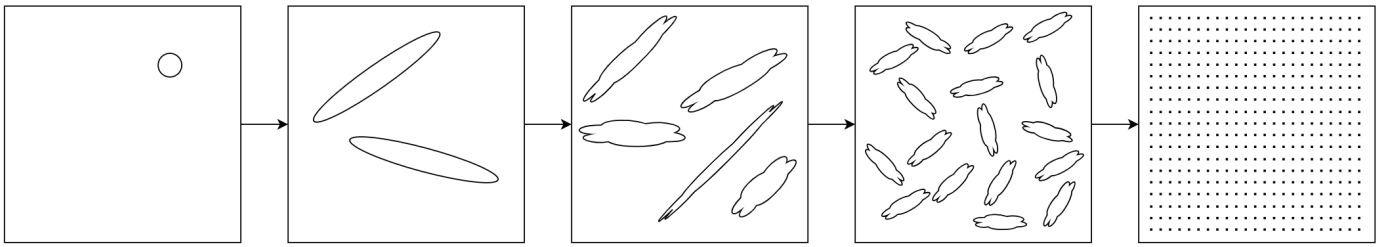
LLM???????

??

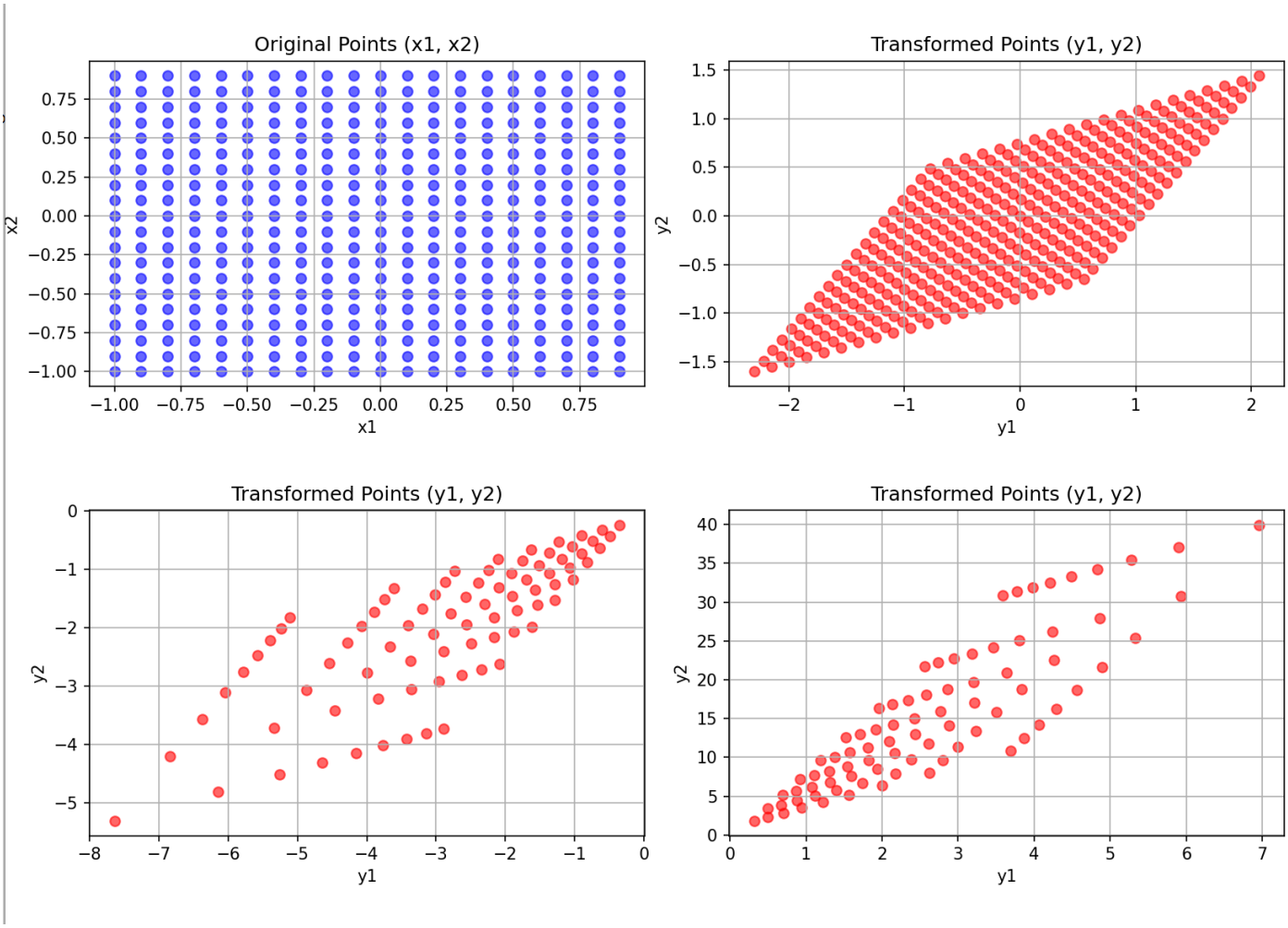
1. 两个32bit的浮点数乘法，相当于32位的bit向量做空间映射
2. 每个bit代表了特殊的含义，指数、尾数以及对应的档位
3. 乘法不能充分利用32位的所有表达空间，精度越低的数据信息利用率越高

?????????



1. 每一层的所有的hidden status的集合，表示了当前的所有信息（语义）
2. 每经过一层相当于把一种向量空间的表示，映射到另外一个向量空间
    1. 每一次梯度下降都优化一点映射关系，增加不同类的距离，减少同类的距离
    2. 对输入的**hidden status**理解得越深刻（抽象得越高级），映射分隔得越准确，输出的结果越好
3. 经过多次映射之后就能直接得到结果需要的index
4. 非常大的线性映射的参数：DeekseekV3 671B中，大头（98%）参数是，256专家*60层*44MB=670GB
5. 每个expert有三个linear，总参数量= hidden_size * moe_intermediate_size * 3层 = 7168 * 2048 * 3 = 44MB

???? ?????

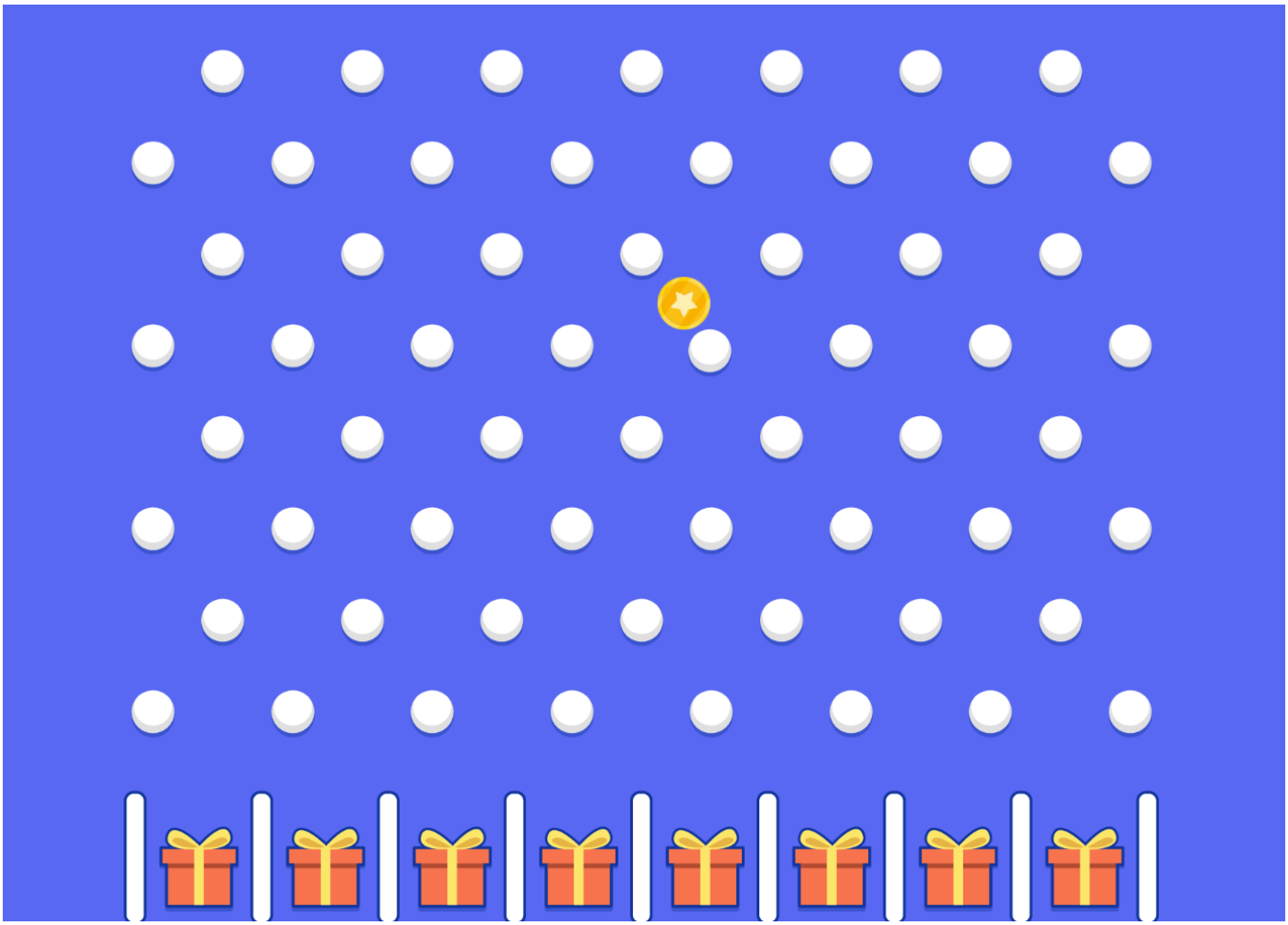


???Dot

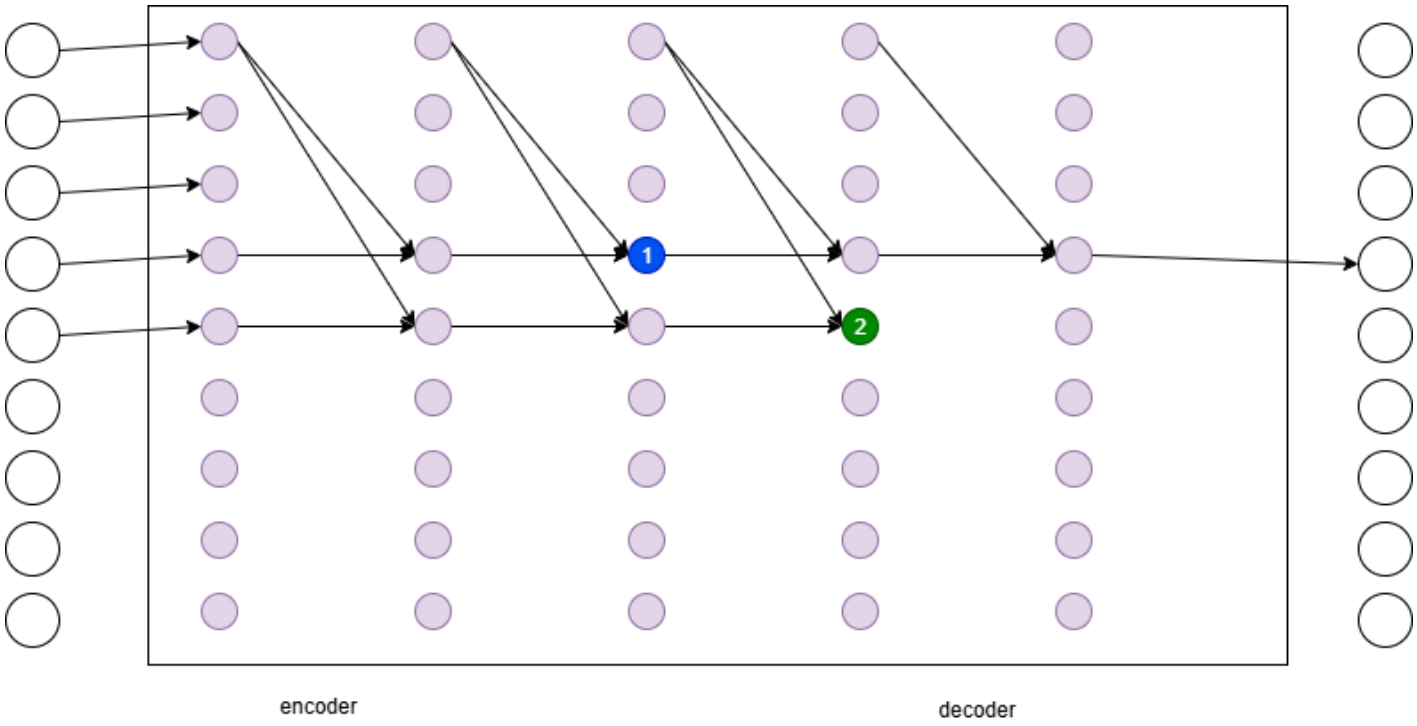
1. 对B进行非线性映射， A = B *(C+D) 等价于 A = B*C + B*D， ABCD都是矩阵
    1. 实际上增加了映射的空间灵活度，实验下来，使用得当可能可以提升精度
2. ResNet的典型 A = B*C + B
    1. 显然 D是 1，不对 B进行改变

LLM?????

1. 每一层都根据前面的数据映射到另外一个表示空间
2. 多层的 LLM可以等价于一个大型卡罗搜索树
    1. 剪枝的，均匀宽度的
    2. 大规模参数，多维度
3. 行为可以比喻为 Plinko（中文常译为“钉板游戏”或“弹珠盘”）
    1. 从上到下不断在转换在对应空间内的分类
    2. 当前 layer的所有 token的所有 hidden status的所有可能数值共同表示了所有的可能分类
    3. 最终的 hidden status都对应了一个 token
4. 宏观来说
    1. 每个 token的单体空间（不大）组合成的组合空间（很大）就是一句话
    2. LLM进行编码（编码可以没有）+ 解码 生成（映射）另外一个组合空间，也就是输出的一句话
    3. 因为组合空间非常大，需要按照空间的规律进行有限的映射



?????



1. 所有内部节点表示：同层的所有前面　　　　　　　token的总结（映射）的结果
    1. 矛盾：既要表达当前节点的局部语义，又要表达到当前　　　　　　　token的整体语义

Revision #18
Created 2025-07-27 06:46:14 UTC by Colin
Updated 2026-04-29 07:34:27 UTC by Colin