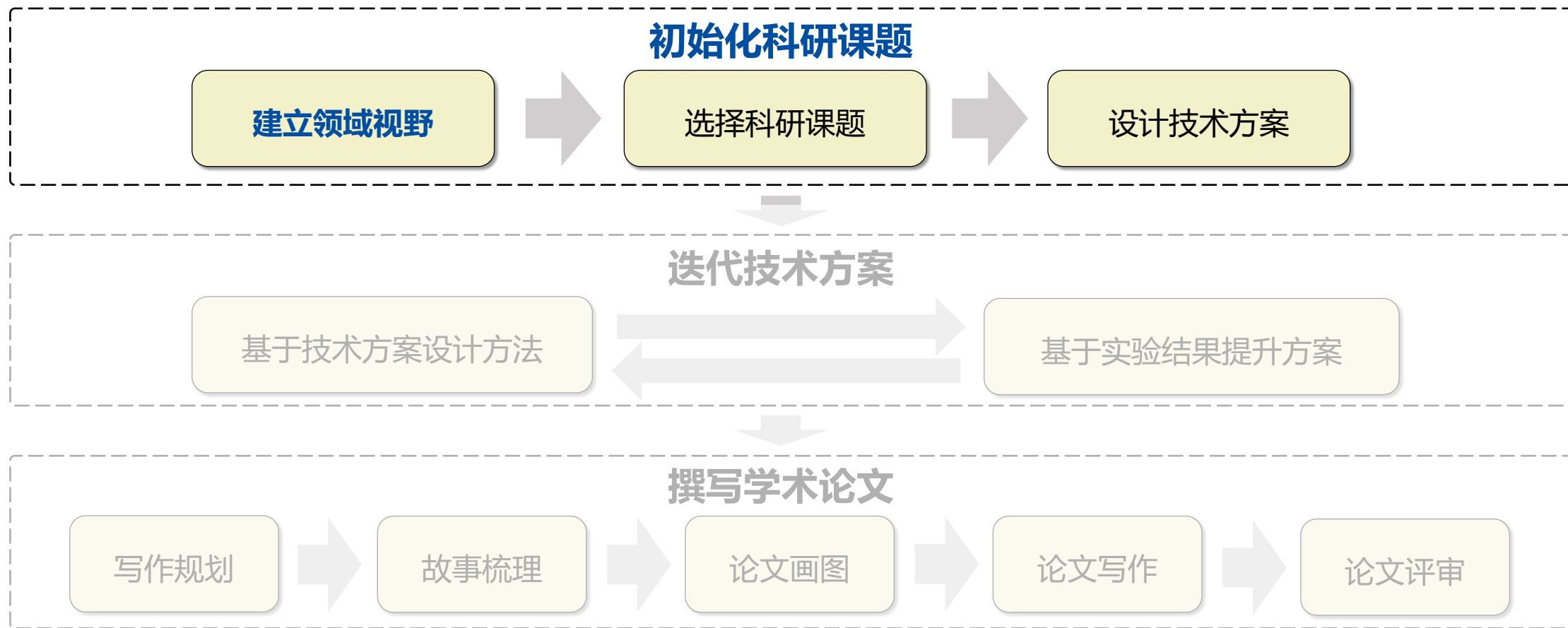


第一部分：科研课题初始化



什么是领域的视野

“领域” 的定义：特定的科研方向，如3D reconstruction、Novel view synthesis。

什么是领域的视野

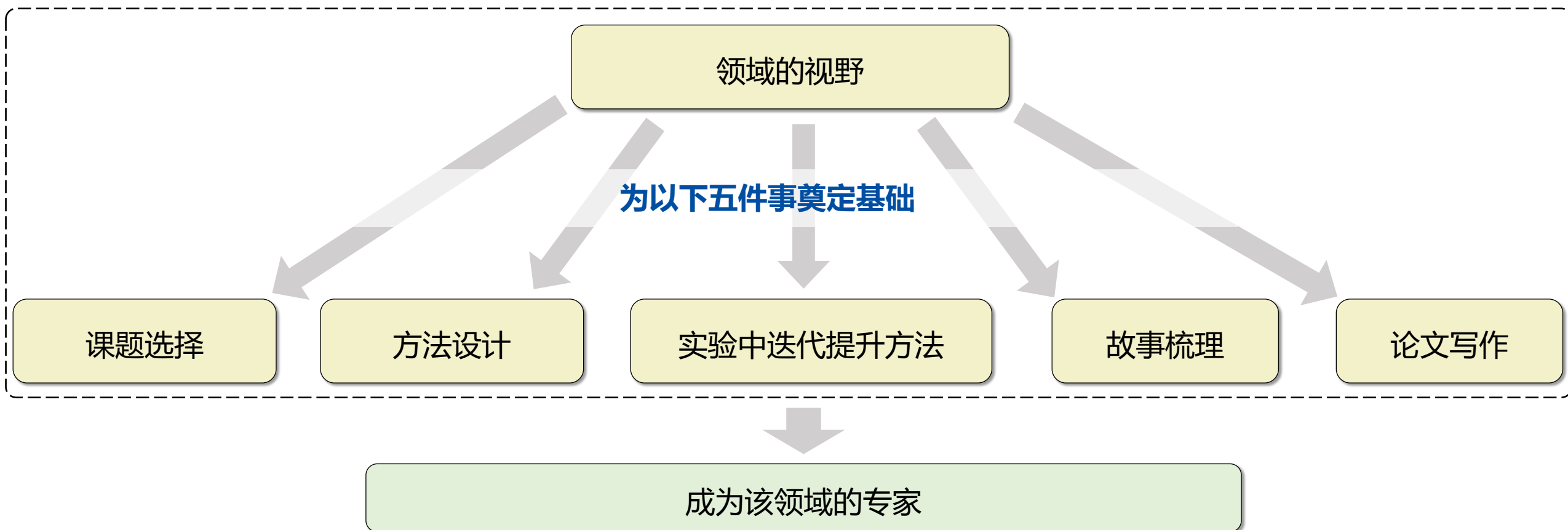
对“技术演变”的视野

- 有哪些milestone papers？
- 领域技术如何随着时间的推移而演变？

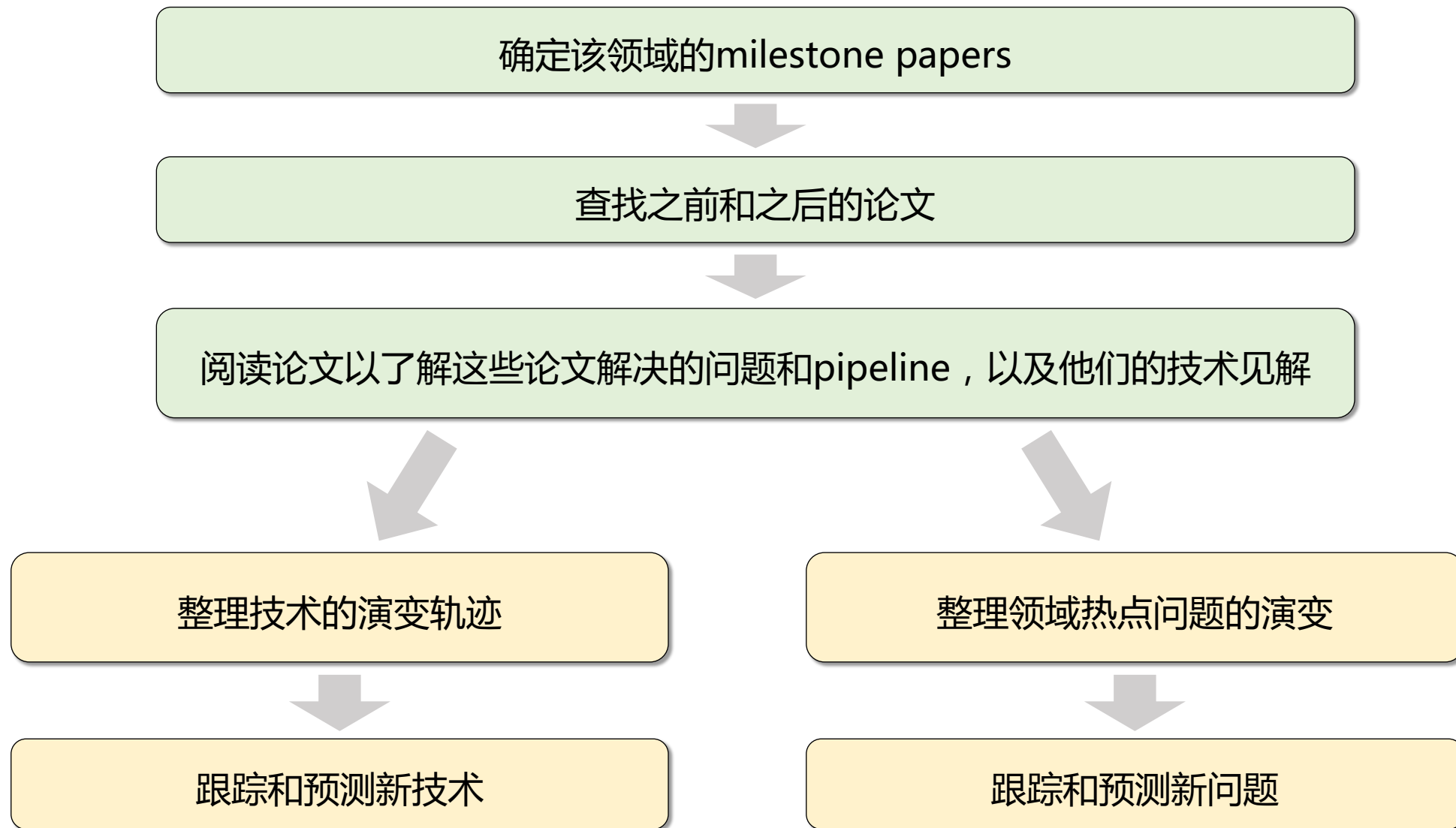
对“重要问题”的视野

- 这个领域的终极目标是什么？
- 该领域已经达到了什么水平？
- 还有哪些重要的问题仍未被解决？
- 现阶段的热点话题是什么？

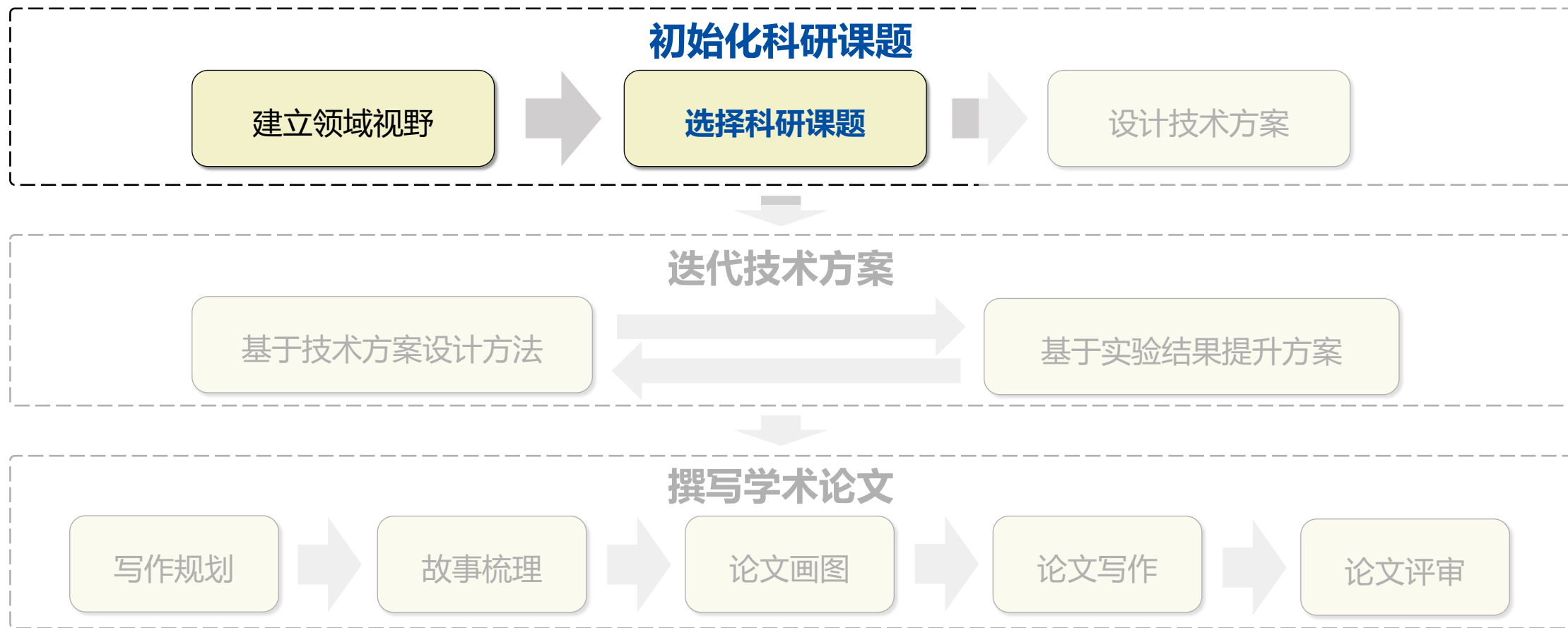
为什么我们需要建立视野



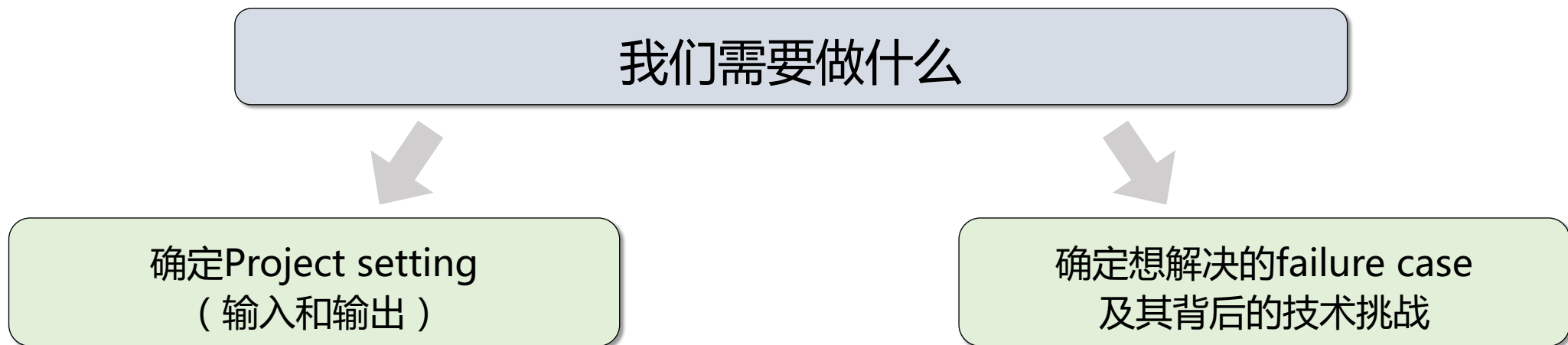
如何建立领域的视野



第一部分：科研课题初始化



课题选择



为什么课题选择非常重要

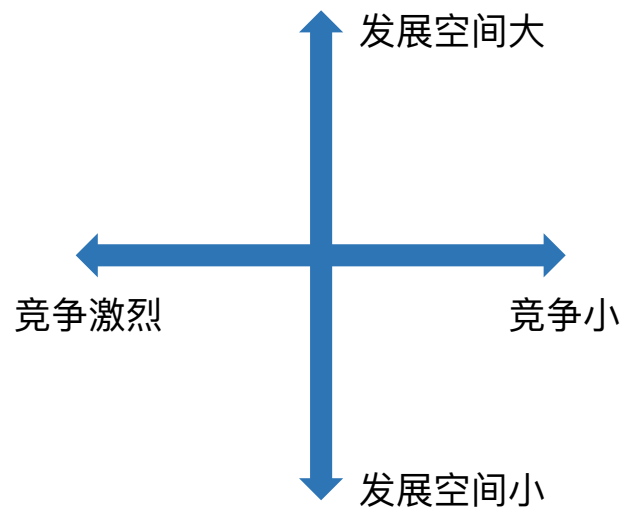
- 一个好的课题将使科研变得容易很多。

什么是好的课题

发展空间大、竞争程度小

任务难度与个人能力相匹配

该课题的影响大



如果任务太难，可能会导致无法完成。
如果任务太简单，可能没法锻炼自己的技术能力。

考虑该项目可以产生的学术影响程度。

如何找到好的课题

寻找好课题的四个步骤

建立领域的视野

- 整理milestone papers/技术范式。
- 了解它们是如何随着时间演变的。

列出领域内的课题

- 列出该领域的重要问题。
- 判断哪个问题是当前时间点应该去解决的。

判断哪些课题是好课题

- 评估课题的竞争程度和发展空间。
- 考虑任务的难度与个人能力相匹配。

在选题上Aim High

- 考虑任务的影响力。
- 技术风险性大的课题往往有更高的收益。

Aim high: 努力完成一个达到你能力极限的科研课题。

不同阶段的学生应该如何选择科研课题

Junior students, Senior students

发展空间小 发展空间大



Junior students, Senior students

竞争小

竞争激烈



Junior students

Senior students

技术风险小

技术风险高



Junior students, Senior students

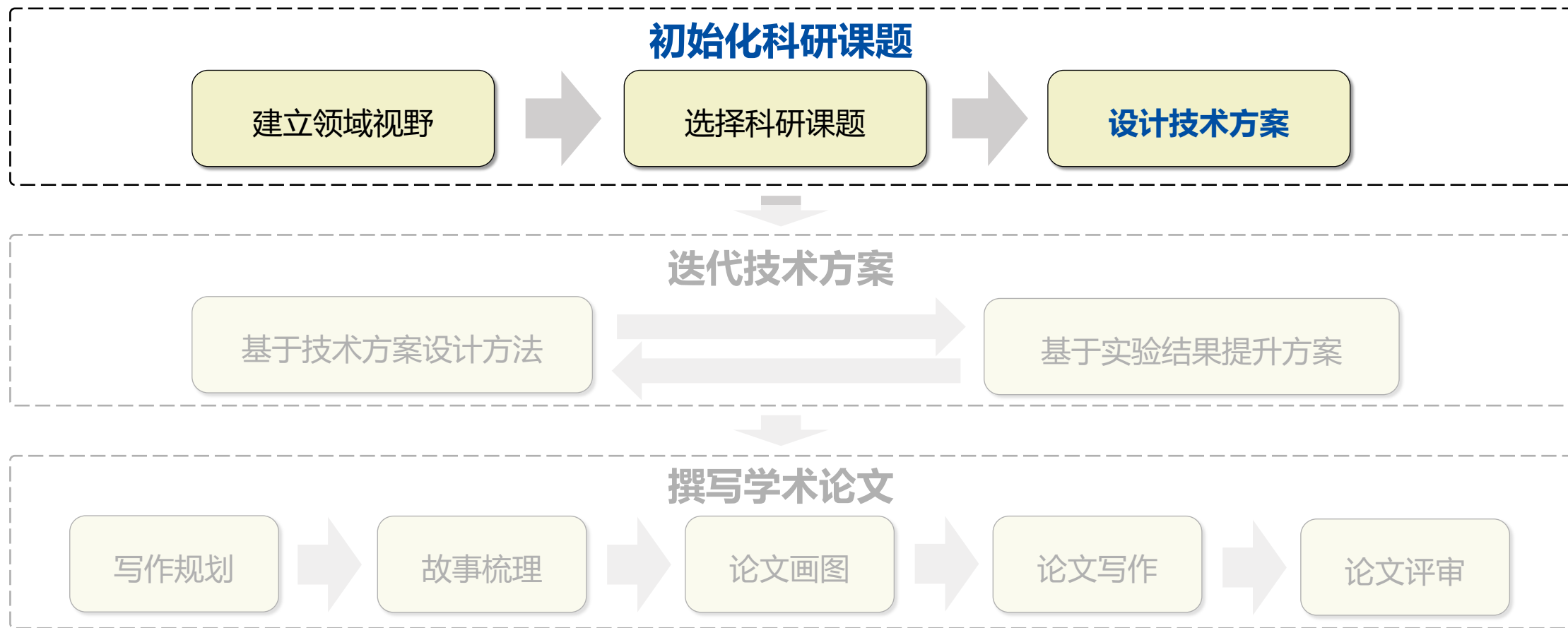
Senior students

well-defined task

ill-defined task



第一部分：科研课题初始化



方法设计

什么是方法设计

- 在选定课题后，我们需要设计一种新颖的pipeline来达到SOTA performance。

为什么要有方法设计的系统性思路

- 一个清楚的方法设计思路能更有效地解决问题并达到SOTA。
- 让我们提出的方法更有动机，同时使得论文的故事更吸引人。
- 保证了方法的技术创新性，以免审稿人说它没有novelty。
- 同时，该系统性思路可以有效提升我们的技术insights。

如何设计方法

分析原因

- 遵从第一性原理，分析为什么当前的SOTA效果不够好。

设计方法

- 根据分析的原因，设计一种能达到SOTA的方法。

判断合理性

- 判断提出的技术方法是否合理

改进方法

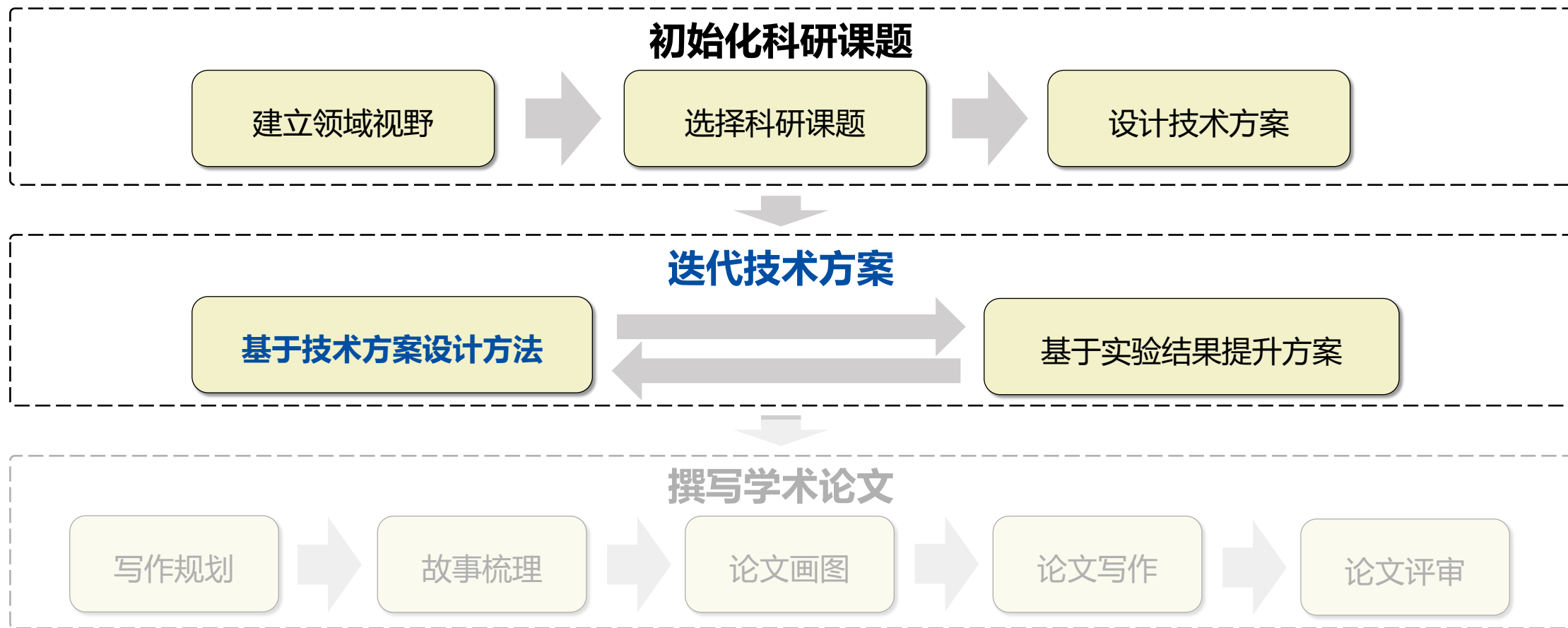
- 通过读论文、讨论和做实验改进技术方法

如何设计方法

三个要点

- 不只要Work，还要保证论文方法的**技术创新性**。
- 争取在Pipeline层面设计解法。
- 要判断论文方法是否有足够技术贡献，是否有新的知识、新的认识、新的观点。

第二部分：实验中迭代提升方法





实验设计

实验设计是什么

- 规划要做哪些实验，实现并提升Technical idea，达到SOTA指标。

实验设计的两个好处

- 让研究变得更简单：想清楚实验的目的，明确科学研究的方法。
- 让实验更简单：降低实验难度，提高实验效率。

如何设计实验（简单版）

- 核心原则：减少实验中包含的探索点的数量。

如何减少exploration points的数量

分解pipeline

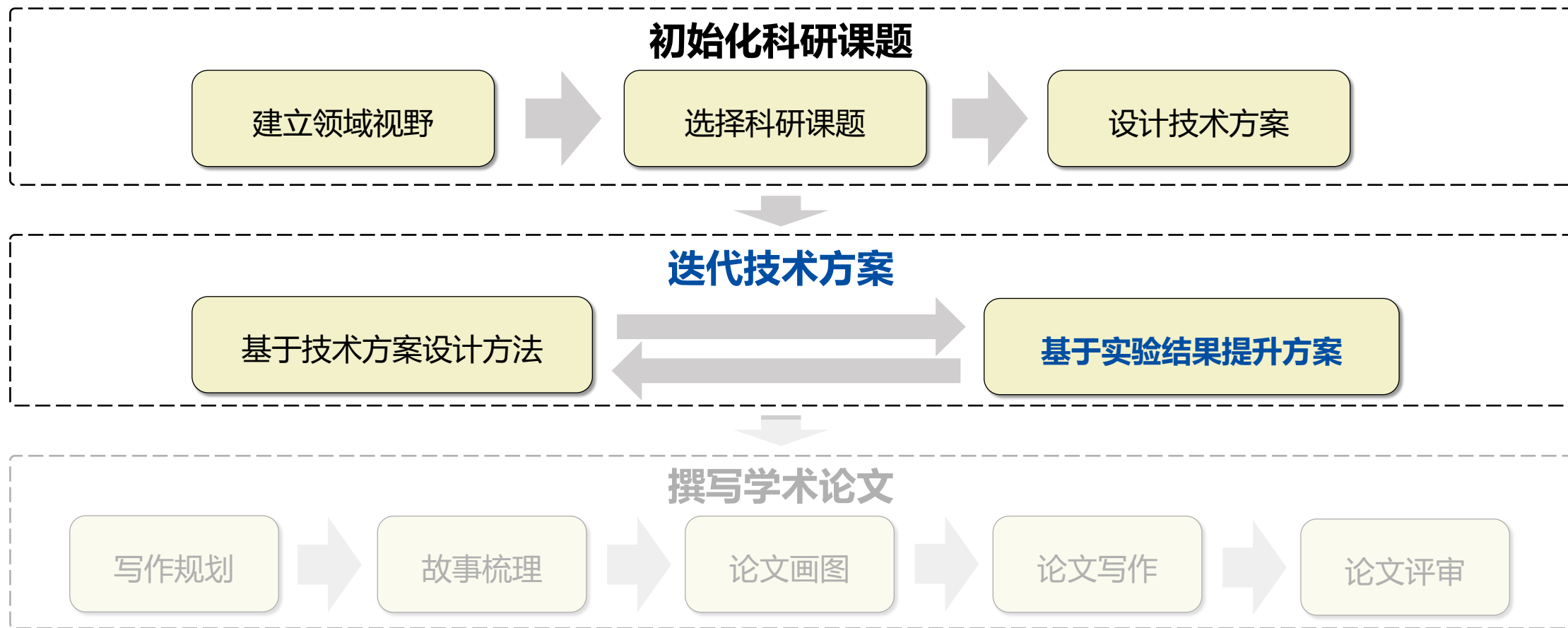
将idea分解为不同的组件，从可控的idea开始，不断添加探索性和创新性的框架/模块。

分解实验设置

从一个简单的setting开始探索，然后逐渐增加难度，之后进入到真正的setting。

我们还应该考虑exploration points的重要性，进行实验优先级排序

第二部分：实验中迭代提升方法



为什么需要改进方法

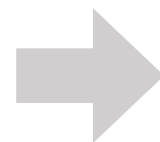
- 因为初始提出的方法通常会不work。

理想情况：

提出一个idea



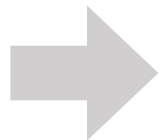
进行实验



Good results

实际情况：

提出一个idea



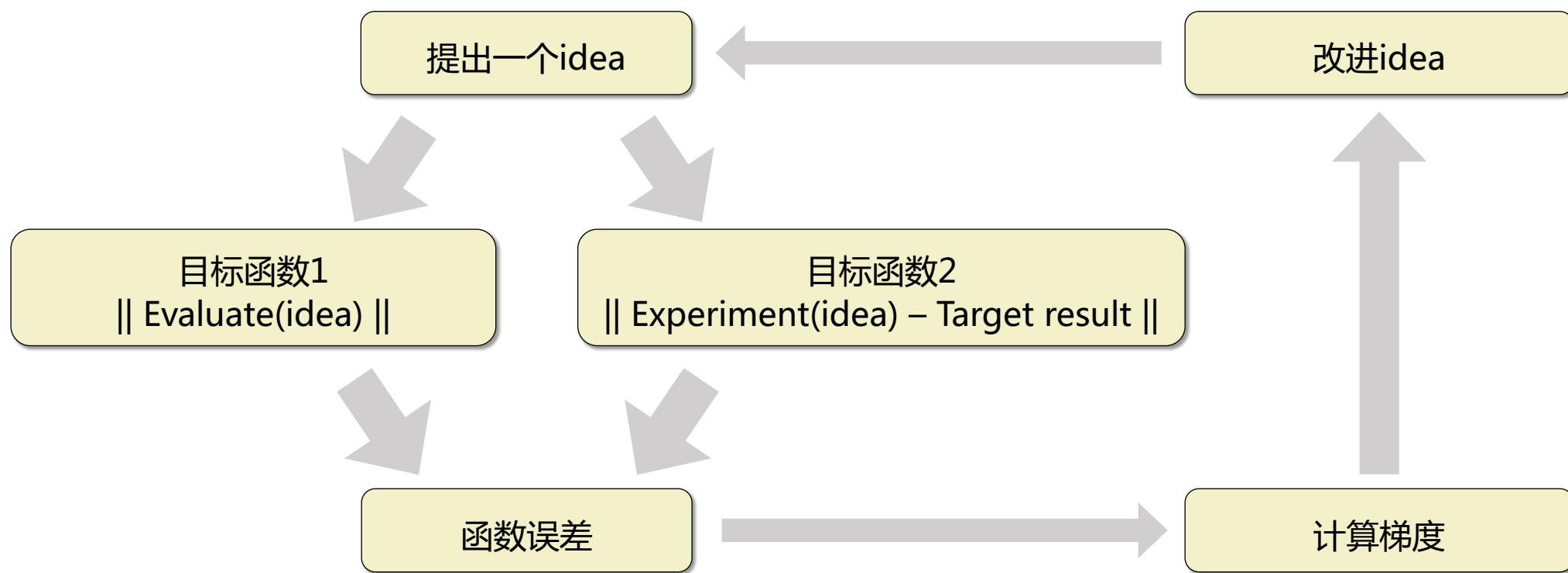
进行实验



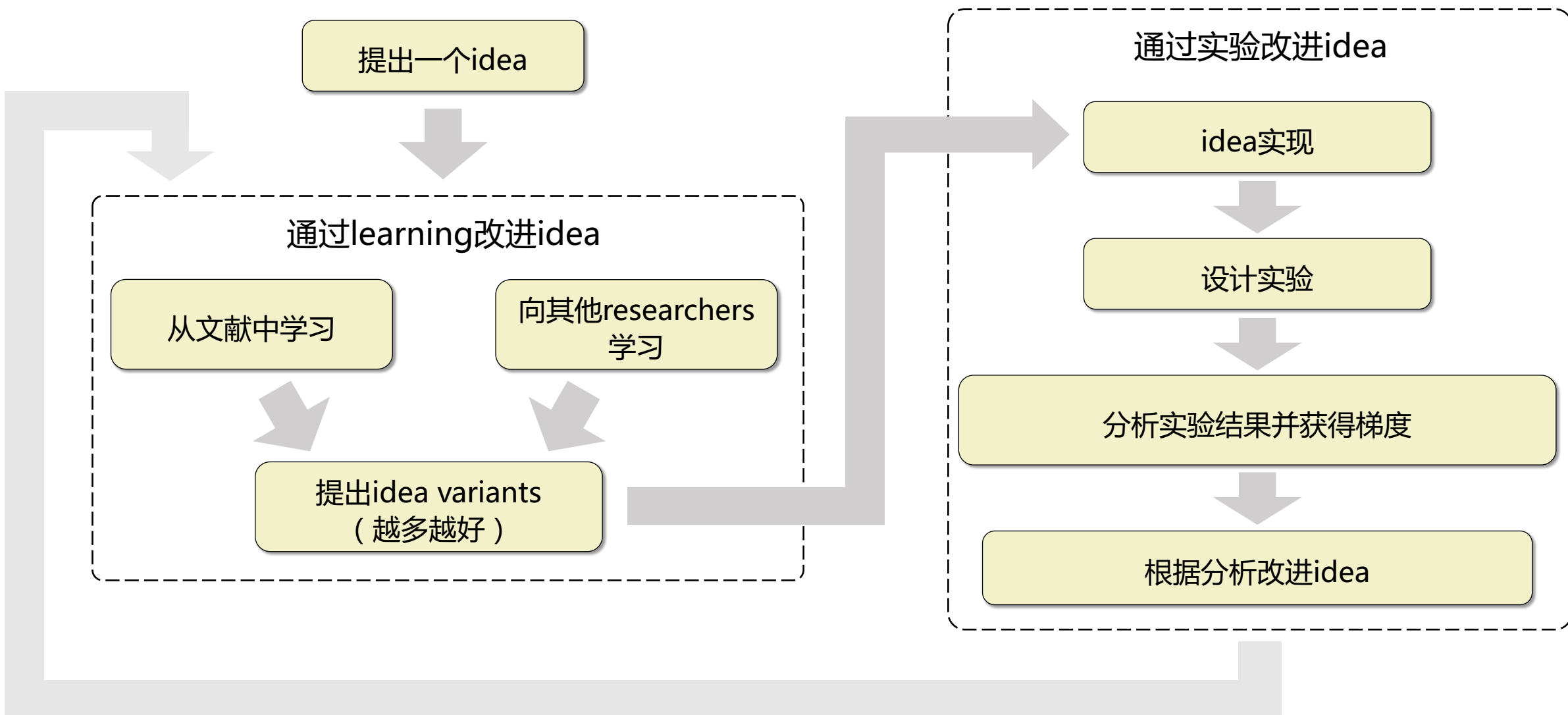
Not working

如何改进方法

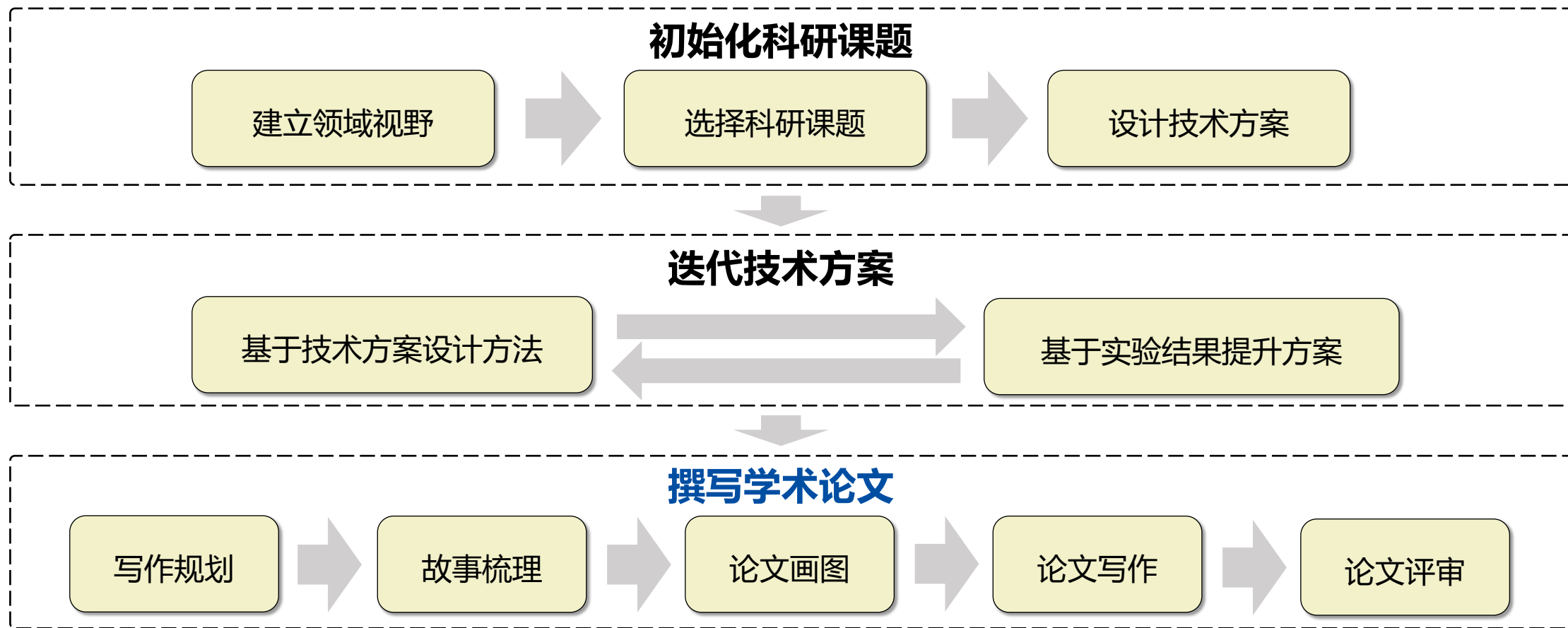
- 可以把改进方法的过程当作SGD优化过程。



如何改进方法



第三部分：论文写作



写作规划

Step 1 画一个清楚的pipeline figure的草图（理清楚方法的流程步骤），梳理论文story，写一版introduction的初稿。

Step 2 列出要做的comparison experiments和ablation studies，并开始做起来。

Step 3 写method，同时做实验。

Step 4 改introduction和method，同时做实验。

Step 5 实验做差不多以后，写experiment。

Step 6 写related work。

Step 7 Review论文。改论文的introduction、method和experiment。

Step 8 写abstract，取论文名字。

Step 9 反复review论文，改论文。



为什么我们需要写作规划

为什么规划很重要

- 大幅**提高投稿率**！！！！
- 让自己做论文更轻松，**避免最后几天熬夜**。

怎么梳理论文的故事

首先，整理论文方法流程

给论文画一个pipeline figure的草图，帮助自己直观地看清论文方法：

- 按照“输入→模块1→中间输出1→模块2→.....→模块N→输出”的风格，用文字描述pipeline。（**我们需要清楚地描述 pipeline**）。
- 选择流程图的布局并绘制简单的流程图。

怎么梳理论文的故事

第二，按序回答以下问题

1. 我们的pipeline有哪些贡献（提出新任务、确定新的技术挑战、提出新的技术贡献）？
2. 我们的贡献有哪些好处？它们解决了哪些技术挑战？
3. 我们的论文带来了哪些新的insights？
4. 我们如何通过介绍以往的方法引入我们关注的技术挑战和新的insights？

第三，概述论文故事

1. 介绍论文的任务。
2. 通过讨论以前的方法来引入我们解决的技术挑战。
3. 为了应对这一技术挑战，我们提出 xx 贡献。
4. 我们的贡献有哪些技术优势，有哪些新的insights？

论文画图



画好看的图的目的: **提高中稿率.**

怎么论文画图

绘制pipeline图的步骤流程

- 按照 “输入→模块1→中间输出1→模块2→.....→模块N→输出” 的风格，用文字描述pipeline。（ **我们需要清楚地描述 pipeline** ）。
- 选择流程图的布局并绘制简单的流程图。
- 可视化 “输入”、“模块”、“中间输出” 和 “输出” 等元素。
- 给流程图涂上颜色。
- 优化流程图布局，保证简洁。避免过多的空白区域和大的空白区域。

怎么写论文的方法

写method的步骤

- 画一个pipeline figure的草图。
- 梳理Method section，确定各个subsection将涵盖哪个方法模块。
- 分别写各个subsection。每个subsection包括三个部分：**模块的具体设计**、**模块的motivation**、**模块的技术优势**。
 - 首先写清楚模块的具体设计：给定xxx输入，第一步做xxx，第二步做xxx，第三步做xxx，最终得到xxx输出。
 - 然后模块的motivation、技术优势。



论文自我评审

评审论文时我们要做什么

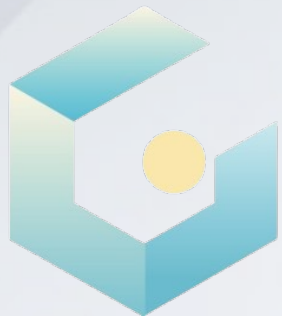
- 确定审稿人可能指出的潜在问题。

为什么要自我评审

- 以提高论文的中稿率。

如何自我评审论文：A checklist

1. 技术贡献是否足够	1.1 想解决的failure cases很常见
	1.2 提出的技术已经被well-explored了，该技术带来的performance improvement是可预见的/well-known的
2. 论文写作是否清楚	2.1 论文Introduction是否清楚描述论文贡献
	2.2 论文pipeline figure是否清楚描述pipeline与技术贡献
	2.3 缺少技术细节，不可复现
	2.4 每个方法模块是否都写了motivation
3. 实验效果是否足够好	3.1 是否比之前方法好很多
	3.2 实验效果是否让人impressive
4. 实验测试是否充分	4.1 缺少ablation studies
	4.2 缺少重要的baselines、缺少重要的evaluation metric
	4.3 数据太简单，无法证明方法是否真的work
5. 方法设计是否合理	5.1 实验的setting不实际
	5.2 方法存在技术缺陷，看起来不合理
	5.3 方法模块的鲁棒性：是否需要在每个场景上调超参
	5.4 新的方法设计在带来benefit的同时，引入了更强的limitation，导致新方法的收益为负



Thank you!



Sida Peng



Jun Gao



Songyou Peng



Qianqian Wang