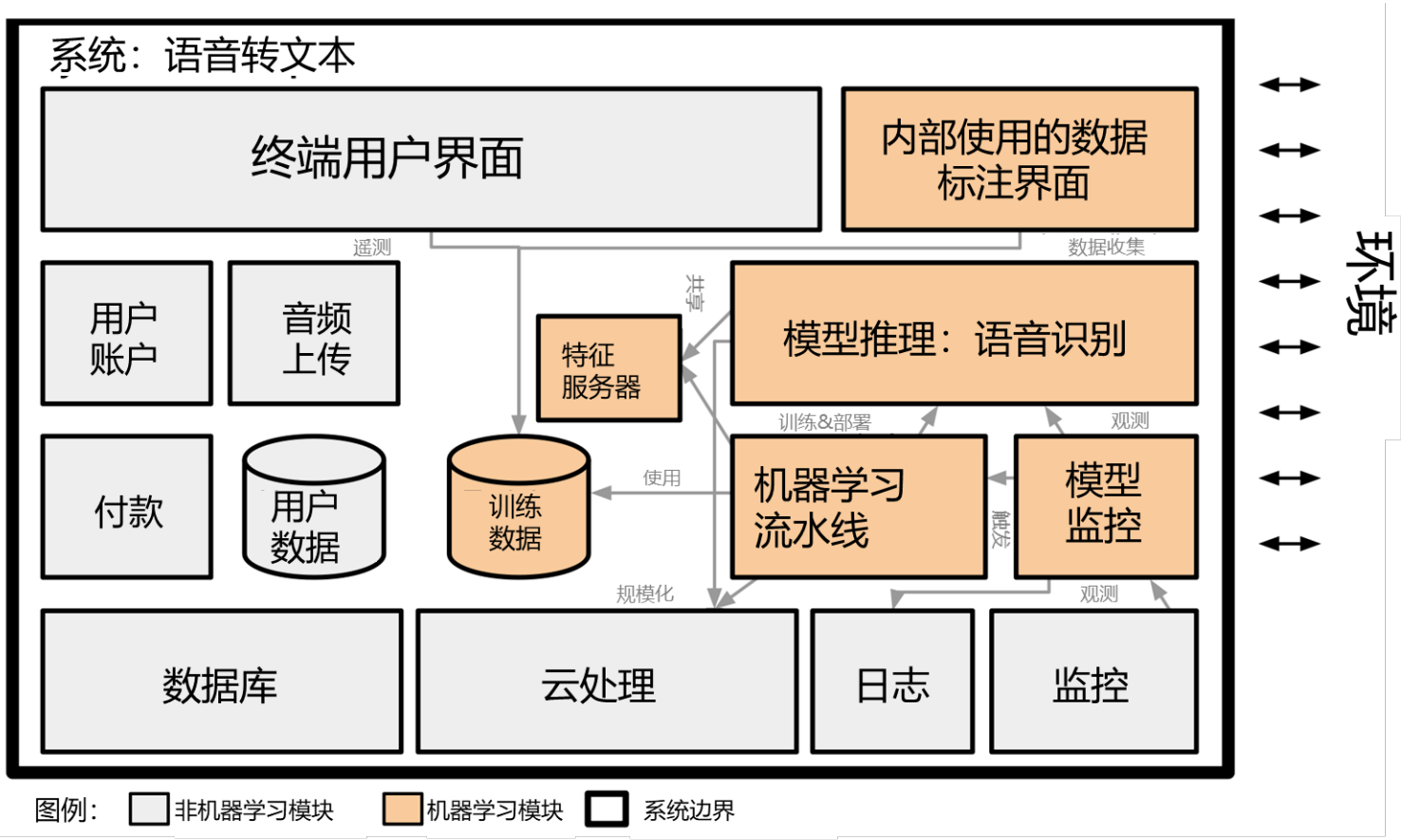
- **可以从以下几个方面思考：**
  - 1. 机器学习的挑战
  - 2. 开发软件产品的挑战
  - 3. 产品发布后运维的挑战
  - 4. 团队组织或管理的挑战
  - 5. 商业的挑战
  - 6. 安全或伦理的挑战
- **每位同学选择至少1个方面，将你的见解写在微信群中**

案例：语音转文本

除了机器学习模型以外，构建语音转文本的商业化产品还需要哪些其他模块？

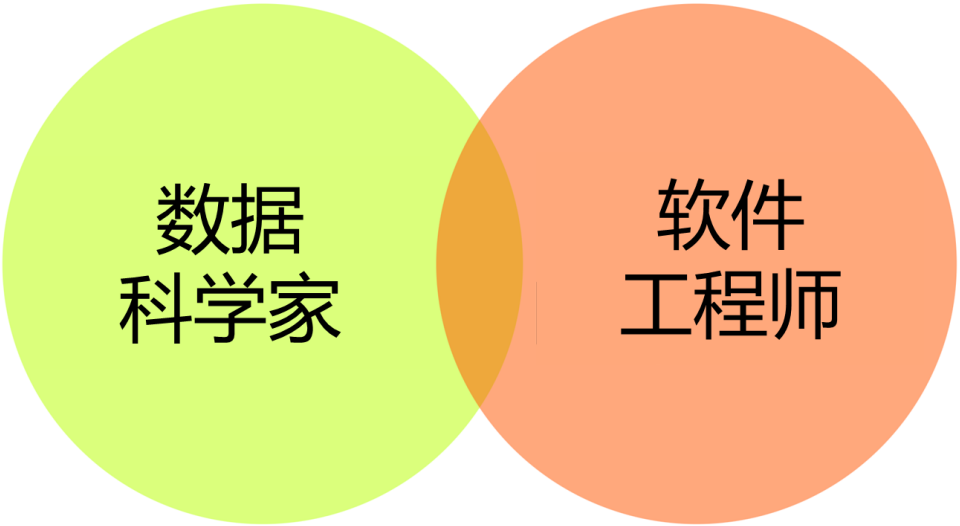


案例：语音转文本


系统：语音转文本
终端用户界面
内部使用的数据标注界面
遥测
数据收集
用户账户
音频上传
特征服务器
共享
模型推理：语音识别
训练&部署
观测
付款
用户数据
训练数据
使用
机器学习流水线
触发
模型监控
规模化
观测
数据库
云处理
日志
监控
环境
图例：
非机器学习模块
机器学习模块
系统边界

开发团队

- 通常使用固定数据集进行训练和评估
- 注重准确性
- 使用Jupyter Notebook或类似工具进行原型设计
- 精通建模技术和特征工程
- 模型规模、可更新性和实施稳定性通常不重要



- 构建产品
- 关注成本、性能、稳定性和发布时间
- 通过客户满意度识别质量
- 产品必须具备可扩展性，能够处理大量数据
- 需要自动检测和处理错误
- 产品需长期维护、升级和扩展
- 需考虑安全性、可靠性和公平性方面的要求

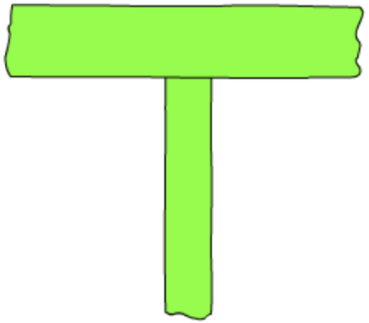
数据工程师 + 领域专家 + 商务团队 + 项目经理 + 设计师+ 安全与安保专家 + 律师 + 社会科学家 + ...

开发团队

T型人才

I型：擅长一件事

通才
很多方面都很擅长
但没有一项是专家



T型
很多方面都很擅长
在其中一项是专家

课程介绍

具有软件工程风格的课程

- 专注于工程判断
- 强调权衡和决策，而非单一正确答案
- 注重实践参与、系统构建、测试和自动化
- 注重团队合作

×

互动性较强的课程
积极参与课堂讨论
贡献个人观点和经验

√

先修要求

北京大学 PEKING UNIVERSITY

- **具备一定的机器学习经验**
  - 了解基本的数据科学流程，如数据清洗、特征工程、使用机器学习库
  - 对机器学习方法有宏观了解
    - 监督学习
    - 回归、决策树、神经网络
    - 准确率、召回率、精确率、ROC曲线
  - 最好有使用notebooks、sklearn或其他框架的经验
- **具备基础的编程和命令行技能**
- **不需要软件工程知识**
  - 有产品团队的团队合作经验会有帮助，但不是必须的
  - 不需要了解需求分析、软件测试、软件设计、持续集成、容器、流程管理等

摸底问卷（不计分）：自我评价对本课程对所需机器学习知识的掌握程度，可回顾或补充学习缺失的知识

授课方式与成绩评定

- **课堂讲授（课堂参与20%）**
  - 理论知识讲解与研讨（~80分钟）
  - 工具/系统实操演示（~20分钟）
- **单人实践（日常作业40%）**
  - 课堂演示工具/系统的课后练习，完成指定任务
- **团队实践（项目大作业40%）**
  - 3~5人一组
  - 完成一个智能化软件系统——大模型智能体的开发
  - 设立三次课堂汇报，用于展示小组阶段性成果
- **采用等级制评分，无期中和期末考试**

授课内容

北京大学 PEKING UNIVERSITY

周次	日期	授课内容	实践内容
1	9月12日	课程介绍与智能化软件概述	
2	9月19日	AI系统开发过程与团队合作	Lab1: Git合作
3	9月26日	AI系统需求工程	Lab2: Restful API封装与调用
4	10月3日	放假	-
5	10月10日	团队项目汇报1	大模型智能体：需求工程
6	10月17日	模型质量	Lab3: Zeno模型测试
7	10月24日	AI系统质量	Lab4: AI辅助编程
8	10月31日	AI系统设计	Lab5: Docker容器化部署
9	11月7日	数据质量	Lab6: Kafka流处理
10	11月14日	规模化AI系统	Lab7: Kubernetes容器编排
11	11月21日	团队项目汇报2	大模型智能体：系统设计与实现
12	11月28日	ML流水线自动化	Lab8: MLFlow自动化流水线
13	12月5日	AI系统运维	Lab9: Jenkins持续集成
14	12月12日	AI系统遥测	Lab10: Prometheus/Grafana监控系统
15	12月19日	负责任的AI系统	Lab11: 可解释性工具
16	12月26日	团队项目总结汇报	大模型智能体：开发运维一体化

AI系统开发团队与需求

质量驱动的AI系统设计

AI系统的开发运维一体化

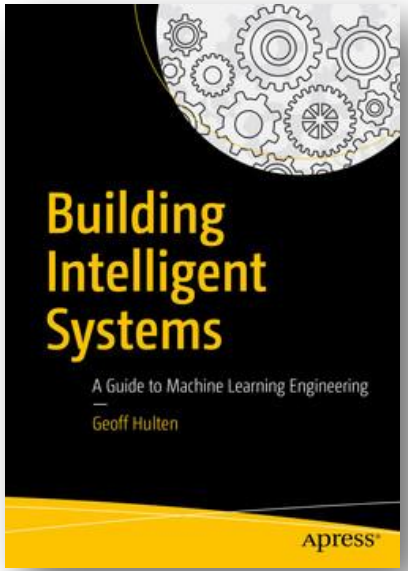
团队实践

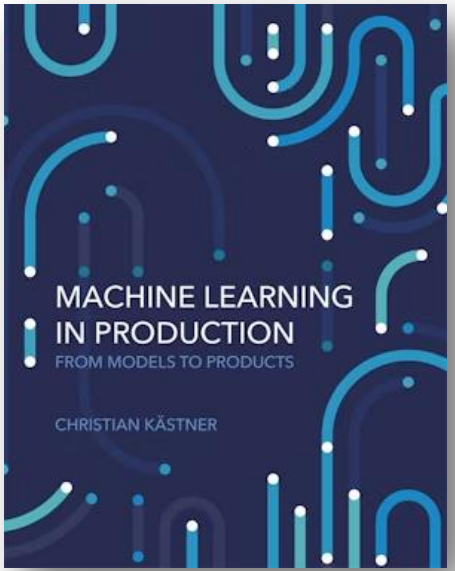
北京大学 PEKING UNIVERSITY

- **3~5人一组团队合作**
- **完成一个大模型智能体开发**
  - 选定一个课程，针对该课程开发“课程问答+”智能体
  - 具有简单的知识问答功能与一个额外功能（如化学课-实验报告评测；计概课-代码审查等，团队自行提出）
- **体验智能化软件生命全周期，包括需求-开发-运维三大阶段**
- **分数占比40%，主要考虑如下方面：**
  - 客观部分：完备性与健壮性、软件使用效果、持续运维设计情况
  - 主观部分：教师评价、组内互评、组间互评（可能）
- **后续布置详细要求**

参考资料

- **本课程参考自CMU的Machine Learning in Production**
  - 最新课程网址：https://mlip-cmu.github.io/s2025/
- **参考书**





在课程群中发电子版，仅限本课程使用

开源版：https://mlip-cmu.github.io/book/

关于生成式AI的使用

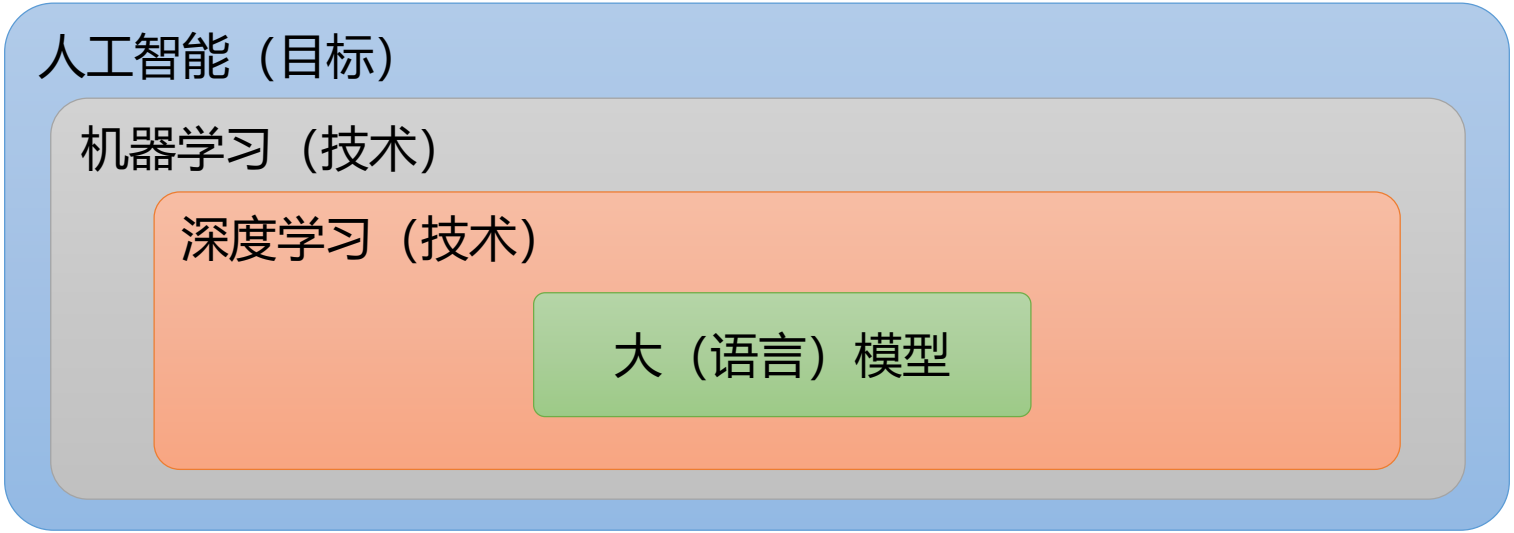
- 在课堂讨论环节不鼓励使用生成式AI，请独立思考
- 在单人作业和团队作业环节不限制生成式AI的使用
- 课程将讲解使用AI辅助编程和开发的技巧
- 鼓励探索如何利用AI提高开发效率，并分享经验
- 需对自己提交的内容负责！

特别声明

- **本课程为首次开课，具有很强的实验性质**
  - 开设一门新课类似“开发一个新软件”
  - 单人实践内容虽有借鉴，但进行了简化调整
  - 团队实践为全新设计，各项机制需在开展过程中完善
- **部分内容属于前沿研究，可能存在争议，尚无定论**
  - 许多概念的内涵和外延都在快速变化
  - 课程尽量从统一视角给出一致的术语定义，不一定是主流观点
  - 例如，课程名字就有非常多的说法
    - ML-Enabled Systems, Production ML Systems, AI-Enabled Systems, or ML-Infused Systems; SE4AI, SE4ML
    - AI Engineering/ML Engineering，MLOps，AIOps，ML Systems Engineering

智能 与 人工智能

- **人类历史上发明的各种计算器/机都是在追求智能**
  - 随着现代计算机在计算能力上超越人类，人们开始不再把计算等同于智能
- **人工智能一词出现于1956年的达特茅斯会议**
- **本课程观点：人工智能是企图了解智能的实质，并生产出一种可以按人类智能相似的方式做出反应的智能机器**



智能化软件概述

什么是软件?

具体的软件产品，如：Windows、Office、AutoCAD、微信app等

软件=程序+文档

什么是软件?

北京大学 PEKING UNIVERSITY

软件范型—

- 编程模型（what to be)
- 构造方法（how to do)
- 运行机理（how to operate)
- 质量保障（how well)

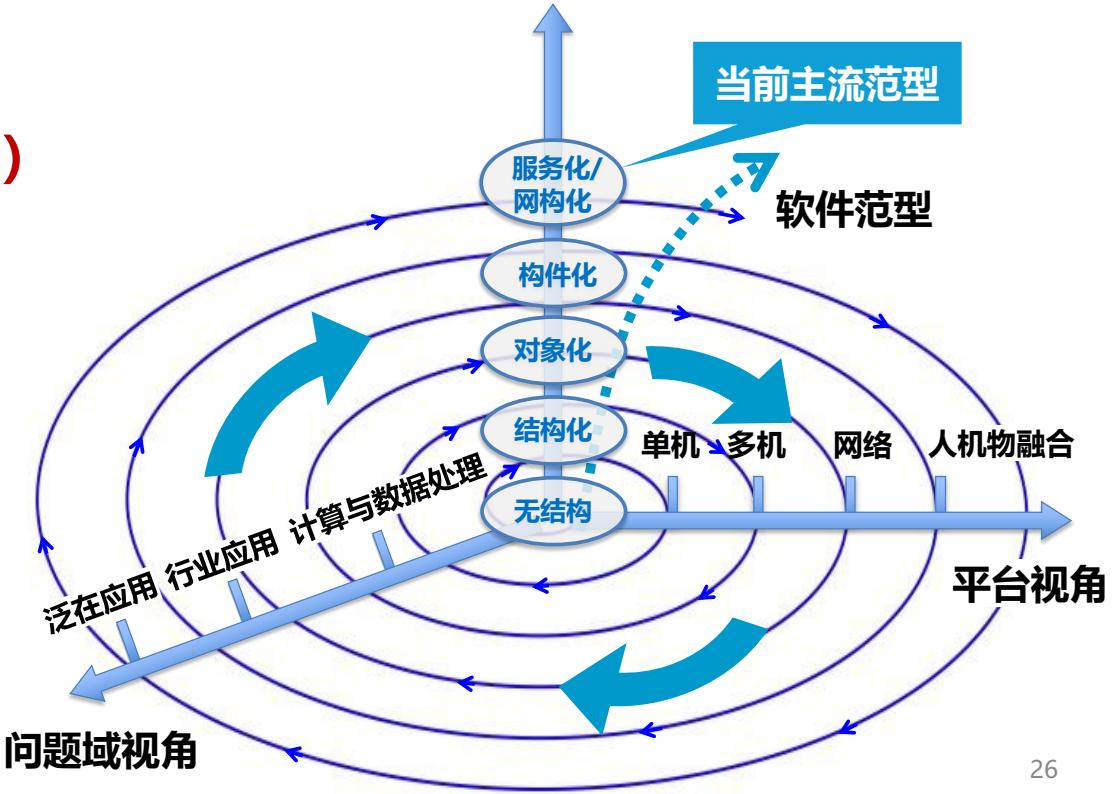
什么是软件?

北京大学 PEKING UNIVERSITY

软件范型—

- **编程模型 (what to be)**
- **构造方法 (how to do)**
- **运行机理 (how to operate)**
- **质量保障 (how well)**

每一轮范型变迁，均导致软件技术体系的一次螺旋式上升重构，产生重大技术突破



什么是软件?

图灵奖：以太网、万维网、浏览器、....


T. Berners-Lee
2016年图灵奖


R. Metcalfe
2022年图灵奖

图灵奖：TCP/IP、分布式系统....


R. Kahn
&
V. Cerf
2004年图灵奖


B. Liskov
2008年图灵奖


L. Lamport
2013年图灵奖

图灵奖：面向对象编程、数据抽象、个人计算、....


K. Nygaard
&
O-J. Dahl
2001年图灵奖


A. Kay
2003年图灵奖

图灵奖：C语言、UNIX 操作系统、软件工程....

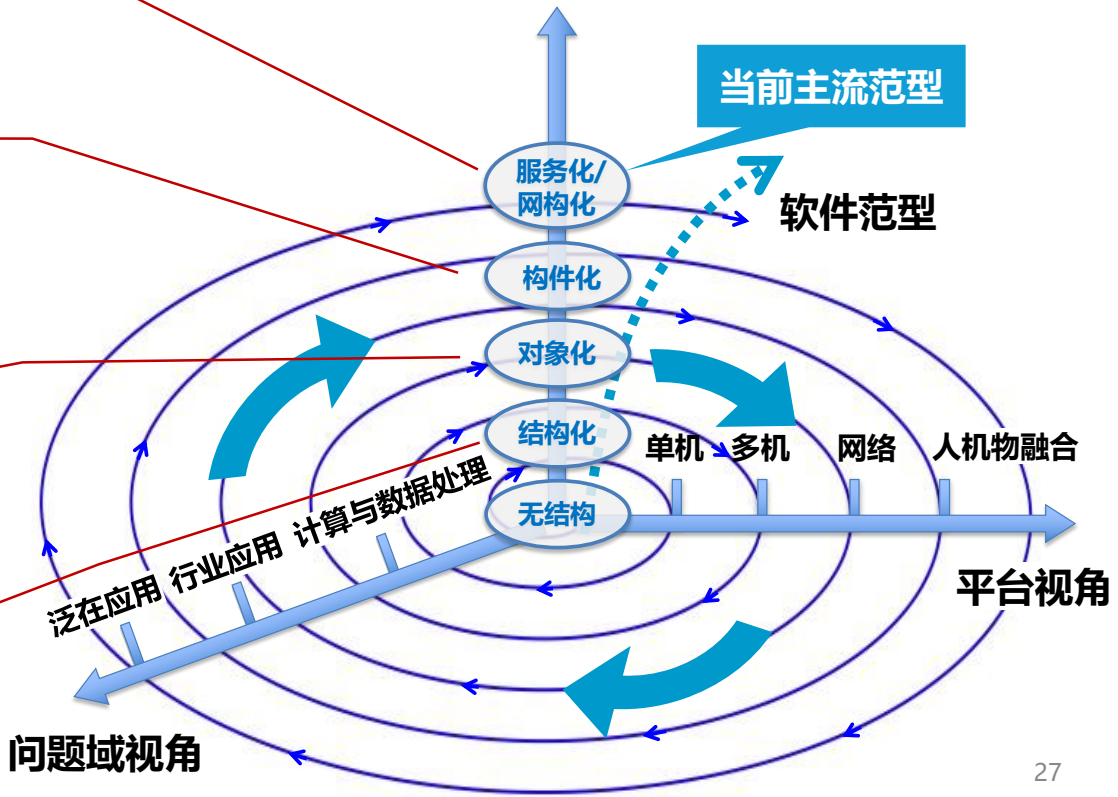

E. Dijkstra
1972年图灵奖


J. Backus
1977年图灵奖

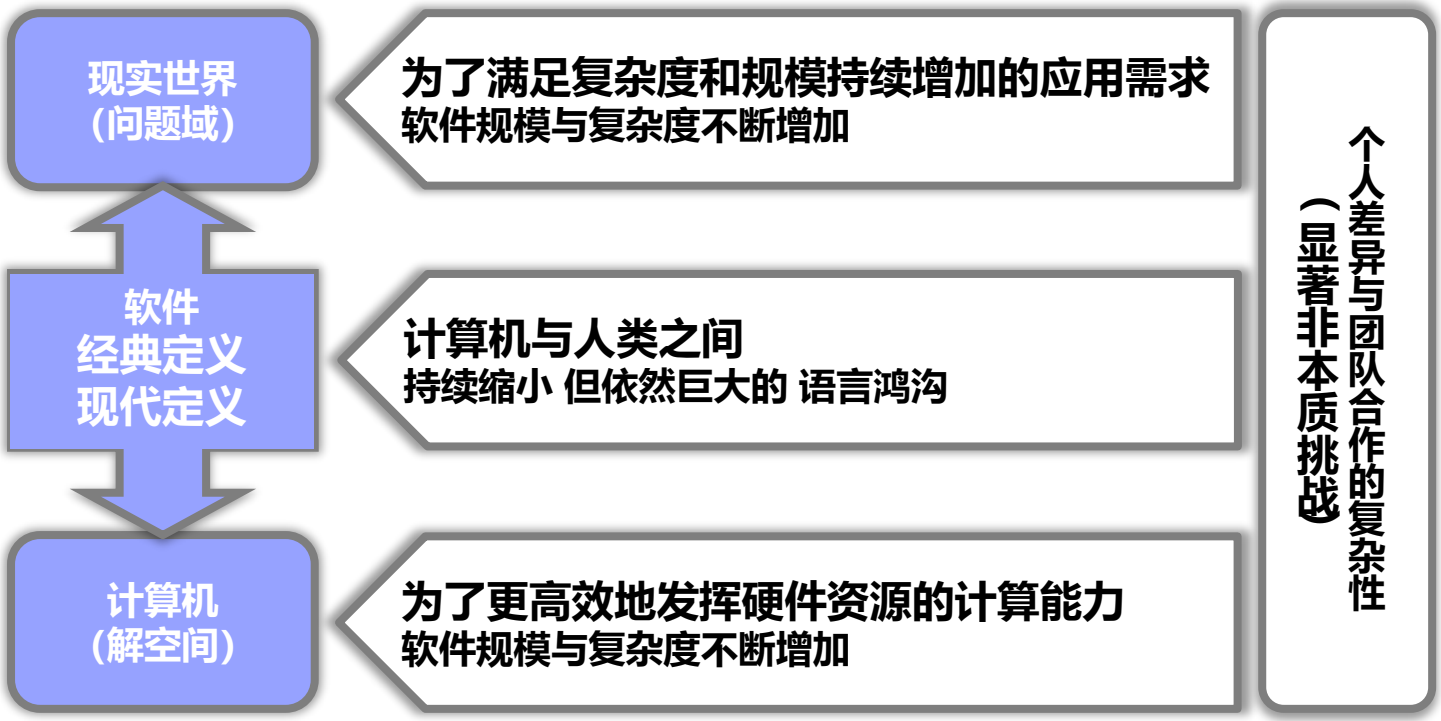

K. Thompson
& D. Richie
1983年图灵奖


N. Wirth
1984年图灵奖

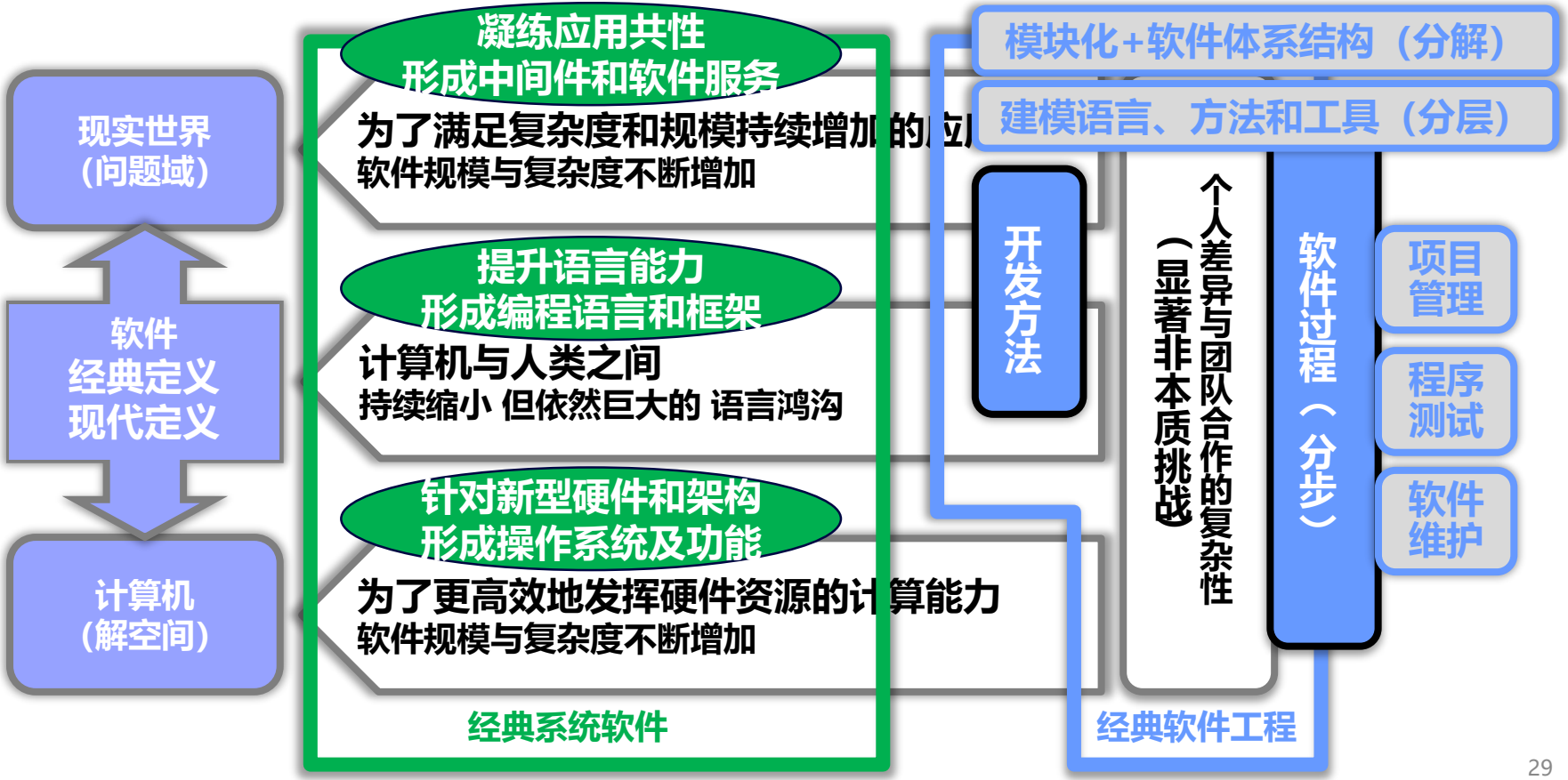
每一轮范型变迁，均导致软件技术体系的一次螺旋式上升重构，产生重大技术突破



软件开发为何难？　软件本质使然



软件开发为何难？ 传统应对之道

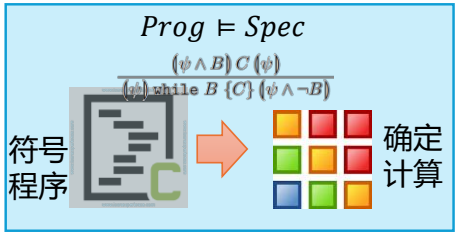


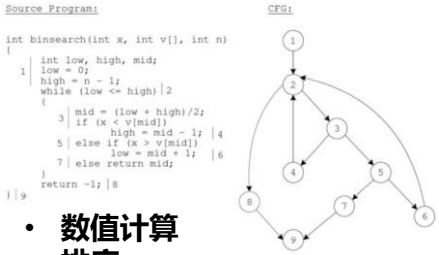
什么是智能化软件?

现有主流范型：均由人或机器辅助编写程序实现确定性计算过程和输出

由人或机器辅助人编写程序完成

代码部件





- **数值计算**
- **排序**
- **搜索引擎**

- **形态**：符号化的代码实现图灵机，数据作为处理对象
- **行为**：功能由可解释的程序代码描述的明确的计算过程实现

什么是智能化软件?

北京大学 PEKING UNIVERSITY

引入机器学习模型后，形态和行为发生重大变化

由人或机器辅助人编写程序完成

代码部件

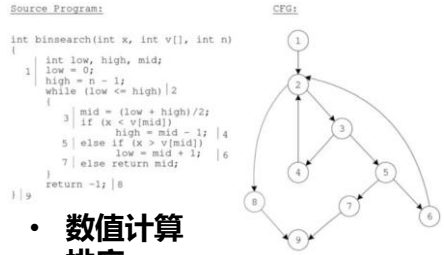
SW

符号程序

$Prog \vDash Spec$

$$\frac{(\psi \wedge B)\ C\ (\psi)}{(\psi)\ \text{while}\ B\ \{C\}\ (\psi \wedge \neg B)}$$

确定计算



- **数值计算**
- **排序**
- **搜索引擎**

- **形态**：符号化的代码实现图灵机，数据作为处理对象
- **行为**：功能由可解释的程序代码描述的明确的计算过程实现

Ai

通过机器学习自动生成

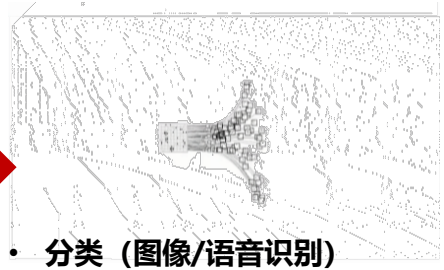
机器学习模型

$Y = f(X; W)$

$\arg\min_W \Sigma_i loss(Y_i, f(X_i; W))$

非确定计算

模型部件



- **分类（图像/语音识别）**
- **回归（预测）**
- **内容生成**

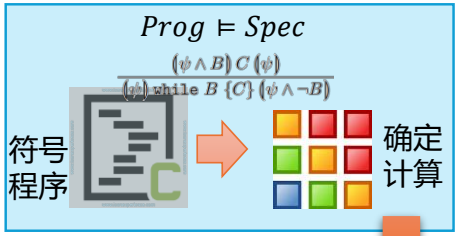
- **形态**：对神经网络的模拟，权重数据是关键，框架代码相对简单固定
- **行为**：功能由主要由不可解释的权重数据决定；更强的功能意味着更深的网络和更多权重数据

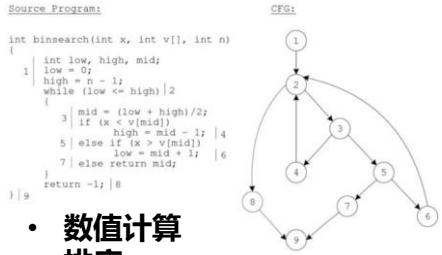
什么是智能化软件?

智能化软件的内涵：融合了程序员编写的确定性符号计算程序部件和机器学习生成的非确定性概率计算模型部件，具有复杂网络结构，并可能呈现涌现行为的软件系统

由人或机器辅助人编写程序完成

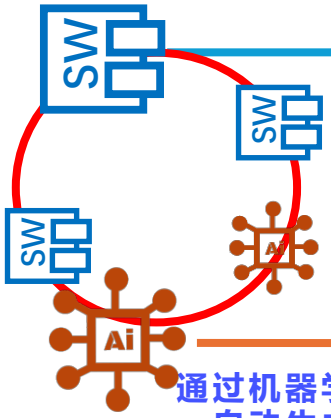
代码部件



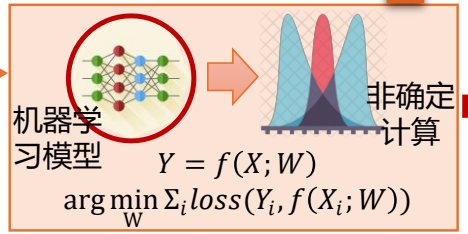


- **数值计算**
- **排序**
- **搜索引擎**

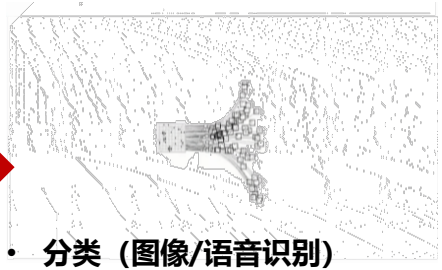
二者混合交织，协同工作



通过机器学习自动生成



模型部件



- **分类（图像/语音识别）**
- **回归（预测）**
- **内容生成**

- **形态**：符号化的代码实现图灵机，数据作为处理对象
- **行为**：功能由可解释的程序代码描述的明确的计算过程实现

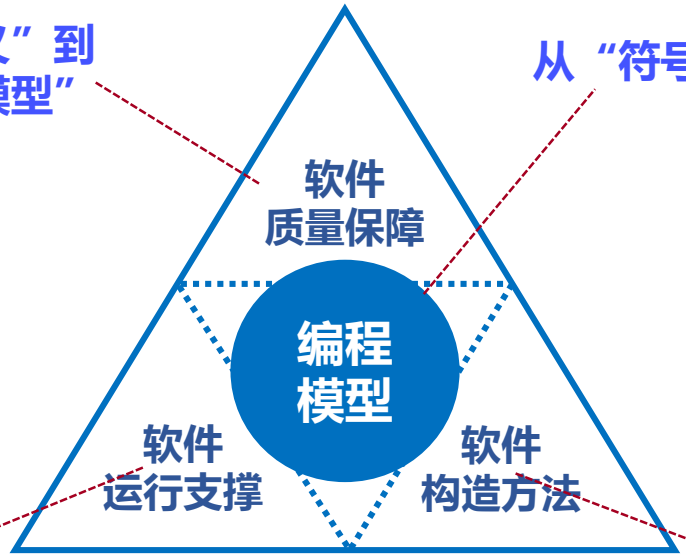
- **形态**：对神经网络的模拟，权重数据是关键，框架代码相对简单固定
- **行为**：功能由主要由不可解释的权重数据决定；更强的功能意味着更深的网络和更多权重数据

智能化软件为何开发更难？

软件范型四要素均在发生转变

从“确定性行为和可解释语义”到
兼有“非确定行为和难解释模型”

从“符号软件”到“神经-符号融合软件”



从“相对可控的软件个体及其交互”到
“复杂难控的自主协同和群智涌现”

从“知识驱动、人工开发”到
“数据/知识双驱动、自动合成

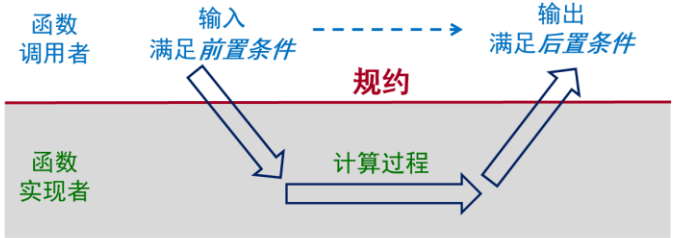
挑战1：缺少规约

北京大学 PEKING UNIVERSITY

- 传统软件工程通过规约明确要做什么以及如何测试

函数的规约

- 在给定规约下
  - 函数实现的正确性：对于所有满足前置条件的输入、函数的返回值和额外影响都满足后置条件的约定
  - 函数实现者的责任：根据规约来实现后置条件
    - 无需关心函数如何被调用，可以始终假设输入参数是有效
  - 函数调用者的责任：根据约定的前置条件来调用函数
    - 无需关心函数是如何实现的，可以始终假设函数的行为是正确的



6

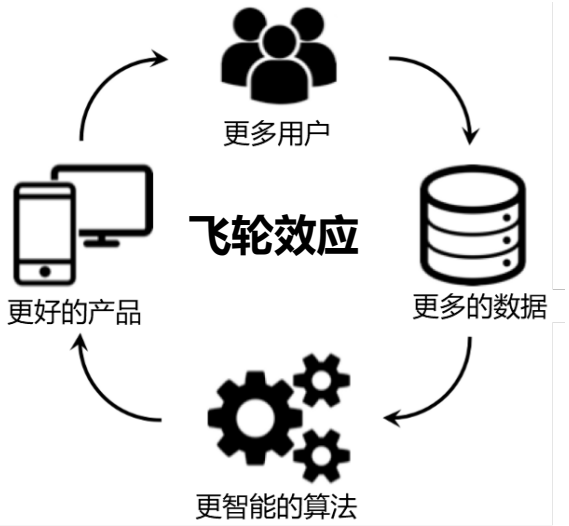
- 机器学习模型通常是黑箱，即便编写了规约，也无法确定模型是否会遵循这些说明

```
/**
 * compute deductions based on provided adjusted
 * gross income and expenses in customer data.
 *
 * see tax code 26 U.S. Code A.1.B, PART VI
 */
float computeDeductions(float agi, Expenses expenses);
```

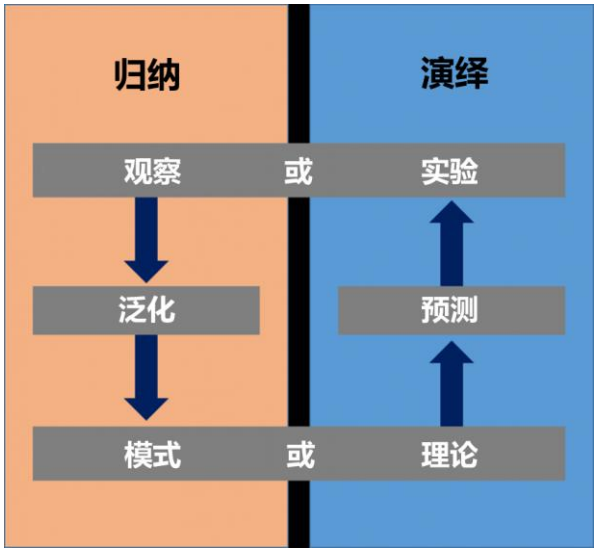
```
/**
  Return the text spoken within the audio file
  ????
*/
String transcribe(File audioFile);
```

挑战2：大规模数据

- 机器学习模型通过数据获取“规约”
  - 数据越多越好



- 从演绎推理（应用逻辑规则）=> 归纳推理（从观察中归纳）



导致可扩展性问题

挑战3：模型会出错

- **而且往往是以意想不到的方式出错**
- **由于没有明确的规约，这些错误很难预见和捕捉**
- **“正确” 意味着什么?**
  - 只能评估它在某些测试数据上是否表现得足够好（平均而言）
- **系统必须能够容忍一些错误的预测，并且要意识到自身可能会对世界产生怎样的影响**

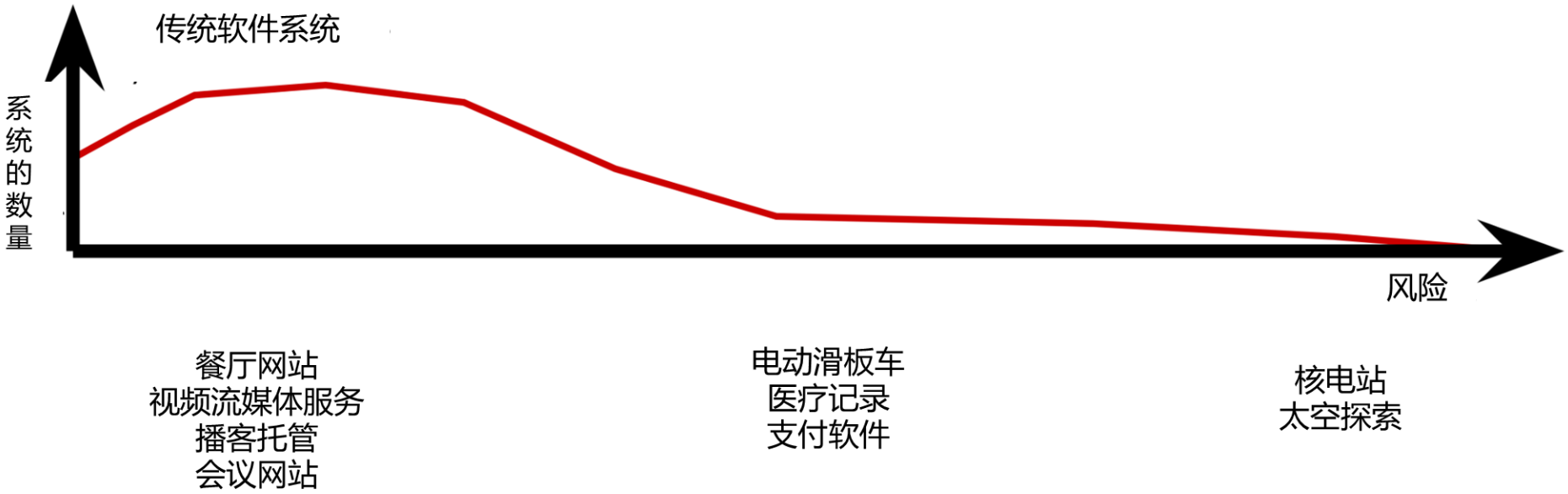
如何应对?

北京大学 PEKING UNIVERSITY

- **机器学习带来的挑战并非全新**
- **传统软件开发已有应对之道**
  - 开发包含不可靠模块的安全软件
  - 非机器学习的大数据系统、云计算系统
  - “足够好”且“符合用途”而非“绝对正确”的系统
- **机器学习只是加剧了这些挑战**

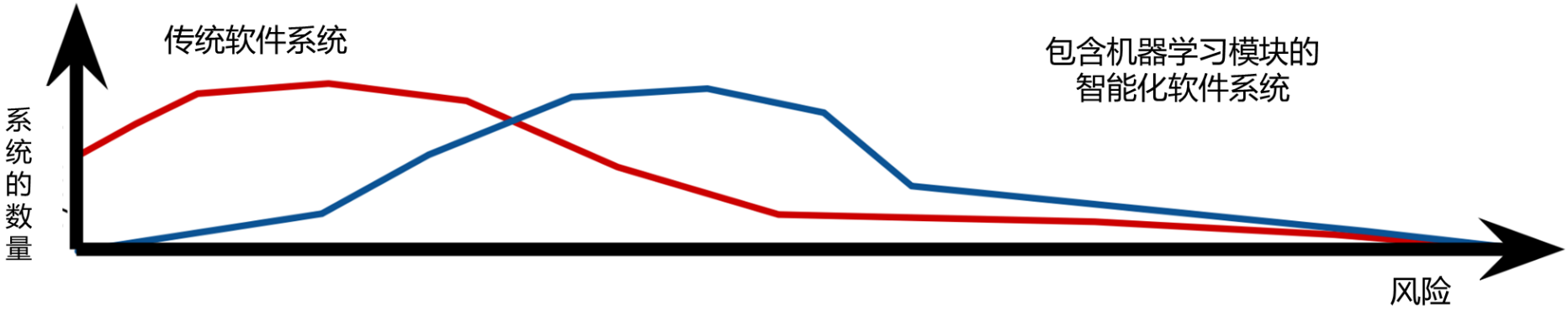
如何应对?

传统软件系统 到 智能化软件系统 的复杂性变化



如何应对?

传统软件系统 到 智能化软件系统 的复杂性变化



游戏人工智能
交通规划
照片滤镜

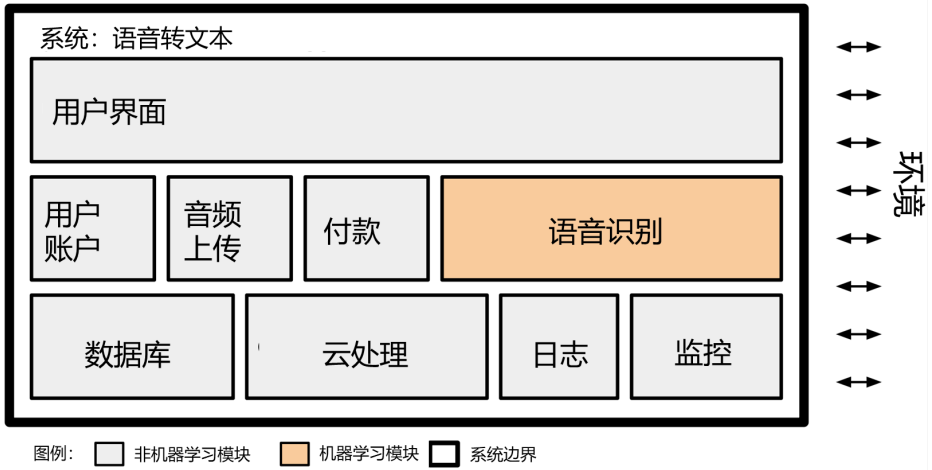
照片搜索
社交媒体推送
人行道机器人配送
智能家居设备
自动转录

自动驾驶车辆
预测性警务
深度伪造

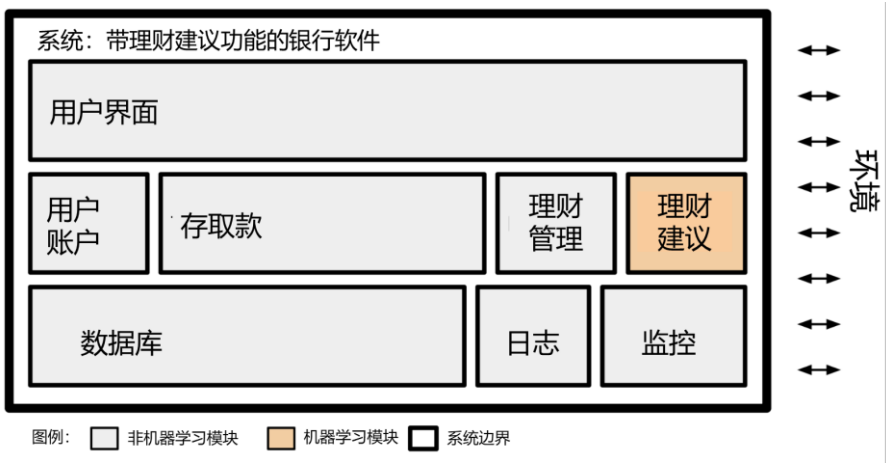
模型视角 vs 系统视角 的智能化软件

北京大学 PEKING UNIVERSITY

模型是系统的一个组成部分



机器学习模型作为系统的核心部件



机器学习模型作为系统的非核心部件

模型视角 vs 系统视角 的智能化软件

北京大学 PEKING UNIVERSITY

模型是系统的一个组成部分

- 淘宝的产品推荐
- 滴滴的定价计算
- 汽车中的自适应巡航控制
- 安卓（Android）系统的智能应用推荐
- 基于社交媒体数据的时尚趋势预测
- 疫情中的感染追踪与预测
- 手机运营商对网络问题的自适应应对
- 电脑游戏中基于技能的玩家匹配
- ……

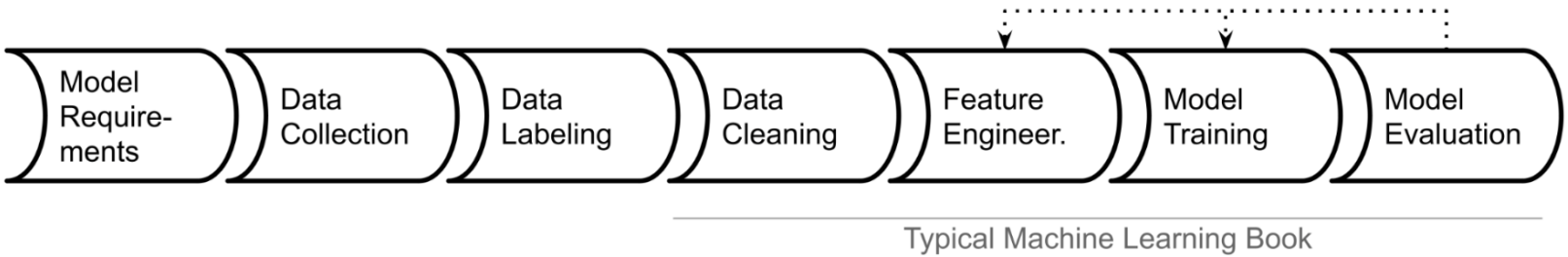
请举出同一个机器学习模型在不同的系统中分别作为核心和非核心部件的例子?

模型视角 vs 系统视角 的智能化软件

北京大学 PEKING UNIVERSITY

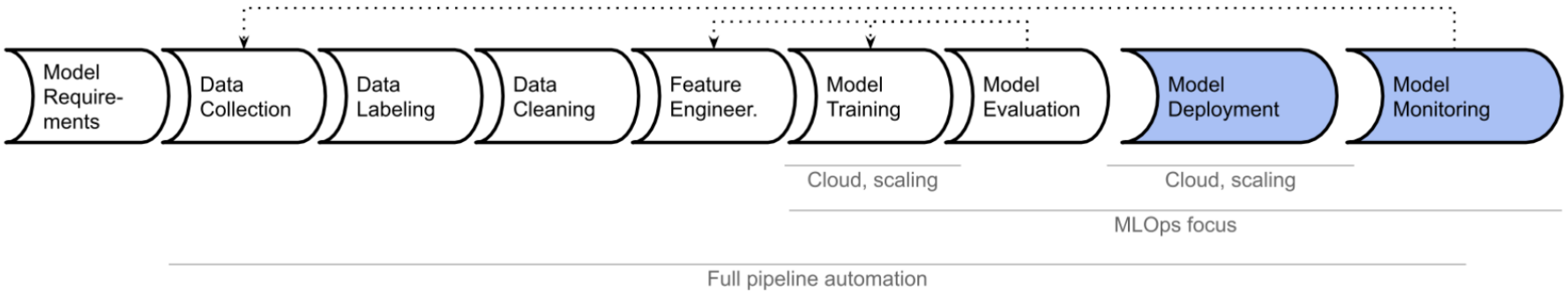
功能组成：传统的模型（数据科学）视角

- 关注：基于给定数据构建模型，评估准确性



模型视角 vs 系统视角 的智能化软件

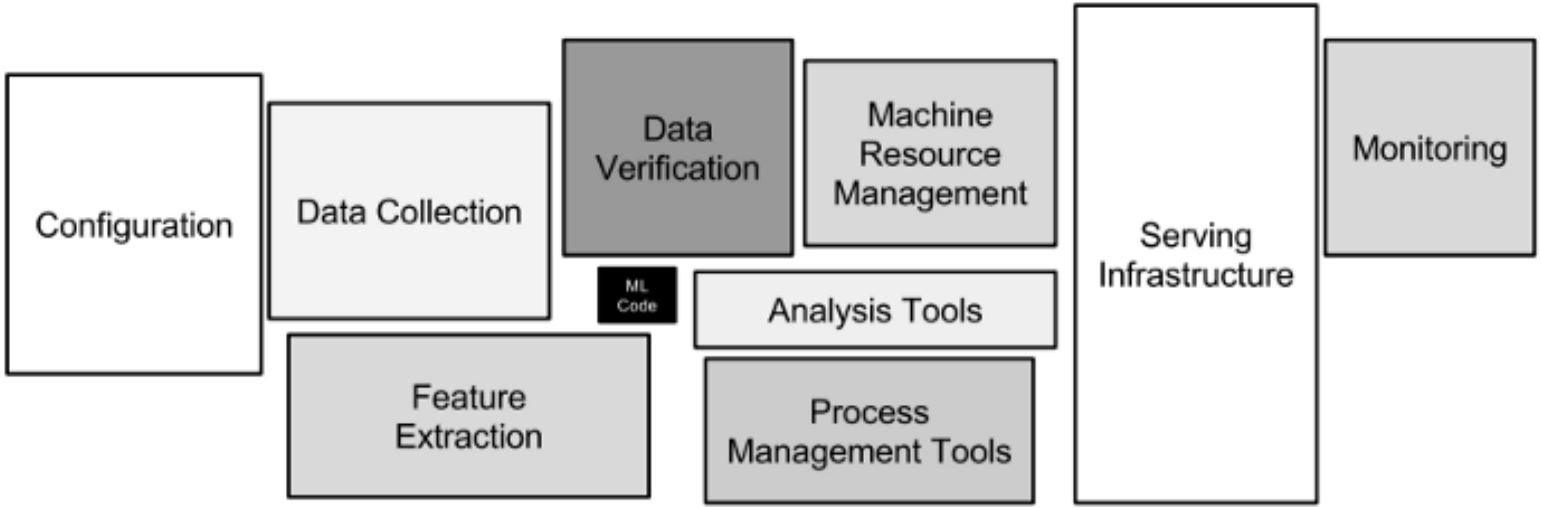
- **功能组成：自动化流水线与机器学习运维（机器学习工程）视角**
  - 关注：实验、部署、规模化的训练与服务，以及模型监控和更新



模型视角 vs 系统视角 的智能化软件

北京大学 PEKING UNIVERSITY

- **功能组成：自动化流水线与机器学习运维（机器学习工程）视角**
  - 机器学习的“技术债”

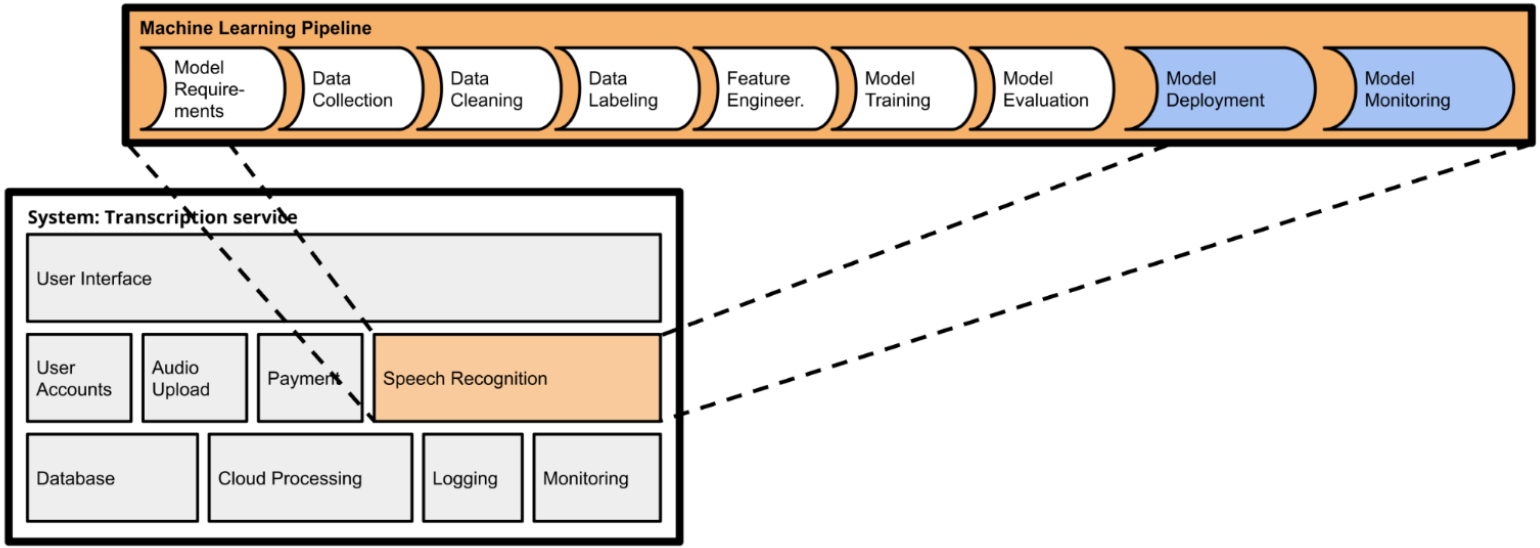


Sculley, David, et al. “Hidden technical debt in machine learning systems.” NIPS 2017.

模型视角 vs 系统视角 的智能化软件

功能组成：系统视角

- 智能化软件系统中，除了利用机器学习实现的功能，还有着更为复杂多样的其他功能实现，如机器学习部件与非机器学习部件的交互、系统需求、用户交互、安全性以及产品交付等



本讲小结

- **机器学习是智能化软件中的一部分**
- **数据科学家和软件工程师有不同的目标和重点**
  - 构建系统需要两者的合作
- **机器学习对软件技术带来了新挑战，并加剧了已有的问题**
- **开发智能化软件需要具备整体系统视角，而不仅仅局限于模型或机器学习流水线**

谢 谢

欢迎在填写问卷反馈

北京大学
PEKING UNIVERSITY