Transformer

??

1. 精细的抽象，记忆空间特别大
2. 大数据，大算力，大模型
3. scaling特性，可以训练很大的模型，用很多数据获得更多智能
4. “硬件彩票”，高强度对着GPU设计，能打满GPU利用率
5. 在nlp问题里面，通过逐个处理新的token，递归 得进行抽象
6. 使用少量的权重对自然语言逐个token进行计算
    1. 先用kq权重映射到当前token对应的空间，再用v权重映射到输出
    2. 通过多层叠加，使得kqv权重能表达整个sequence的范围
    3. 通过少量权重、大量的计算扩展了表达空间
    4. weight的存储效率非常高，充分的训练，提高了对样本抽象的质量
7. 语义压缩
    1. LLM难以处理细粒度的语义差异。它们的内部概念结构与人类对类别归属的直觉不相符
    2. LLM侧重于统计压缩，力求最大程度地减少冗余信息；而人类则更注重适应性和丰富性，强调保持灵活性和上下文的完整性
8. hidden size 7168的向量维度是不是太少?
    1. 要表达所有的语义
    2. 线性表达的表达效率低?
    3. 7168 hidden 向量既要表达当前需要生成的token，还要表达高级的语义总结抽象，短期和长期的目标矛盾
        1. 用于长期的hidden向量专门计算，并缩减维度尺寸（1024?)，可以减少KVcache和计算量
        2. 训练模型，自动在sequence中间生成特殊的token，用于表示抽象的高层目标
9. 使用自然作为输入输出的接口
    1. 能使用自然语言影响内部变量进行特定取舍的调整
    2. 能使用自然语言进行自动推理和迭代，输出更好的答案
10. 真正突破是打通了语言
    1. 可以理解和映射语言到其他的领域。
    2. 足够大的理解空间，上下文，可以处理一个足够复杂的问题

??

1. 记忆和人类不一致，而且没有统一的表达，不通用
2. 没有自主意识，还是在算概率，逃不开数学上的特征分割，虽然不能证明当前的数学基础、梯度下降是错误的，但是AGI肯定不是只有这些，AGI更多的是一种复杂的工程，而不是简单几个公式
3. Transformer架构个在处理长上下文时，会受到二次复杂度（浪费算力），以及长度外推能力弱的限制。

Attention

1. 每层每个token计算的输入是前面所有token的key和value
   1. 从信息流的角度来看，不是一个树形的拓扑
      1. 实际上一段自然语言通过字、词、短语、句子等层级结构组合出表达的语义
      2. 类比于卷积的空间约束，自然语言语义应该需要树形层级结构的约束
   2. 所以可以采用KV cache进行缓存并加速
2. 多层之间不能共享权重
3. 动态性欠缺
4. 随序列长度增加而变慢的attention机制
   1. 从信息量来说，句子长了，包含的信息肯定是变多了，空间复杂度在O(T)和O(1)之间
      1. Transformer 的时间复杂度为：O (T^2)、空间复杂度为：O (T^2)
      2. RWKV的固定大小的status存储器也是不合理的（不考虑外部记忆）,时间复杂度为：O(T)、空间复杂度为O(1)
   2. 相对于人脑，每个新的token都要重新计算一遍底层的语义，而不是直接根据前面语句的总结进行调整
   3. 缺乏更高维度的动态性
      1. MOE、CoT、稀疏都能提高动态性能力
5. 速度显存恒定的FFN全连接网络
6. 类比于人类记忆，“检索”和“更新”是两个步骤
   1. 梯度下降不能区分这两个步骤，只能通过优化器策略来调整?
   2. 两步骤的训练策略能更好的约束模型的梯度下降到更合理的最低值?
   3. 和binary神经元的问题一致：检索输入的选择（检索）和权重的更新
      1. 和attention的注意力+MLP机制类似，更新策略有提升空间
   4. 类似于Yan2.0 Preview基于的是RockAI首创的非Transformer架构
      1. 前向过程中，既能通过门控式更新保留长期依赖，又能基于输入分布特性灵活整合新知识
      2. 不同于「上下文工程」等方案对记忆信息的显性存储，RockAI将有效信息隐式地记忆到多层神经网络的权重，通过神经网络的多层级抽象、非线性建模等能力，实现更优的记忆性能
      3. 将记忆能力「注入」模型本身。它不再是一个外挂模块，而是模型的一部分
      4. 我：其实本质上和上下文（kv cache）没有区别，kvcache也是前面的存储信息决定了后面的选择，而且都是使用线性映射来做信息变换和选择

??API

1. transformer利用了语言的特性，在attention中间，使用自然语言作为通用的输入输出接口和表达
   1. 自然语言离散的Token效率比较低，作为CoT太慢，潜式思维链（Latent Chain-of-Thought）是一个改进方向
2. attention中间的输入输出都是用于表达语言的一个序列的特征。
3. 这个序列非常重要，因为序列都是由编码的token组成，可以非常灵活得表示一个非常复杂的语义
4. 那么这个跟语言其实是有异曲同工的效果，语言也使用
5. 很多个文字来表达一个语意，每个文字的空间不是很大
6. 搜索空间
   1. 就比如说汉字也就几千个字transformer相当于是在模仿语言，
   2. 把整个模型的搜索空间限制为自然语言这个尺度（约束），极大缩小了模型的搜索空间

?????

1. LLM 之所以在简单理解任务中无法提供准确且稳定答案，是因为这些模型缺乏对语言的真正理解：它们生成的词语如同语义「黑箱」，只是近似于语言的表面统计和解析过程中较「自动化」的部分。
2. LLM 并不适合作为语言理论，因为它们的表征能力几乎是无限的，这使得它们的表征既是任意的，又缺乏解释性基础，属于通用函数逼近器这一类别，而后者已被证明能够逼近任何数学函数。
3. 较自动化部分，表示抽象不够高级，只是对低级别的概念进行抽象，不能进行高级别的抽象。虽然级别低，但是 AI的信息容量足够大，和人类比，更像是记忆力很好的书呆子
4. 它们看似合理的表现，隐藏了语言建模方法本身固有的缺陷：智能实际上无法作为统计推断的副产品而自然产生，理解意义的能力也不能由此产生。

Revision #38
Created 2025-01-11 09:44:04 UTC by Colin
Updated 2026-05-05 14:38:29 UTC by Colin