各位同事，大家好!

今天我非常荣幸地和大家分享一个对我们研发团队而言至关重要的课题——大语言模型（LLM）的应用与实践。本次培训，我们将结合实际场景，尤其是以我们内部部署的 DeepSeek 为例，深入探讨如何借助大语言模型提高生产效率、优化研发流程。

首先，让我们从背景谈起。当下，我们已经全面进入人工智能的智算时代。从过去 70 年缓慢积累，到今天 AI 技术成熟落地，大规模应用时代真正到来。支撑这一巨大发展的三大基石分别是数据、算法与算力。这三者缺一不可：数据的丰富和高质量使模型得以学习更多知识，算法的优化则帮助模型更高效地理解和处理这些数据，而算力的提升，则为模型规模扩张和性能提升提供了强大基础。

纵观过去几年，尤其从 2015 年开始，以 GPT 系列为代表的大语言模型迅速崛起，参数规模和训练数据量剧增，训练算力需求在短短几年内翻了数十万倍。特别是去年 ChatGPT 的问世，标志着大模型进入了全新的时代。预计未来 10 年，算力需求还将持续爆炸式增长 500 倍以上。这种惊人的需求增长也催生了全球范围内芯片产业的快速布局与竞争。

在这样的时代背景下，国产化 AI 芯片与国产开源大模型迅速发展，成为我们国家在全球 AI 竞争中的战略重点。过去几年，国外模型生态较为封闭，中国的研发企业面临芯片供应、算法受限的双重压力。然而，以 DeepSeek 为代表的国产大语言模型迅速崛起，在国际权威模型评测中取得卓越成绩，其快速成长的开源社区也为我们的自主创新和技术突破提供了重要支撑。

那么，到底什么是大语言模型，它是如何工作的呢？简单来说，大语言模型是通过在海量文本数据上进行预训练，让模型掌握语言的基本结构与知识。为了进一步提高模型表现，我们重点关注三个方面：

第一是数据的质量与规模，数据越丰富、越全面，模型的表现就越强大；

第二是模型本身的结构与参数量，以及训练策略。目前主流的强化学习（RLHF）和指令微调（Instruction tuning）策略，都能显著提高模型在实际任务中的表现；

第三是算力，拥有充足的算力才能支持更大规模的训练、更高效的推理。

结合到我们具体的研发院应用场景中，DeepSeek 大模型已经开始展现出巨大潜力，例如：

- 在代码生成与代码审查方面，DeepSeek 模型帮助开发者快速、高质量地完成编程任务，有效减少人工审查的负担；

- 在文档自动撰写、技术文档整理和知识库维护方面，大模型也能自动生成结构清晰、语言准确的技术文档，极大提升了信息流转的效率；

- 在辅助决策方面，DeepSeek 能够迅速分析大量数据，并给出有效建议，辅助研发人员做出更加精准高效的决策。

当然，技术在应用落地中也会遇到不少挑战。例如，模型的输出精准度不够，容易产生错误信息；上下文处理能力有限，无法处理长文档；以及本地算力资源不足等问题。针对这些典型问题，我们提出了以下具体应对策略：

- 首先，为了提升输出精准度，我们应该优化 Prompt 设计，明确需求，避免模糊措辞，同时也要提供必要的背景信息，将问题进行分步细化，减少模型出错的概率；

- 对于上下文窗口限制，我们建议合理分割任务或文档，精炼输入内容，并且灵活运用链式 Prompt 策略，提高模型处理长文本的能力；

- 在算力资源方面，我们需要持续推动与新兴 AI 芯片供应商合作，并且在 IT 部门的支持下启动算力资源的持续优化与升级规划，确保算力不会成为应用落地的瓶颈。

今天的分享只是起点，希望各位同事能够充分了解和掌握 DeepSeek 大模型的使用技巧，并将其应用到日常研发工作中去，共同推动我们研发团队迈向新高度。

谢谢大家的倾听！期待我们后续更多的交流和探讨！