# AiU AI 认证测试工程师（CTAI）大纲

版本 1.01R 2019

人工智能联盟（AiU）

## 版权声明



## 感谢主要作者：

- Vipul Kocher, Saurabh Bansal, Srinivas Padmanabhuni and Sonika Bengani

## 感谢合著者：

- Rik Marselis and José M. Díaz Delgado

## 感谢评审委员会：

Albert Tort, Alfonso Fernández, Amit Dang, Ana Laura Ochoa Moreno, Andreas Hetz, Ángel Rayo Acevedo, Aurelio Gandarillas, Baris Sarialioglu, Björn Lemke, Christine Green, Daniel Garcia Castillo, Daniel Tolosa, Dario Iván Rosas Miranda, Durga Mohapatra, Emilie Potin-Suau, Erik van Veenendaal, Girish Nuli, Girts Baltaisbrencis, Guino Henostroza, Gustavo Márquez Sosa, Gustavo Terrera, Héctor Ruvalcaba, Javier Alejandro Chávez Crivelli, Jeff Nyman, Joel Oliveira, José Antonio Rodríguez Gómez, Juan Pablo Rios Alvarez, Julie Gardiner, Kimmo Hakala, Kristine Corbus, Kyle Alexander Siemens, Lorena Parra Rubio, Maarten-Jan van Gool, Manuel Fischer, Márton Görög, Maximiliano Mannise, Melissa Pontes, Miaomiao Tang, Michaël Pilaeten, Michel Dussouchaud, Nadia Soledad Cavalleri, Nora Alriyes, Paweł Noga, Petr Neugebauer, Ralf Pichler, Ram Shanmugam, Raúl Hussein Galindo, Richard Seidl, Sammy Kolluru, Samuel Ouko, Santiago de Jesús González Medellín,

Sebastia Małyska, Sergio Emanuel Cusmai, Serge Wolf, Shantel Yanique Stewart, Silvia Nane, Sunil Godse, Siva Prasad. B, Søren Wassard, Tariq King, Tetsu Nagata, Vikas Dhaka, Tom Van Ongeval, Werner Lieblang, Wim Decoutere, Yogesh Ahuja, Young jae Choi

## 修订历史

| 版本 | 日期 | 修订说明 |
|---|---|---|
| AiU A 2019 | 2019 年 3 月 2 日 | 首次测试版 |
| AiU B 2019 | 2019 年 5 月 8 日 | 二次测试版 |
| AiU 1.0R 2019 | 2019 年 7 月 16 日 | 首次发布 |
| 版本 1.01R 2019 | 2019 年 9 月 3 日 | 二次发布 |

## Table of Contents

## 业务成果

| | |
|---|---|
| BO-1 | 理解使用机器学习（ML）方法的人工智能（AI）的行业趋势。 |
| BO-2 | 比较机器学习（ML）的各种算法来帮助选择最合适的算法。 |
| BO-3 | 评估监督学习和无监督学习模型 |
| BO-4 | 设计和执行 AI 系统的测试案例 |
| BO-5 | 使用不同的方法将透明度引入模型工作 |
| BO-6 | 定义 AI 系统的测试策略 |
| BO-7 | 理解 AI 在手动测试和自动化测试中的使用场景 |
| BO-8 | 使用基于 AI 的测试执行工具自动执行测试 |

## 学习目标/知识认知水平

学习目标（LOs）是描述学习每个章节后应掌握知识的简短陈述。依照已修订的 Bloom 分类，学习目标定义如下：

- K1:记忆.动作动词如：记忆、回顾、选择、定义、查找、匹配、关联、选择
- K2: 理解. 动作动词如：总结、概括、分类、比较、对比、演示、解释、重述
- K3: 应用. 动作动词如：实现、执行、使用、应用

有关 Bloom 分类的更多详细信息，请参阅参考中的 [BT1] 和 [BT2]。

## 实际目标

实际目标 （HOs） 是描述为了理解学习的实际内容而应该执行内容的简短陈述。

实际目标定义如下:

- HO-0: 现场演示练习或录制视频
- HO-1: 指导性练习. 学员按照讲师执行的步骤顺序练习
- HO-2: 提示性练习. 学员按照讲师提供的提示进行练习
- HO-3: 无指导无提示性练习

## 前置条件

强制要求

- 无

建议要求

- ISTQB® 基础级别或者相当能力
- 编程语言基础知识 - Java/Python/R
- 统计学基础知识
- 软件开发或测试经验

# 第一章- 人工智能导论

## 关键字

人工智能、机器学习、监督学习、无监督学习、监督分类、监督回归、无监督聚类、无监督关联、强化学习、深度学习、神经网络、跨行业数据挖掘标准流程、机器学习生命周期

| | | |
|---|---|---|
| LO-1.1.1 | K2 | 解释人工智能（AI） |
| LO-1.1.2 | K1 | 回顾不同类型的 AI- 弱 AI、强 AI、超级 AI |
| LO-1.2.1 | K2 | 解释机器学习以及机器学习是实现 AI 的一种方法 |
| LO-1.2.2 | K2 | 解释监督机器学习和监督分类与监督回归的区别 |
| LO-1.2.3 | K2 | 解释无监督机器学习和比较无监督聚类和无监督关联 |
| LO-1.2.4 | K1 | 回顾强化学习的定义和应用 |
| LO-1.3.1 | K2 | 解释深度学习和神经网络的类型 |
| LO-1.4.1 | K2 | 解释跨行业数据挖掘标准流程的不同阶段-机器学习生命周期的流程 |
| LO-1.4.2 | K3 | 应用识别适当机器学习问题类型所涉及的步骤 |

## 1.1 人工智能(AI)

人工智能就是机器解决通常由人类解决的问题而获得的智能。

IBM Watson、微软 Cortana、苹果 Siri 和自动驾驶车辆是大量现有知名 AI 应用的一些示例。

### 1.1.1 人工智能定义 (AI)

| | | |
|---|---|---|
| LO-1.1.1 | K2 | 解释什么是人工智能 (AI). |

AI 是一门创造机器的艺术，用于需要执行人员智能执行的功能。[KUR] Al 在医疗保健、制造业、电子商务、零售、社交媒体、物流和其他行业领域发挥着主导作用。处理能力的提升、数据可用性和新的技术是促进 AI 应用提升的部分原因。AI 应用范围从监控房屋、决定股票投资、帮助选择食谱、甚至于帮助选择生活伴侣！

AI 是一个概括术语，它涵盖了使机器智能化的科学，无论是机器人、冰箱、电视、汽车、硬件还是软件部分。机器学习是 人工智能 的子集。机器学习 和 人工智能 经常互换使用，但它们不是一回事。

1.2 章节机器学习详细介绍了机器学习的内容。

### 1.1.2 人工智能的类型

| | | |
|---|---|---|
| LO-1.1.2 | K1 | 回顾人工智能的类型-弱 AI，强 AI，超级 AI |

AI 可以大致分为弱 AI，强 AI 和超级 AI.

- **弱 AI**: 指为特定环境中执行特定任务而编程的机器。如：游戏机器，语音助理和目前所有的 AI.
- **强 AI**: 具有一般认知能力的机器常被称为强 AI。这些 AIs 可以像人类一样推理和理解环境，并采取相应的行动。例如，常识推理。目前，强 AI 尚未实现，没有人知道什么时候或者它是否会变成现实。
- **超级 AI**:指能够复制人类思维，思想和情感的机器。超智能状态的机器比人类更聪明和睿智。考虑到当前 AI 发展的现状，超级 AI 不会很快实现。

## 1.2 机器学习(ML)

### 1.2.1 机器学习的定义

| | | |
|---|---|---|
| LO-1.2.1 | K2 | 解释机器学习以及机器学习是实现 AI 的一种方法 |

阿瑟·塞缪尔将机器学习定义为, “一个研究领域，使计算机能在不进行显式编程的情况下进行学习”。机器学习根据经验进行系统学习和改进，并随着时间的推移，基于之前的学习，完善可用于预测问题结果的模型。

除了机器学习之外, 人工智能还使用知识表达和推理来处理不同的场景。搜索、调度和优化的概念都属于人工智能的范围，但不一定是机器学习。

用于实现 AI 的技术包括：

- 机器学习(ML)
- 自然语言处理 (NLP)
- 机器人技术
- 语言处理
- 计算机视图

用于对机器学习算法分类的方法包括:

- 监督学习
- 无监督学习
- 强化学习

### 1.2.2 监督学习 - 分类和回归

| | | |
|---|---|---|
| LO-1.2.2 | K2 | 解释监督机器学习和监督分类与监督回归的区别 |

**监督学习:** 在监督学习中, 在训练阶段，要根据带有标记的样本数据进行模型学习。有标记的样本数据作为映射函数的培训师/主管，该函数推断出训练阶段输入数据和输出结果之间的关系。在检验阶段，该映射函数将应用一组新的数据来预测有标记的输出结果。一旦输出结果的准确度令人满意，则可以采用该模型。

监督学习能解决的问题被分成两类：

**分类:** 当问题需要将输入分类到几个预先决定的类之一时，将使用监督学习。在训练期间，当输出数据是离散的或输出数据落在类数之间时，使用此类模型。图像中的人脸识别或对象检测是可以使用分类的问题示例。分类的其它一些应用包括垃圾邮件检测（垃圾邮件或者无垃圾邮件）、基于 X 射线等疾病的诊断，通过驾驶员辅助系统正确识别路标等。一些常用的分类算法是逻辑回归、最近邻居、支持向量机和神经网络。

**回归:** 当输出结果是连续或者数字的性质，例如，预测一个人的年龄/体重，预测股票未来的价格等，使用回归学习。该类问题最常用的算法是线性回归，这是一种将输入数据和输出结果作为线性方程关系的简单算法。其它的一些算法是逻辑回归、支持向量机、套索回归等。

### 1.2.3 无监督学习 – 聚类和关联

| | | |
|---|---|---|
| LO-1.2.3 | K2 | 解释无监督机器学习和比较无监督聚类和无监督关联 |

**无监督学习:** 干净的，有标记的数据并非随时可用，因此某些问题需要在没有明确训练集合的情况下解决。这种没有提供显式标记的机器学习被称为无监督学习。这种问题的目标是学习没有关联标记输入数据的模式和结构。

根据输出的类型，无监督学习被进一步分为以下两种方法：

**聚类:** 这种无监督学习模型是按照一些共同特征或属性对输入数据进行分组。具有类似属性（未标记）的输入数据被分组在一个集群中。因此，输出是输入数据的集群。例如，在市场分析中的客户细分。

**关联:** 关联规则挖掘是在数据属性之间查找有趣的关系或者依赖关系。有趣的关联发现为市场决策提供了信息来源。例如，市场篮子数据分析，基于客户购物行为学习的产品推荐系统是基于关联规则建模的好例子。

### 1.2.4 强化学习

| | | |
|---|---|---|
| LO-1.2.4 | K1 | 回顾强化学习的定义和应用 |

它是一种机器学习（ML）的类型，代理（算法）通过和环境迭代式交互来学习，从而从经验中学习。当代理做出正确决定时将会得到奖励，做出错误决定时会被惩罚。基于奖励或惩罚的学习被定义为“强化学习”。为了实现预期的目标，为算法设置适当的环境，选择正确的策略以及设计奖励函数，是实施强化学习的关键挑战。机器人、自动驾驶车辆和聊天机器人是可以使用强化学习的应用实例。

## 1.3 深度学习 (DL)

### 1.3.1 深度学习和神经网络的类型

| | | |
|---|---|---|
| LO-1.3.1 | K2 | 解释深度学习和神经网络的类型 |

深度学习指的是系统从海量数据集中获得经验。深度学习使用人工神经网络分析大型数据集, 例如，自动驾驶车辆、大型文本处理和计算机视觉应用系统等。[AG1]

深度学习是机器学习的子集，机器学习是人工智能的子集。深度学习使用和机器学习相同的学习类型（监督学习、无监督学习和强化学习）。

**人工神经网络（ANN）：** 人工神经网络的灵感来自于人类大脑的结构。‘神经元’作为人工神经网络的基础单元，作用于输入刺激产生输出信号。输入通过激活函数层来产生输出。这些图层形成网格状网络。

每个人工神经网络至少有两层-输入层和输出层。这两层直接的所有层都称为隐藏层。一些不同类型的神经网络是:

**深度神经网络（DNN）：** 深度神经网络是有两层甚至更多隐藏层的人工神经网络（ANN）。

**卷积神经网络 (CNN) ：** 卷积神经网络是一种人工神经网络，它产生于对人脑视觉皮层的研究，自 20 世纪 80 年代以来被用于图像识别。与其它神经网络不同，卷积神经网络直接处理输入图像，不要序列化/矢量化输入数据并通过过滤器提取要素。卷积神经网络提供图像搜索服务，自动驾驶车辆，自动视频分类系统等。

**循环神经网络 (RNN):** 这些人工神经网络可以预测时间序列问题的未来。它们遵循任意长度的输入数据序列方法，而不是像其它神经网络那样固定长度的输入。每个输入和输出都独立于所有其它层。输出层的反馈被反复馈送到

同一网络，直到达到一定的置信区间水平。循环神经网络能够分析时间序列数据比如股票价格，并告诉您何时买入及何时卖出。在自动驾驶车辆中，他们会预测轨迹帮助避免事故。

## 1.4 机器学习流程的各个阶段

典型的机器学习项目遵循数据挖掘(CRISP-DM)框架的跨行业标准流程的所有阶段-行业标准和灵活的框架。

### 1.4.1 机器学习流程的各个阶段 - CRISP-DM 流程

| | | |
|---|---|---|
| LO-1.4.1 | K2 | 解释跨行业数据挖掘标准流程的不同阶段-机器学习生命周期的流程 |

CRISP-DM 在数据挖掘生命周期中通常有六个步骤。为了满足机器学习项目要求，通过增加第七步骤对其进行自定义。

机器学习 CRISP-DM 框架的七个步骤是：[DSCl] [SMU]

1. **数据采集**:从所有外部和内部源收集数据 (例如数据库, CSV 文件, 社交媒体等.)
2. **数据准备**: 清洗并重塑原始数据. 使用特征工程创建新属性，这是一个从现有数据创建新变量的过程。降维，数据插补，缺失值的空值处理等都是数据准备中的一些方法。
3. **建模**: 选择模型或者算法，将可用数据分为训练数据集和测试数据集。模型是通过在训练数据集上执行机器学习算法获得的。使用测试数据集来评估和增强模型的性能直到得到令人满意的性能。
4. **评估**: 根据各种指标（在 3.2 指标中讨论）评估模型，并在最终部署前对其建立基线。
5. **部署**: 部署和监控在生产环境中指标的基准模型。
6. **操作**: 定期进行维护和操作。当指标低于特定阈值是，重新生成和优化模型。

7. **优化**: 由于概念漂移（6.1.3 概念漂移风险），随着更好的算法可用，或者一些性能上的重大问题，已部署的解决方案可能会被替换。

步骤 1~4 被归类为脱机阶段的一部分，其输出是经过训练的模型。

步骤 5~6 是联机阶段的一部分，将在脱机阶段训练的模型与系统的其余部分集成并部署在生产环境中。优化步骤是重新执行步骤 1~6.

## 1.4.2 识别机器学习问题类型的步骤

| | | |
|---|---|---|
| LO-1.4.2 | K3 | 应用识别相应机器学习问题类型所涉及的步骤 |

理解我们正在努力解决的问题以及解决这些问题所需的学习类型是非常重要的。下面将讨论一种识别机器学习（ML）问题类型的方法：

1. 如果问题涉及多个状态的概念，并且涉及在多个状态中移动，那么探索 RL。
2. 如果存在输出变量，那么是监督学习。
   2.1. 在输出是离散和分类的情况下，这是一个分类问题。
   2.2. 在输出在性质上是数值和连续性的，则她是一个回归问题。
3. 如果给定数据集中未提供输出，那么探索无监督学习。
   3.1. 如果问题涉及对数据进行分组，则它是一个聚类问题。
   3.2. 如果问题涉及查找共同发生的数据项，则应用关联规则挖掘。
   3.3. 如果原始数据是非结构化的，可以使用深度学习算法自动探索提取要素。

上述步骤的先决条件是，有足够的数据可用于分析适当的机器学习问题类型。

# 第二章 - 测试人工智能系统概述

**关键字:** 离线阶段, 在线阶段, ISO25010

| | | |
|---|---|---|
| LO-2.1.1 | K2 | 解释在训练（离线）阶段和训练后集成（在线）阶段在人工智能-机器学习系统中执行的测试。 |
| LO-2.2.1 | K2 | 将人工智能系统的测试和非人工智能系统的测试进行比较。 |
| LO-2.3.1 | K1 | 回顾 ISO 25010 质量属性，尤其是功能适用性、性能、可靠性、可维护性和其它参数，如复杂性、可扩展性和持续学习，作为参数用于评估经过训练过的模型。 |
| LO-2.3.2 | K1 | 回顾 ISO 25010 提到的 AI 测试特有的质量属性-智能行为、道德和个性。 |

## 2.1 AI 测试阶段

### 2.1.1 AI 系统的离线测试和在线测试

| | | |
|---|---|---|
| LO-2.1.1 | K2 | 解释在训练（离线）阶段和训练后集成（在线）阶段在人工智能-机器学习系统中执行的测试。 |

机器学习生命周期（如章节 1.4 机器学习流程的各个阶段中描述的）被分为两个阶段：离线和在线。在这些阶段执行的不同测试类型是

**离线阶段测试:** 在此阶段，将测试被训练过的模型。各种指标被用来评估训练模型的参数，以验证其达到目标的程度。监督学习和无监督学习有不同的参数。通常，将测试模型的功能属性，但不测试非功能属性。由于模型部署环境和训练环境不一样，因此在此阶段对模型进行性能测试没有太大意义。模型训练时间是作为非功能参数训练的参数。对于需要频繁训练的模型，这是一个重要的参数.离线测试在第三章节-人工智能系统的离线测试中介绍。

离线测试的另外一个重要方面是能否解释模型的行为。可以使用不同的方法和算法。第五章可解释的 AI 中介绍了测试这一方面。

**在线测试阶段:** 在该阶段，将测试训练模型和其它系统的集成，包括所有 AI 和非 AI 的组件。将会执行功能测试和非功能测试如性能测试。

如果系统的输入可以是非文本、非结构化输入，因此根据所使用的工具，对于机器学习部件相关的某些测试的自动化支持可能会受到限制。

在线测试将在第四章节-AI 系统的在线测试中介绍。

## 2.2 人工智能与非人工智能测试

### 2.2.1 AI 系统测试 vs. 传统系统测试（非 AI）

| | | |
|---|---|---|
| LO-2.2.1 | K2 | 将人工智能系统的测试和非人工智能系统的测试进行比较。 |

- AI 系统的测试准则不易获得
- 与传统测试不同，AI 系统的测试输出是非确定性的. 因此，AI 系统的输出具有概率，测试案例的结果也是概率而不是有个明确的成功/失败。
- AI 系统的逻辑是从用于训练模型的数据中产生的。该逻辑不能由于检查，尤其是神经网络。这使得很难理解为什么产生特定的输出。一个正确的或者期望的结果不能保证正常工作。
- 离线阶段的测试是一个额外的阶段，需要专门的技能和技术，如数据清洗、预处理和测试训练模型。
- 在线测试阶段需要对 AI 系统如何工作的深度理解。AI 系统和其它 AI 系统或者非 AI 系统的集成增强了不同测试技术的需求。
- 与非 AI 系统相似，需要 AI 系统中执行功能测试和非功能测试。
- 在线阶段的测试可以作为正常黑盒系统和系统集成测试执行，而不用担心集合中是否有一个或者多个人工智能组件。

## 2.3 AI 质量属性

### 2.3.1 评价 AI 系统的质量属性

| | | |
|---|---|---|
| LO-2.3.1 | K1 | 回顾 ISO 25010 质量属性，尤其是功能适用性、性能、可靠性、可维护性和其它参数，如复杂性、可扩展性和持续学习，作为参数用于评估经过训练过的模型。 |

AI 系统的质量属性可以基于 ISO 25010 质量属性和其它质量属性组合来评价。从 AI 测试角度来看，一些重要的质量特征包括

- **功能适用性** -功能正确性，完整性和适用性 。
- **可靠性**
  - 可用性- 系统在正常操作期间的可用性
  - 容错 - 系统处理损害的、不完整或者不相干数据而不发生故障的概率。
- **性能效率**
  - 时间行为 - 系统对来自她的数据的相应速度有多快。
  - 资源利用率 - 系统使用哪些资源或者多少资源来执行功能。
- **可维护性**
  - 可分析性 - 评估一个或多个部件的预期变更对产品或者系统影响的有效性和效率程度，或者诊断产品的缺陷和失效原因，或者识别被修改的部件。在 AI 的情况下，可分析性还指能够理解为什么系统能够做出决策的能力。理想情况下，可解释性（可检性）应该是 ISO 25010 关于 AI 系统的单独质量属性。
  - 可测试性 -在给定的上下文中，人工智能组件支持测试的程度。支持程度（可测试性）越高，通过测试发现缺陷越容易。

其它重要的参数：

- **复杂性** – 时间和空间负责性
- **可扩展性** – 系统通过增加资源来处理更多负载的能力

- **持续学习**– 系统具备从新数据，尤其是真实环境中的数据持续学习的能力

### 2.3.2 基于 AI 的扩展质量属性

| | | |
|---|---|---|
| LO-2.3.2 | K1 | 回顾 ISO 25010 提到的 AI 测试特有的质量属性-智能行为、道德和个性。 |

使用 AI 需要扩展标准质量属性。

除了 ISO 25010 中提到的，智能机器还应该具备下列质量属性。[TDA]

- 智能行为– 智能行为是理解的能力或理解。它基本上是推理、记忆、想象和判断的结合；其中每一个都依赖于其它。智能是认知技能和知识的结合，通过适应性行为来体现。子属性有：学习能力、改进能力、选择透明度、协作和自然互动。
- 道德 – 道德, 就人工智能而言, 是关于区分对错，或者善恶的规则。子属性有：伦理、隐私和人性。
- 个性–个性是形成个人独特个性的特征或品质的组合。子属性是：情绪、移情、幽默和魅力。

## 第三章 - AI 系统的离线测试

**关键字:** 结构化数据、非结构化数据、维度、指标、交叉验证、拟合不足、过度拟合、分析

| | | |
|---|---|---|
| LO-3.1.1 | K1 | 回顾数据准备和预处理所包含的步骤和全面清理数据的必要性。 |
| LO-3.1.2 | K3 | 应用数据的读取和操作，包括结构化数据的筛选步骤。 |
| HO-3.1.2 | H2 | 使用代码（选择的语言）从各种数据源（如 Excel 文件，CSV 文件或数据库）读取数据，并通过删除最不重要的列，添加列和清理结构化文本数据（数据操作）来清理给定的数据集。 |
| LO-3.1.3 | K2 | 总结非结构数据（图像）的各种数据预处理步骤 |
| LO-3.1.4 | K2 | 总结非结构数据（文本）的各种数据预处理步骤 |
| HO-3.1.4 | H1 | 执行非结构数据（文本）数据预处理步骤 |

| | | |
|---|---|---|
| LO-3.1.5 | K3 | 根据不同的类型的问题使用不同的数据操作方法。 |
| HO-3.1.5 | H3 | 使用均值、模式、KNN 方法填充给定数据集的缺失数据值。 |
| LO-3.1.6 | K1 | 列出用于数据可视化的各种类型的绘图。 |
| LO-3.1.7 | K2 | 解释异常值的概念及其存在的各种原因。 |
| LO-3.1.8 | K3 | 应用可视框绘图方法来确定给定数据集的异常值。 |
| HO-3.1.8 | H2 | 为给定数据集画箱线图来标识异常值。 |
| LO-3.1.9 | K3 | 应用降维技术-无关特征消除，主成分分析（PCA）。 |
| HO-3.1.9 | H3 | 执行不相干特征消除，在给定数据集执行 PCA 降维。 |
| LO-3.2.1 | K2 | 解释度量指标在 ML 系统中作用。 |
| LO-3.2.2 | K1 | 列出监督学习和无监督学习的各种模型评估参数。 |
| LO-3.2.3 | K2 | 比较惯性和调整过的 Rand 系数作为无监督聚类的指标。 |
| HO-3.2.3 | H3 | 应用惯性-簇内平方和（WCSS），并用肘形法确定最佳簇数。 |
| LO-3.2.4 | K2 | 比较无监督关联规则挖掘的支持、置信度和提升指标。 |
| HO-3.2.4 | H3 | 计算无监督关联规则挖掘的支持、置信度和提升指标。 |
| LO-3.2.5 | K2 | 解释混淆矩阵 |
| LO-3.2.6 | K3 | 应用公式计算监督分类的准确度、精密度、召回率、特异性和 F1 得分指标，并进行比较。 |
| HO-3.2.6 | H3 | 计算监督分类的准确度、精密度、召回率、特异性和 F1 得分指标。 |
| LO-3.2.7 | K3 | 应用公式计算监督回归的各种指标（根均方误差（RMSE）和 R 平方） |
| HO-3.2.7 | H2 | 计算监督回归的根均方误差 RMSE 和 R 平方。 |
| LO-3.3.1 | K1 | 通过将可用数据集拆分为为训练集，验证集和测试集来定义验证模型的需求。 |
| LO-3.3.2 | K2 | 总结拟合不足，过度拟合和偏差方差权衡的概念。 |
| HO-3.3.2 | H0 | 演示拟合不足和过度拟合来显示偏差方差权衡。 |
| LO-3.3.3 | K3 | 应用拆分测试、K-折交叉验证、bootstrap 和留一法交叉 |

| | | |
|---|---|---|
| | | 验证方法。 |
| HO-3.3.3 | H1 | 在算法中应用 K-折交叉验证方法并比较各种结果。 |
| LO-3.4.1 | K1 | 回顾四种分析方法的特点-描述性的、探索性的、预测性的、规范性的-以及他们在 ML 中的使用。 |

# 3.1 数据准备和预处理

## 3.1.1 数据准备和预处理步骤

| | | |
|---|---|---|
| LO-3.1.1 | K1 | 回顾数据准备和预处理所包含的步骤和全面清理数据的必要性 |

输入数据可以是数据库表，CSV 文件（逗号分隔值）的形式，也可以是非结构化数据，如图像、音频、视频或正在运行的文本。所需数据来自外部和内部的各种来源。

数据采集后，需要经过彻底的清洗和一些处理，才能提供给算法进行训练和测试。

以下步骤用于数据准备和预处理。

- 数据操作
- 数据筛选
- 数据预处理活动包括
    - 数据处理，如处理缺失值
    - 数据可视化，获取全局视图和处理异常和异常值
    - 相关分析和降维

需要执行几种数据格式（例如图像、文本）特定的预处理步骤，以使数据格式适合训练。当数据量过大时，执行数据缩减而不会丢失信息。当结构化数据丢失，需要补充数据值。所有的数据预处理步骤都是为了在模型中获得期望的精确度和更好的可预测性。

## 3.1.2 数据准备

| | | |
|---|---|---|
| LO-3.1.2 | K3 | 应用数据的读取和操作，包括结构化数据的筛选步骤。 |
| HO-3.1.2 | H1 | 使用代码（选择的语言）从各种数据源（如 Excel 文件，CSV 文件或数据库）读取数据，并通过删除最不重要的列，添加列和清理结构化文本数据（数据操作）来清 |

| | | |
|---|---|---|
| | | 理给定的数据集。 |

数据准备包括:

1. **数据操作**: 更改给定数据的结构，例如：增加新列，删除一些行等。
2. **数据筛选**: 缩小结构化数据（表和矩阵）和非结构化数据（图像和文本）的大小，以提升数据质量。

### 3.1.3 处理非结构数据（图像）

| | | |
|---|---|---|
| LO-3.1.3 | K2 | 总结非结构数据（图像）的各种数据预处理步骤。 |

消除图像中的噪点并调整图像大小是设计计算机视觉算法对图像处理的常见操作。【UDI】

### 3.1.4 处理非结构数据（文本）

| | | |
|---|---|---|
| LO-3.1.4 | K2 | 总结非结构数据（文本）的各种数据预处理步骤 |
| HO-3.1.4 | H1 | 执行非结构数据（文本）数据预处理步骤 |

根据 ML 模型的需要，文本数据预处理可以通过多个语法更改步骤完成。例如，移除数字、将字母大写转换为小写、删除标点符号、空白、移除停止符、执行词干分析和流化等。【UDT】

### 3.1.5 数据填充

| | | |
|---|---|---|
| LO-3.1.5 | K3 | 根据不同的类型的问题使用不同的数据操作方法。 |
| HO-3.1.5 | H3 | 使用均值、模式、KNN 方法填充给定数据集的缺失数据值。 |

从字段中收集的数据可能会空值和缺失值，需要用相应的值来替换空值. 空值或者缺失值可以归纳为中心趋势（均值，中位数或模式）、K 最邻近的方法或者基于回归的方法。

### 3.1.6 数据可视化

| | | |
|---|---|---|
| LO-3.1.6 | K1 | 列出用于数据可视化的各种类型的绘图。 |

可视化数据有助于理解其数据结构和属性直接的关系，而今查看提供的数字或文本是不可能的。有各种类型的可视化。最常用的可视化方法包括：

连续值的折线图、离散值的直方图、箱线图、条形图、饼图等。它们可以在第一次使用时就提供关于可用数据有意义的见解。

**绘图类型：**
**单变量**: 最简单的分析形式，其中要分析的数据是单个变量。例如，人口的年龄或者人口的体重等。他们单独分析，且他们之间的关系从来没有被考虑。折线图、直方图、频率分布、条状图和箱线图用来分析单变量数据。

**双变量：** 进行此类分析是为了发现给定数据集中两个变量之间的关系。在 XY 平面上绘制一个变量与另一个变量，有助于发现两个变量之间的第一手关系。例如，考虑人口年龄和体重的关系。对于此类分析，可以使用散点图或者相关图。

**多变量**: 三个或者更多变量的分析。网格图和 3D 绘图是可视化多变量数据并从中发现它们之间关系的一些方法。

## 3.1.7 异常/异常值检测

| | | |
|---|---|---|
| LO-3.1.7 | K2 | 解释异常值的概念以及存在异常值的各种原因。 |

**异常值的原因：**

- 错误: 在这种情况下，异常值是测量，数据输入和采样中的错误结果，例如：大多数记录（以摄氏度为单位）的温度数据，少部分其它记录（以华氏度为单位）的错误数据。
- 自然: 一些异常值可能出现在在自然情况下，例如，如果洪水灾害 100 年发生一次，则它是自然异常值。
- 故意: 为验证检测方法而造的虚拟异常值，例如，用于测试边界案例的实验室培育方案的人工记录。

## 3.1.8 异常值检测技术

| | | |
|---|---|---|
| LO-3.1.8 | K3 | 应用可视框绘图方法来确定给定数据集的异常值。 |
| HO-3.1.8 | H2 | 为给定数据集画箱线图来标识异常值。 |

在使用给定数据集计算各种统计信息是，通过可视箱线图观察数据分布很有帮助。箱线图有助于确定数据集的异常值位置。

箱线图的两端是上四分位数和下四分位数。箱线图外的两条线是上线和下限，是扩展的最高或者最低数据阈值，数据点一旦超出这两条线则被认为是异常值。

### 3.1.9 降维

| | | |
|---|---|---|
| LO-3.1.9 | K3 | 应用降维技术-无关特征消除，主成分分析（PCA）。 |
| HO-3.1.9 | H3 | 执行不相干特征消除，在给定数据集执行 PCA 降维。 |

机器学习问题通常有大量的输入特性，但并不是所有这些特性都有助于分类或者回归输出。输入特性越多，训练集越难可视化。减少被考虑变量数量的技术，称为降维。
使用比原始维度更少维度的需要来自:

- 成本和速度因素
- 内存需求
- 避免冗余
- 识别数据中相关性最强的部分，以便进一步处理

**降维的方法：**

- 不相关特性消除是移除对输出变量没有影响的列:
  - 统一分析有助于删除其值不会跨行更改的列。例如，考虑来自单个零售商店的销售记录数据，然后删除诸如商店名称、商店地址等不会跨行更改的列。
  - 过低密度的数据容易在被调查后删除以节省空间。例如，数据库行号的列值对于每一行都有唯一值，应该被删除。
- 双变量分析在高度相关的输入属性对中删除一个（因此，也称为关联分析），例如，在商店库存数据中，“物料价格”和“物料数量”是高度相关的属性。

- 主变量分析: PCA 可大幅度降低大数据集的维度，但会最大限度地保留信息。它推导出一组新的独立变量（称为主组件），并将其按重要性高低的顺序排列。然后可以选择所需的最重要主组件数（因此，降低变量或维度的数量），但仍保留原始数据集的最大可能信息。

## 3.2 度量指标

### 3.2.1 度量指标的作用

| | | |
|---|---|---|
| LO-3.2.1 | K2 | 解释度量指标在 ML 系统中作用。 |

度量指标是训练模型的评估参数，可以视为对训练模型交付准确和可靠结果的测量。

### 3.2.2 监督学习和无监督学习的度量指标

| | | |
|---|---|---|
| LO-3.2.2 | K1 | 列出监督学习和无监督学习的各种模型评估参数。 |

监督学习和无监督学习的问题目标是不同的。因此，评估模型的度量指标也是不同的。

| 学习类型 | 模型类型 | 度量指标 |
|---|---|---|
| 无监督 | 聚类 | • 惯性值<br>• 调整后的 Rand 系数 |
| | 关联 | • 支持度<br>• 置信度<br>• 提升指数 |
| 监督 | 分类 | • 准确度<br>• 精密度<br>• 召回率/敏感度<br>• 特异性<br>• F1-分数 |
| | 回归 | • 根均方误差 (RMSE)<br>• R 方误差 |

### 3.2.3 惯性和调整后的 Rand 系数

| | | |
|---|---|---|
| LO-3.2.3 | K2 | 比较惯性和调整过的 Rand 系数作为无监督聚类的指标。 |
| HO-3.2.3 | H3 | 应用惯性-簇内平方和（WCSS），并用肘形法确定最佳簇数。 |

对于无监督聚类模型，惯性或者 WCSS（聚类-平方内）是聚类在所有被发现聚类中平均分布。惯性值越小，意味着更好的聚类，因为这意味着群集中的数据点更接近。随着聚类数量的增加，聚类大小（即惯性值）将自然减小。然而，惯性值在聚类超出一定数量时将会停止显著下降。对于无监督聚类的模型，此点显示惯性的最佳值和给定的数据集聚类数-该方法称为肘部法则。

当标签的实际值可用于每个数据点时，调整的 Rand 系数优先于惯性。它是聚类分配（按模型）和实际独立类之间的相似性度量。

### 3.2.4 支持、置信度和提升指标

| | | |
|---|---|---|
| LO-3.2.4 | K3 | 比较无监督关联规则挖掘的支持、置信度和提升指标。 |
| HO-3.2.4 | H3 | 计算无监督关联规则挖掘的支持、置信度和提升指标。 |

**支持**变量集测量它在事务中出现的频次。例如，如果零售店 10 个交易中出现 7 个项目“面包”，它的支持是“70%”。

**置信度**意味着假定 X 已经出现，因变量 Y 出现的可能性。

提升用于消除两个变量集频繁同时发生的情况（因此，置信度指标将会很高）。但是，两个变量可能没有任何相互依赖性。此外，此度量指标可以揭示变量集 X 的多次出现意味着变量 Y 或多或少出现（即 X 和 Y 之间的正向或负向关联）

### 3.2.5 混合矩阵

| | | |
|---|---|---|
| LO-3.2.5 | K2 | 解释混合矩阵 |

监督分类指标使用由真阳性，假阳性，真阴性，假阴性计数组成的混合矩阵计算。

| 混合矩阵 | 正向目标 | 负向目标 |
|---|---|---|
| 正向模型 | 真阳性<br>TP | 假阳性<br>FP |
| 负向模型 | 假阴性<br>FN | 真阴性<br>TN |

### 3.2.6 精密度、召回率、特异性和 F1 得分

| | | |
|---|---|---|
| LO-3.2.6 | K3 | 应用公式计算监督分类的准确度、精密度、召回率、特异性和 F1 得分指标，并进行比较。 |

模型的**准确度**显示在测试数据集中经过训练的模型准确预测占完成总数的百分比或比例。如果一类数据和其它数据相比占主导地位，该指标就不是一个好的的指标选择。
准确度 = (TP + TN) / (TP +TN + FP +FN).

**精密度**意味着模型预测真阳性的准确性。
精密度 = TP/ (TP+FP).

**召回率**是测量模型失败或者错过的检测阳性。
**召回率**= TP/ (TP+FN).

为了尽量减少假阳性，就需要高精密度，而为了尽量减少假阴性，召回率就会很高。

与精密度一样，与召回率相反，**特异性**测量模型区分真阳性的准确度。
特异性 = TN / (TN + FP)

**F1-系数** 是计算精密度和召回率的谐波均值。低 F1-系统代表模型在检测阳性的质量较差。F1-系数的值介于 1 和 1 之间。接近 1 表示质量良好，没有错误数据干扰结果。

### 3.2.7 RMSE 和 R 平方

监督回归模型指标代表回归线和实际数据点的拟合程度。

**RMSE** (根均方误差) 是数据点和回归线距离的测量。是测量预测误差的标准偏差.如果以不同单位测量同一数据集，RMSE 的值会发生变化。

R **平方**是将回归线和使用均值作为预测变量进行比较来测量预测的好坏。值得范围是从 0 到 1，是独立于所使用数据点的单位。

## 3.3 模型评估

指标值依赖于所选择用于训练和验证的数据点。因此，以完全不偏不倚的方式用于训练和验证的数据点成为模型评估的关键方面。

### 3.3.1 训练集, 验证集和测试集

| | | |
|---|---|---|
| LO-3.3.1 | K1 | 通过将可用数据集拆分为训练集，验证集和测试集来定义验证模型的需求。 |

训练集包含用于训练模型的数据。机器学习算法使用验证集来评估训练是否有效。在每一次运行机器学习（这是一个多次迭代的过程），训练数据集和验证数据集将会被合并，并使用不同的方法拆分，以便算法使用不同的组合来学习。

测试数据集是被分出来单独的数据集，用于在机器学习完成后用来验证算法是否经过充分训练。测试数据集不应在训练过程中使用。 [Wikil]

在后训练阶段，机器学习模型使用不同于训练数据集的数据集来评估和测试。然而，为了保证度量指标的右相刑，用于训练，验证和测试阶段的数据集要来自相同或者相似的数据源。使用来自不用数据源的数据集的训练模型的性能可能会非常差。

### 3.3.2 拟合不足和过度拟合

| | | |
|---|---|---|
| LO-3.3.2 | K2 | 总结拟合不足，过度拟合和偏差方差权衡的概念。 |
| HO-3.3.2 | H0 | 演示拟合不足和过度拟合来显示偏差方差权衡。 |

如果受监督的机器模型太简单而无法满足训练数据点（即不能表示数据趋势），这是拟合不足的例子。相反，过度拟合模型试图过多地拟合训练数据点，这通常会导致后续验证阶段或测试阶段预测的准确性较差。

模型的拟合不足和过度拟合的特点也可以用偏差误差和方差误差来解释。如果模型过于简单，并且无法用所提供的功能表示，它被认为是偏差高和预测准确性较差。

如果模型预测性能随着训练数据集的轻微改变而大服务改变，该模型被认为是高方差模型(过于依赖训练数据集)。好的模型需要达到低偏差和低方差。这称为偏置方差权衡.

### 3.3.3 交叉验证方法

| | | |
|---|---|---|
| LO-3.3.3 | K2 | 应用拆分测试、K-折交叉验证、bootstrap 和留一法交叉验证方法。 |
| HO-3.3.3 | H1 | 在算法中应用 K-折交叉验证方法并比较各种结果。 |

把有效数据集分成训练数据集和验证数据集的方法将会导致高偏差和高方差。为了克服这一点，在得出模型指标之前，必须尝试多个拆分组合。一些有用的方法，比如拆分-测试，引导，K-折交叉验证和留一交叉验证。这些方法中的每一个都会多次重复训练和验证过程，并且模型性能是在所有运行中求平均值。

**拆分-测试** 将数据划分为多个部分，用作训练数据集和测试数据集，但是每次划分的比例不同。这将有助于揭示不同的拆分如何产生不同的结果。

**Bootstrap** 通过在完全数据集中随机挑选数据点用于训练，使用剩余的数据集用于验证。

**K-折交叉验证** 通过把数据集拆分成 K 部分和使用 K-1 子集用作训练，并使用剩余的自己用于验证。

**留一交叉验证**和 K-折交叉验证相似，只不过每部分都有一个数据点，这需要更多的执行时间。

## 3.4 分析

### 3.4.1 分析类型

| | | |
|---|---|---|
| LO-3.4.1 | K1 | 回顾四种类别分析的特点-描述性、探索性、预测性、规范性-以及他们在机器学习中的使用。 |

**分析**是理解可用数据集的主要任务之一。数据分析可在四个级别上完成，后续每个级别都是前一个级别的自然延伸。

**描述性分析**是根据中央趋势（均值，中位数，模式）的度量来推导出数据的

统计概要。这有助于深入了解过去并回答：“发生了什么？”［DA1］

**探索性分析**是在高级别上可视化数据集以查看其模式和变体。

**预测分析**是对输入变量进行建模并预测结果的概率。

**规范性分析**是比较预测分析产生的所有可行性结果，并在其中选择/规定最佳预测。

# 第四章 - AI 系统的在线测试

**关键字:** 语言分析方法,聊天机器人

| | | |
|---|---|---|
| LO-4.1.1 | K2 | 解释智能应用系统人工智能部件和非人工智能部件和他们的功能测试和非功能测试需求。 |
| LO-4.1.2 | K1 | 展示人工智能部件和非人工智能部件的信息导向和行为导向的交互。 |
| LO-4.2.1 | K3 | 使用语言分析测试设计方法生成测试场景。 |
| HO-4.2.1 | H1 | 演示使用语言分析测试设计方法生成测试场景。 |
| LO-4.3.1 | K3 | 使用从给定需求和体系架构的测试案例来测试聊天机器人。 |
| HO-4.3.1 | H3 | 在系统级别测试聊天机器人和报告缺陷。 |

## 4.1 AI 应用的结构

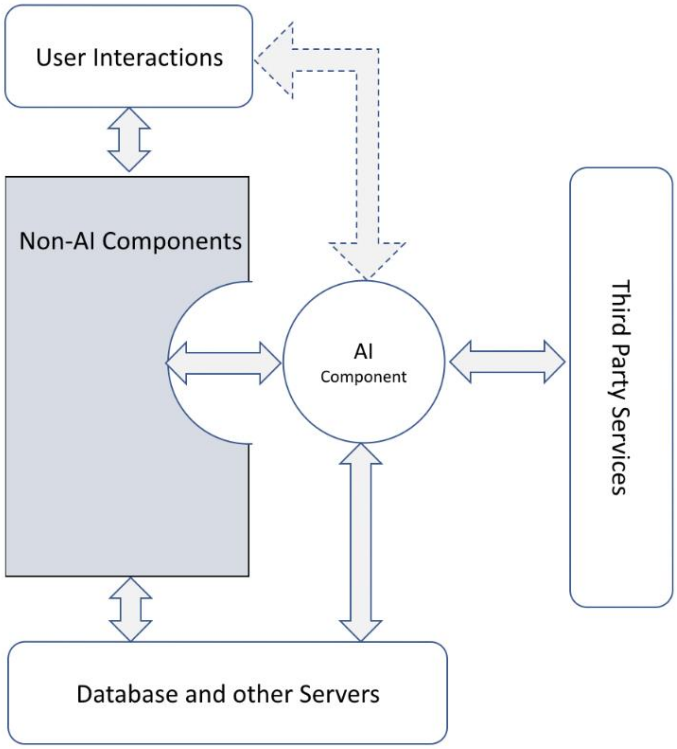
### 4.1.1 解释智能应用系统人工智能部件和非人工智能部件和它们的测试需求

| | | |
|---|---|---|
| LO-4.1.1 | K2 | 解释智能应用系统人工智能部件和非人工智能部件和他们的功能测试和非功能测试需求。 |

典型的 AI 应用需要被检查它是否是一个完整的 AI 应用，或者由较大的

非 AI 应用调用较小人工智能组件集组成的整个系统。此分析需要了解不同的组件（AI 或者非 AI）如何协同工作以提供期望的功能。需要从以下角度来了解不同组件之间的接口：

- 正在传递的输入数据
- 生成输出
- 各组件执行的操作
- 直接用户交互（如有）
- 第三方系统交互（如有）
- 调用频率
- 并行交互的数量（如可能）
- 约束，如
  - 时间
  - 持续时间
  - 输入和输出的范围 Ranges of inputs and outputs
- 必要的先决条件
- 假设/通用设置
- 使用相关转换的可视化输入输出链
- 错误和异常及其处理方式
  - 组件
  - 组件链和最终处理
- 日志记录




此外，系统需要解释系统得出给定解决方案的原因。此外需要使用技术如语言分析和探索性测试来识别给定 AI 系统的测试场景。

需要确保的另一个复杂问题是是否存在 AI 系统的组合。单个独立的 AI 系统可能不能独立的解决真实世界的问题。为了处理那些场景，组合的 AI 系统可以用来对问题进行建模。为此，需要测试策略来处理 AI 系统的组合。

在 AI 和非 AI 交互中，需要确保人工智能组件，非人工智能组件的测试覆盖度，以及涉及两部分之间的交互。

此处提供了测试此类系统的示例：

在邮件应用中，句子完整性由人工智能组件提供。电子邮件系统的用户界面表示非人工智能部件，人工智能组件和其交互以提供建议的文本。非人工智能部件显示建议和根据用户输入进行操作。如果建议被接受，这可能导致存储一些数据以用于将来的学习。

同一个邮件系统也提供在线拼写检查，可能通过人工智能组件提供，也可能通过非人工智能组件提供。

如果我们分析家口，我们发现 UI（非人工智能组件）正在将部分写入的文本传递给人工智能组件。如果有来自人工智能组件的建议，UI 将会通过文

本来显示。与时间和持续时间相关的约束是，建议必须在输入文本后几百毫秒内给出，而且必须是一个连续的过程。测试人员需要确定系统响应速度是否足够快。如果存在时间或者持续时间问题（比如模型预测速度慢），当系统显示建议句子时，用户已经编写其他的文本。这将会使建议句子语法不正确或者建议错误。

输出的 GUI 部分和非 GUI（AI）部分的交互问题的示例可能与建议文本的显示相关。建议文本可能显示在 GUI 的错误位置。建议的句子需要在视觉上与实际句子进行区分，并需要根据用户的操作进行更改。这些行为的任何错误数据 GUI 和非 GUI 交互。

错误/异常处理部分的一个示例是语句完成组件失败（由于某种原因）并且 GUI 能够处理它的情况。

邮件系统的设置包含语言设置和词典设置。例如，英语作为语言设置，美式英语作为词典设置。语句完成系统应该提供美语建议。在这种情况下，拼写和语法检查系统应该使用美式英语。这意味着由于两个交互 AI 或者 AI 和非人工智能组件的假设/设置不匹配，人工智能组件生成的语句不应该被拼写检查组件标记为不正确。

下图显示此类系统被发现的一些问题的示例，A 和 B 显示将一个名字标识为拼写错误而不是另外一个，然而两个名字都在地址簿中。

另一方面，B 和 C 显示不同的速度和缓存。当第一次输入时，名字完成发生，但是名字在几秒内没有被标记为错误，后面被标记为错误。然而，一旦被标记为错误，则删除名称紧接着自动完成组件立即把它标记成错误。

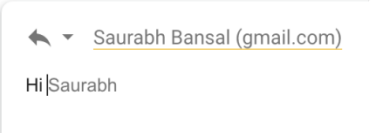
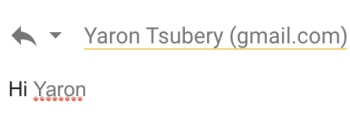
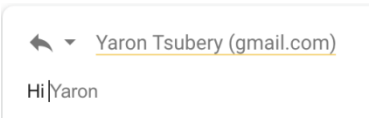



Image A




ImageB




ImageC

## 4.1.2 人工智能与非人工智能的交互

| | | |
|---|---|---|
| LO-4.1.2 | K1 | 展示人工智能部件和非人工智能部件的信息导向和行为导向的交互。 |

在面向信息的调用中，API 或者 API 应用的服务可以从非 AI 终端应用程序中调用。通常，为响应调用 AI 组件。例如，欺诈检查算法的工作原理是调用预定义的训练模型来返回输入事务是否存在欺诈。

一些测试交互的重要测试如下：

- 基于边界值的测试 –包含输入和输出
- 异常测试用例(a.k.a. 边界用例)
- 与要传递数据的大小和类型相关的测试
- 异常处理测试
  - 未收到响应
  - 操作的请求-响应反馈循环断开
- 错误的输入数据测试
- 错误的输出数据测试
- 请求没有被完成
- 反应没有收到
- 性能相关的测试
- 安全相关的测试
- 稳健性相关的测试

人工智能组件和非人工智能组件的交互可能会影响用户体验。交互可能是下列类型：

- 仅标记 (仅仅向系统馈送信息)
- 面向操作，例如，为用户生成响应，或者对问题进行分类。
- API 无法返回结果的交互 API 失败场景
- 从人工智能切换到非人工智能组件，反之亦然

当为 AI 系统设计测试用例时，有以下测试级别：

- 仅测试人工智能部件的测试
- 仅测试非人工智能部件的测试

- 测试两种部件交互的测试
- 缓存响应 VS 学习响应

在较大的非独立系统，需要查看已部署人工智能模型的覆盖范围以及已部署人工智能模型的性能管理。人工智能组件部署以后，系统需要在已部署环境中进行测试。

## 4.2 语言分析测试设计方法

### 4.2.1 基于语言分析的测试设计

| | | |
|---|---|---|
| LO-4.2.1 | K3 | 使用语言分析测试设计方法来生成测试场景 |

在需求几乎没有文档化的情况下， 语言分析可以用来设计大量场景【LA1】。需求的语言分析可以识别测试对象并对他们进行操作。这种方法帮助发现角（非常用）测试用例一个测试人工智能系统的关键需求。
该方法工作如下:

1. 标识表示测试对象的名词和表示操作的动词。
2. 识别名词的属性。
3. 识别属性的属性，不停的重复直到没有更多的属性被识别，或者你认为已达到足够的深度。
4. 识别适用于名词和动词的形容词和副词，以识别更多属性。
5. 在动词上使用 5W1H（什么、为什么、哪里、何时、谁、怎样）来识别提供功能测试和非功能测试的动名词组合。
6. 每个场景询问下列问题？
    a. 代理人是谁？
    b. 完成场景的工具/方法是什么？
    c. 行动的目的是什么?
    d. 行动的方向是什么？

e. 行动的源头是什么?
f. 行动的动机是什么?
g. 谁拥有行动的手段和结果?
h. 行动在哪里进行?
i. 行动的接收者是谁?

这些问题遵循梵文 [LA2] 的语法规则。

在测试 AI 系统的情况下，数据是测试用例。给定的方法允许识别两者，数据和操作（场景）的组合。

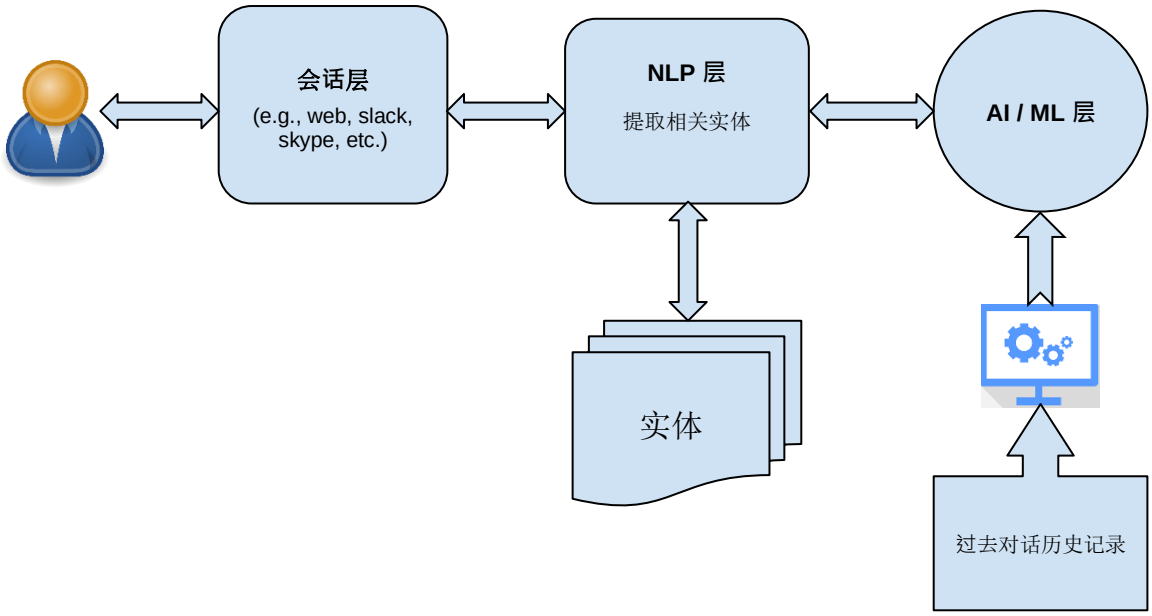
## 4.3 测试 AI 系统

### 4.3.1 测试聊天机器人

| | | |
|---|---|---|
| LO-4.3.1 | K3 | 使用从给定需求和体系架构的测试案例来测试聊天机器人。 |
| HO-4.3.1 | H3 | 在系统级别测试聊天机器人和报告缺陷。 |

聊天机器人目前在很多领域使用，从客户服务机器人，信息机器人到支持机器人。

逻辑架构显示聊天机器人的关键层，包括会话层、NLP 层、AI/ML 层和连接层。会话层管理和终端用户的前端交互，例如，Skype，Slack，Facebook 信使或者 web 前端。NLP 层从最终用户话语中提取相关实体和信息。AI/ML 层解码话语的实际意图，它已经在过去的历史对话中进行了训练，并作为数据输入到系统中。




测试用例应该覆盖每一层的输入数据变量。

会话层需要测试从前端到后端的正确数据传输，反之亦然。在 NLP 层，应该测试原始输入文本语句的数据清洗和预处理是否正确，抽取的实体是否足以在正在进行的会话中传递上下文。应测试 AI/ML 层或者模型（根据历史会话进行训练的）根据抽取的实体（通过 NLP 层）来预测真实意图的准确性，例如，输入文本‘Hi’/‘Hello’/‘Greeting’等，应该预测出‘欢迎’的意图。

此外，如果对话序列发生突然改变，聊天机器人应该能够在各种不同的意图中进行无缝切换。因此，应该识别涉及动态意图切换的测试用例。

非功能需求主要是测试前端的性能，能否承受众多并发客户的调用。

## 第五章 - 可解释 AI

**关键词:** 解读/解释 ML 模型，可解释，LIME，CAM

| | | |
|---|---|---|
| LO-5.1.1 | K2 | 解释可解释 AI（XAI）的需求。 |
| LO-5.1.2 | K3 | 使用 LIME 来解释 AI 模型。 |
| HO-5.1.2 | H1 | 在图像分类器和文本分类器上使用 LIME |
| LO-5.1.3 | K3 | 使用 Grad-CAM 来解释 CNN 模型 |
| HO-5.1.3 | H1 | 使用 Grad-CAM 来确保 CNN 更加透明. |

## 5.1 可解释 AI (XAI)

### 5.1.1 可解释 AI 和它的需求

| | | |
|---|---|---|
| LO-5.1.1 | K2 | 解释可解释 AI（XAI）的需求。 |

一旦 ML 经过训练，它应具有已定义的精度级别，并适用于所有已定义场景的变量。如果模型的预测质量对于某些场景是不够的，对于其它场景是出色的，则它可能是一个有偏差的模型。模型质量低于已定义质量级别的数据集表示一个缺陷。

作为测试人员，如果不使用系统方法，发现全部缺陷是非常困难的。需要通过输入因子的变化来检验模型的行为及其变化，以推断出输入与输出关系的近似值。这种推断称为解读或解释 ML 模型。这种近似关系可能不是实际模型的替换，但它可能足以揭示可能存在的偏见。深入了解模型行为有助于评估其整体质量和模型的部署可行性。要求解读或者解释模型的其它原因可以是安全措施、社会接受度、检测偏差或者人类的好奇心及学习模型 [EA1].

并非所有的模型都能被解释。模型越复杂，解读或者解释它的可能性就越小。非 DL 模型的输出，如随机树林[Wiki2]，决策树[Wiki3],线性回顾[Wiki4]等.都可以从输入变量的角度很好的解释。

DL 模型在实现中本质是复杂的，因此最好将模型作为黑盒来检测，例如通过输入变量的小变动来观察结果变量，并通过简单，可解释的模型近似变量基础模型来观察结果的变化模型。

一些流行和易于使用的模型解释算法、工具和方法包括：本地可解释的模型-不可知性解释 （LIME） [EA2] 和类激活映射 （CAM） [EA3]。

### 5.1.2 LIME

| | | |
|---|---|---|
| LO-5.1.2 | K3 | 使用 LIME 来解释 AI 模型 |
| HO-5.1.2 | H1 | 在图像分类器和文本分类器上使用 LIME |

向 LIME 提供一个样本来研究样本的模型预测及其更密切的变化，并揭示负责模型预测输出的输入要素。LIME 为输入样本生成足够多的接近变量，并为每个变量获取结果。因此，它试图估算出输入变量的微小变化如何改变输出变量的。 既然所有通过 LIME 生成和学习的变量都非常接近给定的样例，LIME 解释被称为“本地”（和输入样例）。因此，它是一种与模型无关方法，因为 LIME 不需要查看模型或者算法。

LIME 可用于图像分类器生成说明。它推断出在决定输出结果方面起重要作用的图像部分。它允许测试人员关联并排除基于不相关功能推导出的模型。

同样对于文本分类器，LIME 可以指出将示例文本分类为预定义分类的单词（文本部分）. 这有助于测试人员评估模型是否是基于不相干单词推导出来的。

### 5.1.3 神经网络 CAM

| | | |
|---|---|---|
| LO-5.1.3 | K3 | 使用 Grad-CAM 来解释 CNN 模型 |
| HO-5.1.3 | H1 | 使用 Grad-CAM 来确保 CNN 更加透明. |

CNN 非常有用，但是他们对于如何得出特定结论不透明。为了给此类模型带来透明度，基于梯度的类激活映射可视化了对这些模型预测非常重要的输入区域。它使用流入 CNN 最终卷积层的特定类的渐变信息，并生成图像中重要区域的粗略本地化地图。

## 第六章 - AI 系统的风险策略和测试策略

**关键字: 预训练模型，概念漂移**

| | | |
|---|---|---|
| LO-6.1.1 | K1 | 回顾测试 AI 系统的风险。 |
| LO-6.1.2 | K2 | 解释在生产系统中使用第三方预训练模型相关的问题。 |
| LO-6.1.3 | K2 | 解释由于上下文变化而再次测试训练模型必要性。 |
| LO-6.1.4 | K1 | 定义 AI 系统的测试环境和回顾它们的挑战。 |
| LO-6.2.1 | K1 | 回顾 AI 应用的测试策略，尤其要考虑测试类型，测试阶段和 |

| | | |
|---|---|---|
| | | 测试级别 |

# 6.1 测试 AI 的风险

## 6.1.1 测试 AI 系统的风险

| | | |
|---|---|---|
| LO-6.1.1 | K1 | 回顾测试 AI 系统的风险 |

测试 AI 系统除了与非 AI 系统关联的风险，还会带来一些额外的风险。其中一些风险是

- 测试条件相关的挑战
  - 由于测试数量众多（每个数据集都是测试），如何验证
    - 测试正确性
    - 测试完整性
  - 缺少可靠的测试语言来指示正确的输出应该是什么和任意输入
  - 识别假阳性和假阴性。因为需要调查故障，所以假阳性容易被识别。假阴性很难被识别，由于这些测试显示为通过。
- 测试数据相关的挑战
  - 在 AI 测试中，测试用例等于测试数据，尤其是对于脱机测试。测试需要大量的测试数据。
  - 数据的可用性是个问题
  - 数据质量差，需要数据清洗
  - 培训和测试需要标记/标识数据。获取此类数据非常昂贵。
  - 生成 ML 系统的角落用例非常困难和成本高
- 调整相关的成本
  - 对于复杂的应用程序，训练和测试脱机模型所需的硬件类型可能非常昂贵。
  - 训练 DNN 的能源成本通常非常高。
- 技能和工具相关的调整
  - 测试 AI 系统的技术要求非常高。测试人员需要理解如何构建 AI 系统以及需要如何测试。
  - 缺少用于测试 AI 系统结构化信息，工具和框架。
- 领域理解和偏差相关的挑战

- 算法最佳质量的正式证明不保证应用程序正确实现或用户正确使用算法
- 需要基于领域完全覆盖输入案例，以避免偏差，覆盖不完整和发生事故的可能性。

### 6.1.2 使用预训练模型的风险

| | | |
|---|---|---|
| LO-6.1.2 | K2 | 解释在生产系统中使用第三方训练模型相关的问题 |

一些组织已提供他们预训练的模型，许多 AI 开发者在使用它们。例如，ImageNet 模型，Inception，VGG，AlexNet 等。这些模型有它们自己需要通过测试发现的偏差和缺点。既然这些系统要么是未知的要么是没记录，使用预训练模型的系统可能导致意外的失败，或者在某些情况下产生不够理想的结果。

### 6.1.3 概念漂移的风险 (CD)

| | | |
|---|---|---|
| LO-6.1.3 | K2 | 解释由于概念漂移而再次测试经过训练的模型的必要性。 |

由于输入要素和输出要素之间的关系发生变化，工作模型可能会随着时间的推移而降低。例如，广告的影响和各种其他类型的营销活动的影响导致潜在客户的行为每隔几个月发生一次变化。这是已知的概念漂移（CD）。[JB2]为了找到发生的概念漂移，需要定期使用最近的数据样本测试工作模型和集成模型。为此，在脱机和集成阶段对模型进行重新训练并重复各种测试和验证。[JB2]

有以下几种方法处理概念漂移，比如：

- 什么都不做
- 使用最近的数据重新训练模型
- 定期使用当前模型上使用最新的数据来更新模型
- 学期变化-当前模型保持不变的合一方法。 新模型采用当前模型的输出，并学习更正预测

- 检测和选择模型–尽可能的检测概念漂移，并选择合适的模型

### 6.1.4 AI 测试环境的挑战

| | | |
|---|---|---|
| LO-6.1.4 | K1 | 定义 AI 系统测试环境和回顾它的挑战. |

AI 系统测试环境能够非常复杂，可能由于不同的用例、上下文以及数据预处理的各种方式和步骤。从离线测试的角度来看，环境需求比从在线测试的角度来看更苛刻。需要大量数据存储、高网络带宽要求以及更大的计算能力来训练/运行模型。

## 6.2 测试策略

### 6.2.1 测试 AI 应用的测试策略

| | | |
|---|---|---|
| LO-6.2.1 | K1 | 回顾 AI 应用的测试策略，尤其要考虑测试类型，测试阶段和测试级别 |

基于实际 AI 的应用程序可能使用一个或者多个 AI 或者非 AI 组件。此类系统的测试策略将包括传统的测试维度以及特定于 AI 组件的新因素及其与其它系统组件的集成。
一些传统的测试策略注意事项是：

- **系统需要的测试级别：** - 单元测试,集成测试,系统测试, 验收测试和系统集成测试。如同 ISTQB® FL 大纲. [ISTQB-FL2018]
- **测试技术** - 白盒测试、黑盒测试和灰盒测试
- **功能测试和非功能测试**- 尤其安全测试，性能测试，稳健性测试和可扩展性测试
- **使用自动化测试**

对于基于 AI 的应用程序测试，我们需要额外考虑以下的方面:

- **离线测试（功能性）** - 测试经过训练的 AI 模型是需要作为 SDLC 的一部分执行的另外一个步骤，在该阶段的技能需求也是需要被考虑的。

- **离线测试 (非功能)** -测试经过训练的 AI 模型的各种非功能方面。在部署之前，需要测试模型的速度、资源利用率、并发性和负载、可扩展性。
- **黑盒测试** - 大多数 AI 模型作为 API 暴露，通常作为黑盒被调用。AI 系统中缺少的一个关键组件是缺少测试 oracle，使得黑盒测试变得困难。蜕变测试等技术在 ML 中越来越受欢迎，因为它不需要 Oracle。如果测试用例成功运行，将执行测试用例中的变量并检查输出。输出仅满足关系的需求不需要任何 Oracle. [Wiki5]
- **白盒测试** - 总之，机器学习系统的白盒测试非常困难，除了一些简单模型，模型内部工作既不清楚也不可访问。然而最近，机器学习系统的一些白盒技术出现了。DeepXplore 就是一个例子，它是一种自动白盒测试产品。使用神经元覆盖，研究人员能够揭示数千种独特错误边界案例行为。 [DX1]
- **数据采集和预处理**-可用数据可能并不总是按照相同格式去训练或者测试其它模型。该部分在章节 3.1 数据准备和处理中详细讨论。
- **将开发环境实现转换到生产环境**- 例如, 将 Jupyter 笔记本转换为在云实例上的 Docker 容器内的服务器上运行的 Python 代码。完成后，生成的模型可以保存为文件并在应用程序中使用。

## 第七章 - 测试中的 AI

**关键字：**测试自动化

| | | |
|---|---|---|
| LO-7.1.1 | K2 | 解释 AI 如何协助各种测试活动-测试计划与估算，风险分析，测试用例设计和测试数据生成，缺陷分配，影响分析，覆盖范围分析. |
| LO-7.1.2 | K2 | 解释 AI 如何支持报告和智能仪表盘. |

| | | |
|---|---|---|
| LO-7.2.1 | K2 | 举例说明各种商用基于 AI 的测试执行自动化工具的的使用。 |

AI 能用于改进 SDLC 中现有的测试流程。其理念是利用测试过程中生成的数据，提供详细的分析，更多的自动化，更深入的见解和模式以及预测和采取纠正措施的能力。

## 7.1 人工智能软件测试生命周期(STLC)

### 7.1.1 AI 支持 STLC 方法

| | | |
|---|---|---|
| LO-7.1.1 | K2 | 解释 AI 如何协助各种测试活动-测试计划与估算、风险分析、测试用例设计和测试数据生成、缺陷分配、影响分析、覆盖范围分析. |

在测试计划和估算方面，软件项目总成本是可以估算的，考虑到与 AI 项目相关的不同输入，包括数据大小、所涉及的工作量、平台选择、应用程序类型、数据准备时间、培训时间和测试时间。鉴于这些输入的历史数据，ML 模型提供更准确的估计。

AI 可帮助确定风险的优先级，获取与计划遵守相关的更准确的指标，并帮助更准确地识别应用程序的性能指标。

根据测试设计，使用 AI 技术，如自然语言出来（NLP）和文本挖掘，可以从文本需求文档自动生成测试用例。此外，AI 应用代码分析（静态和动态），除了分析从测试收集的数据外，还可以标记潜在的性能问题和其它非功能需求潜在的风险。

此外，在历史数据上运行 ML 可以帮助识别测试数据格式和为组件测试和系统测试产生自动化测试数据.

特别是，图像数据和 GUI 元素，AI 可以帮助自动识别渲染不正确的元素。此外，通过基于数据为中心分析不同的可能流的 ML，正确流可以被自动化。

关于缺陷自动预测，基于代码质量度量，模型使用 ML 预训缺陷。对于自动缺陷预测，使用 ML 的模型可以根据代码质量指标预测缺陷。

使用 ML 在代码上进行的影响分析可以根据更改自动识别受影响的模块和文件。

在使用 AI 的覆盖范围分析方面，通过分析捕获的数据流，可以帮助实现全面的测试和代码覆盖率。

### 7.1.2 AI 支持报告和智能仪表盘

| | | |
|---|---|---|
| LO-7.1.2 | K2 | 解释 AI 如何支持报告和智能仪表盘 |

在报告和仪表板的上下文中，使用 ML 有助于生成重点洞察和汇总数据，以便在智能仪表板中演示，而不是纯粹的分析。

## 7.2 基于 AI 的自动化工具

### 7.2.1 工具

| | | |
|---|---|---|
| LO-7.2.1 | K2 | 举例说明各种商用基于 AI 的测试执行自动化工具的的使用。 |
| HO-7.2.1 | H2 | 使用基于人工智能的测试执行自动化工具的演示/试用版。 |

市场上有大量的自动化工具。有些使用 AI 的工具使得自动化更容易或更容易维护。

AI 工具可以使用 GUI Spiders，它遍历整个 GUI 并记录应用程序。通过迭代，他们能够学习、比较和识别 Bug。

某些工具组合了可视化 GUI 测试的元素，并使用 ML 来找出定位器和元素之间的更改和可能的相关性，以及更改是 Bug 还是预期更改。

有些工具使用 NLP，有些在 DOM 级别工作，有些在 UI，有些用于日

志，有些工具结合了功能和性能测试，有些工具综合了上述方法。

基于图像的工具可以使用基于 AI 的图像分类工具来标记 UI 缺陷。这些可能用作具有记录和播放的 Web 浏览器插件。基于 AI 的图像比较优于简单的图像比较。

## 参考

**[AG1]** Neural Networks and Deep Learning - By Aurélien Géron (O'Reilly)

**[BT1]** A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives - By L. Anderson, P. W. Airasian, and D. R. Krathwohl (Allyn & Bacon 2001)

**[BT2]** https://www.apu.edu/live_data/files/333/blooms_taxonomy_action_verbs.pdf

**[DA1]** 4 types of data analytics to improve decision-making https://www.scnsoft.com/blog/4-types-of-data-analytics

**[DI1]** Introduction to k-Nearest Neighbors https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/

**[DSC1]** https://www.datasciencecentral.com/profiles/blogs/the-data-science-project-lifecycle

**[DX1]** DeepXplore: Automated Whitebox Testing of Deep Learning Systems, SOSP '17 Proceedings of the 26th Symposium on Operating Systems Principles (Pages 1-18) http://www.cs.columbia.edu/~junfeng/papers/deepxplore-sosp17.pdf

**[EA1]** Interpretable Machine Learning: A Guide for Making Black Box Models Explainable (Section 2.1 Importance of Interpretability) - By Christoph Molnar https://christophm.github.io/interpretable-ml-book/agnostic.html

**[EA2]** "Why Should I Trust You?" Explaining the Predictions of Any Classifier https://arxiv.org/pdf/1602.04938v1.pdf

**[EA3]** Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization
https://arxiv.org/pdf/1610.02391.pdf

**[ISTQB-FL 2018]** ISTQB Foundation Level Syllabus version 2018. Available at https://www.istqb.org/downloads/category/51-ctfl2018.html

**[JB1]** Machine Learning: Hands-On for Developers and Technical Professionals - By Jason Bell (WILEY 2014)

**[JB2]** A Gentle Introduction to Concept Drift in Machine Learning
https://machinelearningmastery.com/gentle-introduction-concept-drift-machine-learning

**[KUR]** The Age of Intelligent Machines - By Ray Kurzweil (MIT Press 1990)

**[LA1]** Visual Modeling for Test Idea Generation - By Vipul Kocher
https://www.testnet.org/testnet/download/preview-voorjaar-2015/visual-modeling-for-test-idea-generation-v2.pdf

**[LA2]** Vibhakti - By Ujjwol Lamichhane
https://www.scribd.com/doc/30377592/Vibhakti-%E0%A4%B5%E0%A4%BF%E0%A4%AD%E0%A4%95-%E0%A4%A4%E0%A4%BF

**[SMU]**The CRISP-DM User Guide
https://s2.smu.edu/~mhd/8331f03/crisp.pdf

**[TDA]** Testing in the digital age: AI makes the difference - By Tom van de Ven, Rik Marselis and Humayun Shaukat (Sogetibooks 2018)

**[UDI]** Image Preprocessing
https://towardsdatascience.com/image-pre-processing-c1aec0be3edf

**[UDT]** Text Preprocessing in Python: Steps, Tools, and Examples
https://medium.com/@datamonsters/text-preprocessing-in-python-steps-tools-and-examples-bf025f872908

**[ULA]** Unsupervised Learning: Association Rules - By Krzysztof J. Cios, Roman W. Swiniarski, Witold Pedrycz, Lukasz A. Kurgan

[Wiki1] https://en.wikipedia.org/wiki/Training,_validation,_and_test_sets

**[Wiki2]** Random forest
https://en.wikipedia.org/wiki/Random_forest

**[Wiki3]** Decision tree learning
https://en.wikipedia.org/wiki/Decision_tree_learning

**[Wiki4]** Linear regression
https://en.wikipedia.org/wiki/Linear_regression

**[Wiki5]** Metamorphic testing
https://en.wikipedia.org/wiki/Metamorphic_testing