召回聚合

用户的查询意图往往是复杂多样的，可能涉及到不同的领域、主题和语义层面。因此，召回体系中通常通过多路召回的方式从不同角度去理解和满足用户的查询需求。此外，多路召回通过各召回通道并行计算可以在海量数据中能够快速响应，同时实现负载均衡。

在面对多个差异化设计的召回通道同时触发并返回文档的情况下，需要充分发挥各通道的优势，从大量召回候选中筛选整合以实现各通道互补，最终在保障的系统效率的同时，提高召回的准确率和业务目标的达成率，确保整体效果的最优化。召回聚合通过采用轻量级的算法策略对召回结果筛选，最终将数千篇文档传递给粗排模块进行更精细的排序操作。

蛇形合并

蛇形合并策略通过交替、蛇形的方式将来自不同通道的候选项进行融合，通常会先选择一种通道的候选项，然后交替选择其他通道的候选项。

假设有三个召回通道（A、B、C），那么可以按如下方式选取：

- 第1个候选项来自通道A
- 第2个候选项来自通道B
- 第3个候选项来自通道C
- 第4个候选项再从通道A选取，以此类推

RRF合并（Reciprocal Rank Fusion）

RRF 对来自不同召回通道的排序结果进行加权，赋予排名靠前的候选项更高的权重，最终生成一个综合排序。具体来说，对于每个检索结果，计算其在每个路径中的排名，然后取这些排名的倒数之和作为该结果的最终得分。

RRF 的计算公式如下：

$$\text{RRF Score}(d) = \sum_{i=1}^{k} \frac{1}{\text{Rank}_i(d)+K} \tag{1}$$

其中，d 表示召回结果，$\text{Rank}_i(d)$ 表示该结果在第 i 条路径中的排名，K 为调节参数（通常取值为60），用于平滑排名，使得即使排名较低的候选项也能获得一定的权重。

贝叶斯优化合并

贝叶斯调参可以帮助自动化调整融合策略中的参数，从而找到最优的合并方式。通过贝叶斯优化，我们可以根据召回合并过程中不同通道的重要性、权重等参数，寻找最佳的合并策略，以提升召回的质量和效率。

贝叶斯优化（Bayesian Optimization）是一种基于贝叶斯统计方法的全局优化算法，通常用于优化函数比较复杂、代价高昂、无法直接求导的情况。具体步骤如下：

1. **定义目标函数**：定义目标函数 $f(x)$ （x 是超参数）用于评估召回合并策略的效果
    -

$$f(\mathbf{w}) = \mathbf{Metric}(\mathbf{Merge}(\mathrm{Recalls}(\mathbf{w}))) \tag{2}$$

- $\mathrm{Recalls}(\mathbf{w}) = [R_1(w_1), R_2(w_2), \ldots, R_n(w_n)]$ 表示来自不同通道的召回结果，每个 $R_i(w_i)$ 是通道 i 根据权重 w_i 所返回的候选集
- $\mathrm{Merge}(\cdot)$ 表示将各个通道的召回结果按权重加权融合
- $\mathrm{Metric}(\cdot)$ 是根据合并后的候选集评估的性能指标

2. **设置参数空间**：如果使用加权平均的合并方式，可以将每个通道的权重作为调节参数进行优化
   - 比如，设定权重参数 $\mathbf{w} = [w_1, w_2, \cdots, w_n]$，每个召回通道的权重 w_i 影响候选项排名和选择
   -

$$\mathbf{w} = [w_1, w_2, \cdots, w_n] \quad \text{with} \quad \sum_{i=1}^{n} w_i = 1, \quad w_i \in [0, 1] \tag{3}$$

3. **初始化高斯过程**：为待优化的参数空间设定先验分布，通常使用高斯过程（Gaussian Process, GP）作为先验，表示对参数空间的初步认识
   -

$$p(f(\mathbf{w})) \sim \mathcal{GP}(m(\mathbf{w}), k(\mathbf{w}, \mathbf{w}')) \tag{4}$$

   - 其中 $m(\mathbf{w})$ 是目标函数的均值函数，通常假设为零；$k(\mathbf{w}, \mathbf{w}')$ 是协方差函数，用于表示不同参数配置之间的相似性
4. **更新后验分布**：在每次迭代中，根据当前的参数空间和目标函数值，更新高斯过程的后验分布，并基于当前的后验分布生成新的参数选择。贝叶斯调参使用获取函数（Acquisition Function）来选择下一个要评估的参数点。常见的获取函数如期望改进（Expected Improvement, EI）
   -

$$\alpha(\mathbf{w}) = \mathbb{E}[\Delta f(\mathbf{w})] = \mathbb{E}[\max(0, f(\mathbf{w}_{\text{best}}) - f(\mathbf{w}))] \tag{5}$$

   - $f(\mathbf{w}_{\text{best}})$ 是当前最好的目标函数值
   - $\Delta f(\mathbf{w})$ 是期望的改进
5. **选择下一个评估点**：通过最大化获取函数来决定下一个评估点。获取函数根据当前的后验分布选择一个最有可能提升目标函数值的参数组合，从而进行下一轮评估
   -

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \alpha(\mathbf{w}) \tag{6}$$

6. **重复迭代**：贝叶斯优化会根据每一轮评估的结果，调整先验分布，不断优化目标函数，最终找到全局最优的参数配置
   -

$$p(f(\mathbf{w}) \mid \mathbf{w}^*, f(\mathbf{w}^*)) \sim \mathcal{GP}(\mu(\mathbf{w}), \Sigma(\mathbf{w})) \tag{7}$$

最终，贝叶斯优化会返回最优的参数 $\mathbf{w}^*$。

此外，除了参数为各通道的权重，对于每篇文档的综合打分（Query-Doc相关性、Doc质量、Doc时效性等分数）的融合公式的权重参数也可以采用贝叶斯优化的方式设定，以此通过对Doc打分实现排序。

总结

在召回模块中，各召回通道通常会通过不同的算法、模型或策略获取候选文档。由于这些召回通道目标、算法和策略的差异，它们生成的候选项通常具有不同的质量和覆盖面。召回聚合的任务是将这些候选项通过一定的方式进行加权、排序和融合，动态控制各个通道的召回量配额，确保最终的候选集合既具有足够的多样性，又能保持较高的相关性。

参考文献

1. Reciprocal Rank Fusion outperforms Condorcet and individual Rank Learning Methods
2. Practical Bayesian Optimization of Machine Learning Algorithms