(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 103605493 A
(43)申请公布日 2014.02.26

(21)申请号 201310632348.8

(22)申请日 2013.11.29

(71)申请人 哈尔滨工业大学深圳研究生院
地址 518000 广东省深圳市南山区西丽镇深圳大学城哈工大校区

(72)发明人 叶允明 范希贤 黄晓辉

(74)专利代理机构 深圳市科吉华烽知识产权事务所(普通合伙) 44248
代理人 于标

(51)Int.Cl.
G06F 7/08(2006.01)

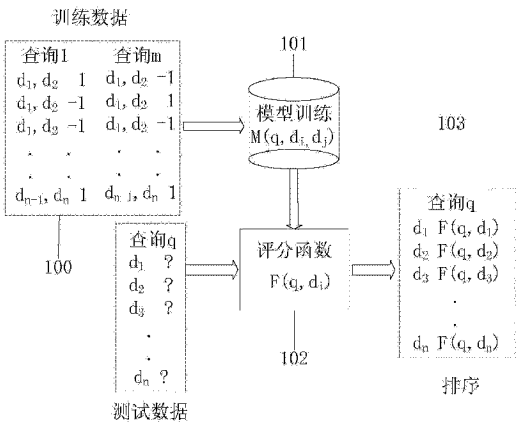
权利要求书2页 说明书7页 附图8页

(54)发明名称

基于图形处理单元的并行排序学习方法及系统

(57)摘要

本发明提供了一种基于图形处理单元的并行排序学习方法及系统，该并行排序学习方法包括构建查询及文档偏序对：针对每个查询，根据训练集中文档与查询的相关度构建出文档偏序对，每一个文档偏序对为一个模型的训练样本；模型参数训练：估计评分函数中关于每个特征的权重参数取值；文档评分：根据模型参数训练步骤中估计出的模型参数和文档评分函数计算每个文档的得分；文档排序：根据每个文档的得分，选择排序算法对文档进行排序，然后把排序后的结果提供给查询用户。本发明的有益效果是本发明的基于图形处理单元的并行排序学习方法及系统，提高排序学习中数据计算速度。



CN 103605493 A

1. 一种基于图形处理单元的并行排序学习方法，其特征在于，包括如下步骤：

构建查询及文档偏序对：针对每个查询，根据训练集中文档与查询的相关度构建出文档偏序对，每一个文档偏序对为一个模型的训练样本，根据每一个查询的相关度列表构建成文档偏序对集；

模型参数训练：根据文档偏序对集估计模型参数，通过估计评分函数中关于每个特征的权重参数取值；

文档评分：根据模型参数训练步骤中估计出的模型参数和文档评分函数计算每个文档的得分；

文档排序：根据每个文档的得分，选择排序算法对文档进行排序，然后把排序后的结果提供给查询用户。

2. 根据权利要求 1 所述的并行排序学习方法，其特征在于：在所述构建查询及文档偏序对步骤中，在训练样本中，每一个查询对应一个文档列表，列表中给出文档与查询语句的相关度；在所述文档评分步骤中，采用线性评分模型，其评分模型函数为 $F(\Theta,d_j)=\sum_i \Theta_i f_{ji}$，公式中，$\Theta_i$ 为模型参数向量中的第 i 维，f_{ji} 文档 d_j 中的第 i 个特征值；在所述文档排序步骤中，采用双调排序对文档进行排序。

3. 根据权利要求 1 所述的并行排序学习方法，其特征在于：在所述模型参数训练步骤中，采用最大似然方法估计模型参数，似然函数为：

$$\prod_{q\in Q} p(\succ_q|\Theta)=\prod_{(q,i,j\in D_S)} p(i\succ_q j|\Theta)^{\delta((q,i,j)\in D_S)}\cdot(1-p(i\succ_q j|\Theta)^{\delta((q,j,i)\in D_S)})$$

公式中，q 是查询集 Q 中的一个查询，Θ 为要估计的模型参数，i、j 分别为第 i、j 个文档，(q, i, j) 表示第 q 个查询项，由第 i、j 个文档组成的查询及文档偏序对，D_S 为所有查询及文档偏序对样本集，δ 为指示函数，$p(\succ_q|\Theta)$ 为在模型参 Θ 下产生查询及文档偏序对 $\succ_q$ 的概率。

4. 根据权利要求 1 所述的并行排序学习方法，其特征在于：在所述构建查询及文档偏序对步骤中，针对每个查询开启一个核函数的线程，采用基于图形处理的多线程并行构建查询及文档偏序对；在所述模型参数训练步骤中，针对于每个查询及文档偏序对开启一个核函数的线程，采用基于图形处理的多线程并行更新模型参数；在文档评分步骤中，针对于每个文档开启一个核函数的线程，采用基于图形处理的多线程并行文档评分。

5. 一种基于图形处理单元的并行排序学习系统，其特征在于，包括：并行构建查询及文档偏序对模块：用于针对每个查询，根据训练集中文档与查询的相关度构建出文档偏序对，每一个文档偏序对为一个模型的训练样本，根据每一个查询的相关度列表构建成文档偏序对集；

并行模型参数训练模块：用于根据文档偏序对集估计模型参数，通过估计评分函数中关于每个特征的权重参数取值；

并行文档评分模块：用于根据模型参数训练步骤中估计出的模型参数和文档评分函数计算每个文档的得分；

并行文档排序模块：用于根据每个文档的得分，选择排序算法对文档进行排序，然后把排序后的结果提供给查询用户。

6. 根据权利要求 5 所述的并行排序学习系统，其特征在于：在所述并行构建查询及文档偏序对模块中，在训练样本中，每一个查询对应一个文档列表，列表中给出文档与查询语句的相关度；在所述文档评分步骤中，采用线性评分模型，其评分模型函数为 $F(\Theta,d_j)=\sum_i \Theta_i f_{ji}$，公式中，$\Theta_i$ 为模型参数向量中的第 i 维，f_{ji} 文档 d_j 中的第 i 个特征值；在所述文档排序步骤中，采用双调排序对文档进行排序。

7. 根据权利要求 5 所述的并行排序学习系统，其特征在于：在所述并行模型参数训练模块中，采用最大似然方法估计模型参数，似然函数为：

$$\prod_{q\in Q} p(\succ_q|\Theta)=\prod_{(q,i,j\in D_S)} p(i\succ_q j|\Theta)^{\delta((q,i,j)\in D_S)}.(1-p(i\succ_q j|\Theta)^{\delta((q,j,i)\in D_S)})$$

公式中，q 是查询集 Q 中的一个查询，Θ 为要估计的模型参数，i、j 分别为第 i、j 个文档，(q, i, j) 表示第 q 个查询项，由第 i、j 个文档组成的查询及文档偏序对，D_S 为所有查询及文档偏序对样本集，δ 为指示函数，$p(\succ_q|\Theta)$ 为在模型参 Θ 下产生查询及文档偏序对 $\succ_q$ 的概率。

8. 根据权利要求 5 所述的并行排序学习系统，其特征在于：在所述并行构建查询及文档偏序对模块中，针对每个查询开启一个核函数的线程，采用基于图形处理的多线程并行构建查询及文档偏序对；在所述并行模型参数训练模块中，针对于每个查询及文档偏序对开启一个核函数的线程，采用基于图形处理的多线程并行更新模型参数；在并行文档评分模块中，针对于每个文档开启一个核函数的线程，采用基于图形处理的多线程并行文档评分。

9. 根据权利要求 5 至 8 任一项所述的并行排序学习系统，其特征在于：该并行排序学习系统采用 CPU 和 GPU 协作框架设计，串行执行代码运行在 CPU 上，并行执行代码运行在 GPU 上，通过 GPU 提供的数据传输方式来交换显存与内存之间的数据，所述并行构建查询及文档偏序对模块、所述并行模型参数训练模块、所述并行文档评分模块、所述并行文档排序模块均运行在所述 GPU 上。

10. 根据权利要求 9 所述的并行排序学习系统，其特征在于：CPU 控制系统的调度给 GPU 分配任务，为 GPU 准备运行空间，GPU 在 CPU 准备好的环境下并行执行计算任务。

基于图形处理单元的并行排序学习方法及系统

技术领域

[0001] 本发明涉及基于互联网的数据处理方法及系统，尤其涉及基于图形处理单元的并行排序学习方法及系统。

背景技术

[0002] 随着网络技术的发展，信息获取变得越来越容易．但从海量且日新月异的互联网上检索信息，在检索过程中还要满足用户所需的响应时间和结果准确度，变得越来越困难。搜索引擎是从海量数据获取有用的信息的一个重要手段．而如何为用户返回与其查询最相关的信息，是搜索引擎发展和吸引用户的一个重要决定因素。

[0003] 商业搜索引擎和推荐系统普遍存在排序问题，互联网搜索引擎提供商的竞争日趋白热化，搜索引擎对于任意查询能有 TB 甚至 PB 量级的规模，每天可能达到亿次级的查询规模。每次查询的返回结果靠人工专家去分类判定然后给出排序结果是不现实的，排序最终归为人工智能问题。

[0004] 排序学习是一种机器学习任务：查询集和每个查询的一系列文档作为输入，通过训练一个系统在未知等级的测试集上获取最优化的预计排名作为输出。排序学习的提出在互联网搜索、商务网站推荐等领域都引起研究工作者的兴趣与深入研究。研究人员在研究信息检索中发掘各种新问题新技术并在历届的 SIGIR 会议上发表探讨，近些年来，排序学习在该会议上是一个热门的研究问题，同时互联网大规模的信息对于排序学习算法的性能是一个重大的挑战，也是排序学习算法后续研究工作的一个方向。

[0005] 但是，目前技术在排序学习中出现了由于海量数据导致计算速度慢的问题。

发明内容

[0006] 为了解决现有技术中的问题，本发明提供了一种基于图形处理单元的并行排序学习方法。

[0007] 本发明提供了一种基于图形处理单元的并行排序学习方法，包括如下步骤：

[0008] 构建查询及文档偏序对：针对每个查询，根据训练集中文档与查询的相关度构建出文档偏序对，每一个文档偏序对为一个模型的训练样本，根据每一个查询的相关度列表构建成文档偏序对集；

[0009] 模型参数训练：根据文档偏序对集估计模型参数，通过估计评分函数中关于每个特征的权重参数取值；

[0010] 文档评分：根据模型参数训练步骤中估计出的模型参数和文档评分函数计算每个文档的得分；

[0011] 文档排序：根据每个文档的得分，选择排序算法对文档进行排序，然后把排序后的结果提供给查询用户。

[0012] 作为本发明的进一步改进，在所述构建查询及文档偏序对步骤中，在训练样本中，每一个查询对应一个文档列表，列表中给出文档与查询语句的相关度；在所述文档评分步

骤中，采用线性评分模型，其评分模型函数为 $F(\Theta,d_j)=\sum_i \Theta_i f_{ji}$，公式中，$\Theta_i$ 为模型参数向量中的第 i 维，f_{ji} 文档 d_j 中的第 i 个特征值；在所述文档排序步骤中，采用双调排序对文档进行排序。

[0013] 作为本发明的进一步改进，在所述模型参数训练步骤中，采用最大似然方法估计模型参数，似然函数为：

[0014]

$$\prod_{q\in Q} p(\succ_q|\Theta)=\prod_{(q,i,j\in D_s)} p(i\succ_q j|\Theta)^{\delta((q,i,j)\in D_S)}.(1-p(i\succ_q j|\Theta)^{\delta((q,j,i)\in D_S)})$$

[0015] 公式中，q 是查询集 Q 中的一个查询，Θ 为要估计的模型参数，i、j 分别为第 i、j 个文档，(q, i, j) 表示第 q 个查询项，由第 i、j 个文档组成的查询及文档偏序对，D_S 为所有查询及文档偏序对样本集，δ 为指示函数，$p(\succ_q|\Theta)$ 为在模型参 Θ 下产生查询及文档偏序对 $\succ_q$ 的概率。

[0016] 作为本发明的进一步改进，在所述构建查询及文档偏序对步骤中，针对每个查询开启一个核函数的线程，采用基于图形处理的多线程并行构建查询及文档偏序对；在所述模型参数训练步骤中，针对于每个查询及文档偏序对开启一个核函数的线程，采用基于图形处理的多线程并行更新模型参数；在文档评分步骤中，针对于每个文档开启一个核函数的线程，采用基于图形处理的多线程并行文档评分。

[0017] 本发明还公开了一种基于图形处理单元的并行排序学习系统，包括：

[0018] 并行构建查询及文档偏序对模块：用于针对每个查询，根据训练集中文档与查询的相关度构建出文档偏序对，每一个文档偏序对为一个模型的训练样本，根据每一个查询的相关度列表构建成文档偏序对集；

[0019] 并行模型参数训练模块：根据文档偏序对集估计模型参数，通过用于估计评分函数中关于每个特征的权重参数取值；

[0020] 并行文档评分模块：用于根据模型参数训练步骤中估计出的模型参数和文档评分函数计算每个文档的得分；

[0021] 并行文档排序模块：用于根据每个文档的得分，选择排序算法对文档进行排序，然后把排序后的结果提供给查询用户。

[0022] 作为本发明的进一步改进，在所述并行构建查询及文档偏序对模块中，在训练样本中，每一个查询对应一个文档列表，列表中给出文档与查询语句的相关度；在所述文档评分步骤中，采用线性评分模型，其评分模型函数为 $F(\Theta,d_j)=\sum_i \Theta_i f_{ji}$，公式中，$\Theta_i$ 为模型参数向量中的第 i 维，f_{ji} 文档 d_j 中的第 i 个特征值；在所述文档排序步骤中，采用双调排序对文档进行排序。

[0023] 作为本发明的进一步改进，在所述并行模型参数训练模块中，采用最大似然方法估计模型参数，似然函数为：

[0024]

$$\prod_{q\in Q} p(\succ_q|\Theta)=\prod_{(q,i,j\in D_s)} p(i\succ_q j|\Theta)^{\delta((q,i,j)\in D_S)}.(1-p(i\succ_q j|\Theta)^{\delta((q,j,i)\in D_S)})$$

[0025] 公式中，q 是查询集 Q 中的一个查询，Θ 为要估计的模型参数，i、j 分别为第 i、j 个文档，(q, i, j) 表示第 q 个查询项，由第 i、j 个文档组成的查询及文档偏序对，D_S 为所有查询及文档偏序对样本集，δ 为指示函数，$p(\succ_q|\Theta)$ 为在模型参 Θ 下产生查询及文档偏序对 $\succ_q$ 的概率。

[0026] 作为本发明的进一步改进，在所述并行构建查询及文档偏序对模块中，针对每个查询开启一个核函数的线程，采用基于图形处理的多线程并行构建查询及文档偏序对；在所述并行模型参数训练模块中，针对于每个查询及文档偏序对开启一个核函数的线程，采用基于图形处理的多线程并行更新模型参数；在并行文档评分模块中，针对于每个文档开启一个核函数的线程，采用基于图形处理的多线程并行文档评分。

[0027] 作为本发明的进一步改进，该并行排序学习系统采用 CPU 和 GPU 协作框架设计，串行执行代码运行在 CPU 上，并行执行代码运行在 GPU 上，通过 GPU 提供的数据传输方式来交换显存与内存之间的数据，所述并行构建查询及文档偏序对模块、所述并行模型参数训练模块、所述并行文档评分模块、所述并行文档排序模块均运行在所述 GPU 上。

[0028] 作为本发明的进一步改进，CPU 控制系统的调度给 GPU 分配任务，为 GPU 准备运行空间，GPU 在 CPU 准备好的环境下并行执行计算任务。

[0029] 本发明的有益效果是：本发明的基于图形处理单元的并行排序学习方法及系统，提高排序学习中数据计算速度。

附图说明

[0030] 图 1 是本发明的并行排序学习模型的系统框图。

[0031] 图 2 是本发明的训练集原始数据示意图。

[0032] 图 3 是本发明的原始查询训练集转化为文档偏序对集示意图。

[0033] 图 4 为本发明的 CPU 及 GPU 硬件架构图。

[0034] 图 5 为本发明的模块图。

[0035] 图 6 为本发明的并行排序学习方法的 CPU 和 GPU 协作框架示意图。

[0036] 图 7 为本发明的多线程构建查询及文档偏序对流程图。

[0037] 图 8 为本发明的多线程模型参数更新流程图。

[0038] 图 9 为本发明的多线程文档评分流程图。

[0039] 图 10 为本发明使用的双调排序流程图。

具体实施方式

[0040] 如图 1 所示，本发明公开了一种基于图形处理单元的并行排序学习方法，包括如下步骤：

[0041] 100 构建查询及文档偏序对：针对每个查询，根据训练集中文档与查询的相关度构建出文档偏序对，每一个文档偏序对为一个模型的训练样本。

[0042] 具体实施过程如下：基于偏序对的排序学习算法的主要思想是，对于任一个查询，对任意两个不同相关度的文档中，都可以得到一个训练实例对。在训练模型时，要使得二类分类的误差最小，即尽可能的分对所有文档偏序对。

[0043] 在训练样本中，每一个查询对应一个文档列表，列表中给出文档与查询语句的相关度，如图 2，其中$d_i^{(j)}$表示在第 j 个查询中的第 i 个文档，$y_i^{(j)}$表示第 i 个文档与第 j 个查询的相关度，n 为文档数目，m 为查询数目。图 3 为根据查询 q 下两个文档间的相关度大小，得到一个文档间的相关度大小比较结果示意图。图中任一个小格表示一个文档偏序对，即模型的一个训练样本。由于用户更关心的是相关度高的文档排在前面，目标优化是使得相关度高的文档尽可能的预测正确。本文发明实施过程中采用大于偏序关系，如图 3 所示$\succ_q$，大于偏序关系用 1 表示，小于关系用 -1 表示。

[0044] 101 模型参数训练：模型训练是本发明中最重要的一步．模型训练的目的是估计评分函数中每个特征的权重参数取值，本发明采用的是最大似然参数估计对贝叶斯个性化排序学习模型的参数进行估计。

[0045] 具体实施过程如下：贝叶斯个性化排序学习模型训练的目的是要估计评分函数中关于每个特征的权重参数取值，最大似然估计和贝叶斯参数估计是常用的办法。最大似然估计相对于贝叶斯参数估计有收敛性好，简单易用等优点。因此，本发明实施中采用最大似然方法估计模型参数。最大似然估计是把要预测的参数看作是已知的量，但取值未知，最后使得模型符合训练样本的概率最大的一系列值为所要的参数值。

[0046] 模型训练是在给定查询集合下，通过最大化后验概率模型为每个文档中找出其正确的排名。然后，根据模型对未标注样本进行等级预测。本发明假设结果文档集合中的文档相关度等级符合某种概率分布，表示为 p(Θ)。由贝叶斯公式得到后验概率可表示为：

[0047]

$$P(\Theta|\succ_q)=\frac{p(\succ_q|\Theta)P(\Theta)}{p(\succ_q)}$$

[0048] 公式中，Θ 为模型参数，$\succ_q$为一个查询及文档偏序对样本。由于在给定训练集下，$p(\succ_q)$可以看成一个常量，因此可以得到概率模型

[0049]

$$P(\Theta|\succ_q)\propto P(\succ_q|\Theta)p(\Theta)$$

。

[0050] 本发明假定两两查询是相对独立的，并对于每个查询，每一对文档之间也是相互独立的。因此对于所有查询 q ∈ Q 的所有输入样本对，上式的似然估计函数$P(\succ_q|\Theta)$可以表示为所有输入样本对的乘积，数学形式表示为公式

[0051]

$$\prod_{q\in Q}p(\succ_q|\Theta)=\prod_{(q,i,j\in D_S)}p(i\succ_q j|\Theta)^{\delta((q,i,j)\in D_S)}\cdot(1-p(i\succ_q j|\Theta)^{\delta((q,j,i)\in D_S)})$$

[0052] 公式中，q 是查询集 Q 中的一个查询，Θ 为要估计的模型参数，i、j 分别为第 i、j 个文档，(q, i, j) 表示第 q 个查询下，由第 i、j 个文档组成的查询及文档偏序对，D_S 为所有查询及文档偏序对样本集，$p(\succ_q|\Theta)$为在模型参 Θ 下产生查询及文档偏序对$\succ_q$的概率。δ 是一个指示函数，表示为公式

[0053]

$$\delta(b) := \begin{cases} 1 & b\text{为真} \\ 0 & \text{否则} \end{cases}$$

。

[0054] 由于在具体实施中，本发明之采用大于偏序关系，即采用的所有指示函数 δ(b) 为真的偏序关系。因此似然函数可以简写为

[0055]

$$\prod_{q\in Q} p(\succ_q|\Theta) = \prod_{(q,i,j)\in D_s} p(i \succ_q j|\Theta)$$

。

[0056] 在本发明中，定义产生文档偏序对的概率为

[0057]

$$p(i \succ_q j|\Theta) = \frac{1}{1+e^{-x_{qij}}}$$

。

[0058] 其中 $x_{qij}(\Theta)=F(\Theta,d_i)-F(\Theta,d_j)$，表示在参数为 Θ 下，文档 d_i 与文档 d_j 的评分之差。评分函数 $F(\Theta,d_i)$ 将在文档评分步骤中介绍。

[0059] 参数估计中，具体的概率 $p(\Theta)$ 未知，但假设其参数形式是已知的，唯一未知的是参数向量 Θ 的值，这也是最大似然估计的基本思想。本发明种假设 $p(\Theta)$ 符合 0 均值，协方差矩阵为 Σ_Θ 的正态分布，数学形式表示为公式：

[0060] $p(\Theta) \sim N(0, \Sigma_\Theta)$

[0061] 结合高斯密度函数上述公式可转换为公式：

[0062] $$p(\Theta) = \frac{1}{\sqrt{2\pi}\sigma}\exp[-\frac{1}{2}\frac{\Theta^2}{\sigma}]$$

[0063] 公式中 σ 为正态分布标准差。本发明中设为 $\Sigma_\Theta=\lambda_\Theta I$，Θ 为模型的参数向量。通过最大化似然估计来获得最优化的检索排序结果过程可形式化为公式

[0064]

$$\begin{aligned}\prod_{q\in Q} p(\succ_q|\Theta) &:= \ln P(\Theta|\succ_q) \\ &= \ln \prod_{(q,i,j)\in D_s} p(i \succ_q j|\Theta)\cdot p(\Theta) \\ &= \ln \prod_{(q,i,j)\in D_s} \frac{1}{1+e^{-x_{qij}}}\cdot p(\Theta) \\ &= \sum_{(q,i,j)\in D_s} \ln\frac{1}{1+e^{-x_{qij}}} + \ln p(\Theta) \\ &= \sum_{(q,i,j)\in D_s} \ln\frac{1}{1+e^{-x_{qij}}} - \lambda\|\Theta\|^2\end{aligned}$$

。

[0065] 采用梯度下降方法对上面似然函数进行参数估计，对上面公式求导得

[0066]

$$\begin{aligned}\frac{\partial \prod_{q\in Q} p(\succ_q|\Theta)}{\partial\Theta} &= \sum_{(q,i,j)\in D_s} \frac{\partial}{\partial\Theta}\ln\frac{1}{1+e^{-x_{qij}}} - \lambda\frac{\partial}{\partial\Theta}\|\Theta\|^2 \\ &= \sum_{(q,i,j)\in D_s} \frac{1}{1+e^{x_{qij}}}\frac{\partial}{\partial\Theta}\frac{1}{1+e^{-x_{qij}}} - \lambda\Theta\end{aligned}$$

。

[0067] 梯度下降法每次迭代的前进方向是由其梯度相反方向决定，使得每次迭代都能使目标函数逐步收敛。梯度下降算法首先对 Θ 随机赋值，根据训练样本改变 Θ 的值，使的目标函数按梯度下降的方向进行收敛，直到满足算法结束条件，算法终止。

[0068] 102 文档评分：即根据 101 步骤中估计出的模型参数和文档评分函数计算每个文档的得分。

[0069] 具体实施过程如下：在文档评分步骤中，本发明采用线性评分排序学习模型（Linear Scoring Learning to Rank Model，LSLRM），其评分模型函数设计为

[0070] $$F(\Theta,d_j)=\sum_i \Theta_i f_{ji}$$

[0071] 公式中，Θ_i 为模型参数向量中的第 i 维，f_{ji} 文档 d_j 中的第 i 个特征值。

[0072] 103 文档排序：根据每个文档的得分，选择合适的排序算法对文档进行排序，然后把排序后的结果提供给查询用户。

[0073] 具体实施过程如下：在本实施过程中，采用了双调排序。对于双调排序，首先要建立一个双调序列。如果把一个有序序列由小到大、另一个有序序列从大到小接在一起，就构成了一个双调序列。因此所谓双调序列是指序列要么先单调递增然后再单调递减，要么先单调递减然后又单调递增。然后进行双调归并，也就是将双调序列不断的划分，分成若干个小的子双调序列，这就是双调归并的过程。在本实施中采用双调排序是为了方法后面的并行化过程。

[0074] 在本发明中还构建一种基于图形处理单元的并行排序学习系统，包括硬件部分和软件部分，硬件部分：采用 CPU 及 GPU 协作框架设计，串行执行代码运行在 CPU 上，并行执行代码运行在 GPU 上，通过 GPU 提供的数据传输方式来交换显存与内存之间的数据；软件部分分为四个模块，包括并行构建查询及文档偏序对模块，并行模型参数训练模块，并行文档评分模块和并行文档排序模块四个部分。所述并行构建查询及文档偏序对模块是根据每一个查询的相关度列表构建成文档偏序对集。所述并行模型参数训练模块是根据查询及文档偏序对集，估计出模型参数．每一个文档偏序对作为一个样本参与参数估计。所述并行文档评分模块是根据模型参数和待排序文档特征值，通过评分函数进行计算每个文档的得分。所述并行文档排序模块是采用并行化排序方法，根据文档得分，对文档进行排序。

[0075] 具体实施过程如下：该并行排序学习系统采用 CPU 及 GPU 框架的设计，如图 4 为系统的硬件框架，CPU 控制系统的调度，给图形处理单元分配任务，为图形处理单元准备运行空间等，图形处理单元在 CPU 准备好的环境下，并行执行计算任务。图 5 为系统模块框图，系统分为四个并行化模块，包括并行构建查询及文档偏序对模块，并行模型参数训练模块，并行文档评分模块和并行文档排序模块。图 6 为本发明基于图形处理单元的并行排序学习系统的软件协作框架，系统利用统一计算设备架构（Compute Unified Device Architecture，简称“CUDA”）编程平台对排序学习算法过程进行加速。

[0076] 在基于 CPU 及 GPU 协作框架的设计中，通过对 CPU 和 GPU 的协作任务进行合理的分配和框架设计，充分利用 CPU 和 GPU 的各自优势，为算法进行加速。本系统将其任务分为两部分来进行分配，一部分是在 CPU 上具有明显运行优势的任务，一部分是在图形处理单元上明显具有运行优势的任务。适合在 CPU 上运行的任务主要包括：模型初始化，数据的 I 及 O 操作，算法逻辑流程的控制，核函数的调用。适合在图形处理单元上运行的任务主要是

数据运算类任务包括：并行构建查询及文档偏序对，针对每个文档训练模型参数，文档评分和对文档排序。

[0077] 在系统软件方面，主要通过为各模块设计核函数来实现算法的加速运行。在并行构建查询及文档偏序对模块中，系统设计一个核函数，该核函数为每个查询在图形处理单元上分配一个线程，共开启 m 个线程，m 为训练集上的查询数，构建出所有的查询及文档偏序对集，其核函数的计算流程为图 7，在图 7 中，对查询 q 来说，文档 i 的相关性高于文档 j。所以，输出文档篇序对 <q, i, j>，y_i^q 代表对于查询 q，文档 i 的相关性。

[0078] 在并行模型参数训练模块中，系统为该模块设计了一个核函数更新模型参数。如图 8，系统为该模块申请与偏序对同样数量的线程。每个线程针对一个文档偏序对进行更新模型参数。每一轮都要针对所有的样本更新一次，然后再 CPU 对所有模型参数进行合并。在图 8 中，如下公式的 $\Theta \leftarrow \Theta + \alpha(\frac{1}{1+e^{\hat{x}_{qij}}}\frac{\partial}{\partial\Theta}\hat{x}_{qij} - \lambda_\Theta\Theta)$ 的含义为：$\frac{1}{1+e^{\hat{x}_{qij}}}\frac{\partial}{\partial\Theta}\hat{x}_{qij} - \lambda_\Theta\Theta$ 是似然函数关于 Θ 的梯度（推导过程见 101 模型参数训练），α 是梯度下降的步长参数。该公式为采用梯度下降法求 Θ 的值。

[0079] 在并行文档评分模块中，系统为该模块设计了一个核函数计算每个文档的得分，如图 9。系统为每个文档开设一个线程，多线程并行计算文档得分。在图 9 中，该 $F(\vec{\Theta}, \vec{d_j}) = \sum_i \Theta_i f_{ji}$ 是一个评分函数，即，根据模型参数训练模块中估计出参数 Θ 的值，对文档 d_j 进行评分，f_{ji} 代表文档 j 的第 i 个特征的值。该评分结果用于文档排序模块对文档进行排序。

[0080] 在并行文档排序模块中，系统采用适合于 GPU 计算的双调排序，其过程如图 10。

[0081] 本发明的提出了一种基于图形处理单元的并行排序学习方法及系统。同时，利用图形处理单元 (GPU) 和中央处理器 (CPU) 之间的计算能力的互补性，本发明设计了一套基于 CPU 及 GPU 协作框架的并行化排序学习系统。系统硬件部分设计为 CPU 及 GPU 协作框架，软件部分设计分四个模块：并行构建查询及文档偏序对，并行模型参数训练，并行文档评分和并行文档排序．本发明的基于图形处理单元的贝叶斯个性化并行化排序学习方法及系统，可以充分利用图形处理设备的高并行性，有效的提高算法的排序性能，非常适合于处理大规模的排序学习问题。

[0082] 以上内容是结合具体的优选实施方式对本发明所作的进一步详细说明，不能认定本发明的具体实施只局限于这些说明。对于本发明所属技术领域的普通技术人员来说，在不脱离本发明构思的前提下，还可以做出若干简单推演或替换，都应当视为属于本发明的保护范围。

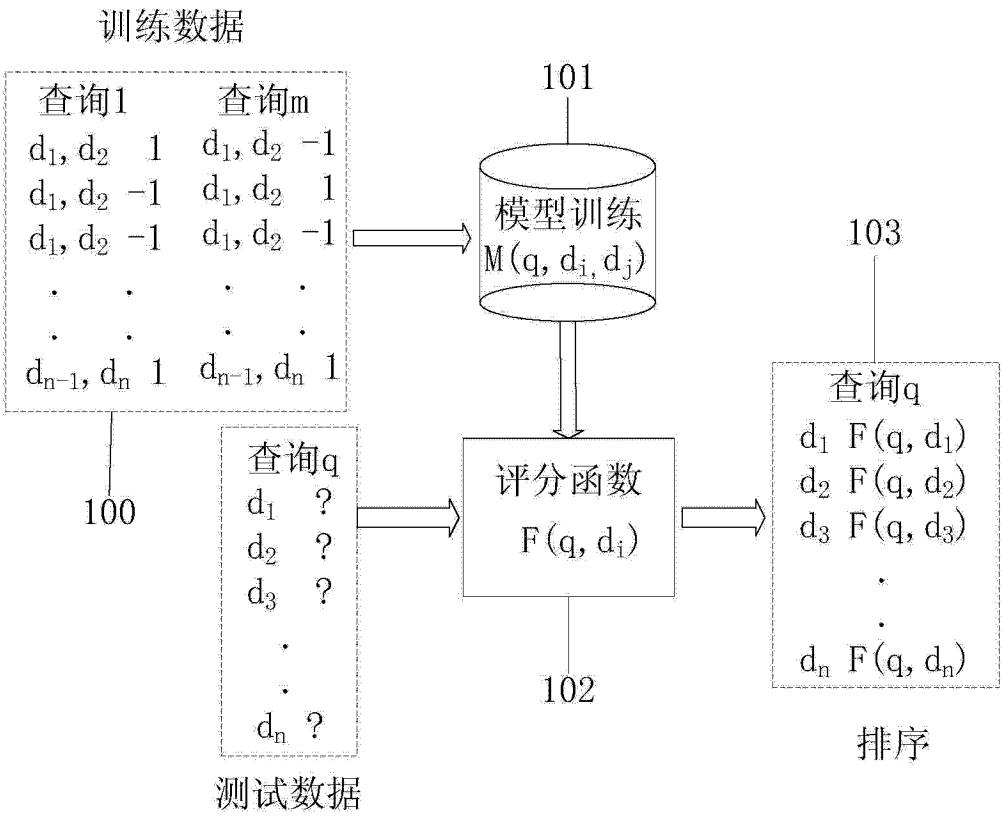

训练数据
查询1 查询m
d1, d2 1 d1, d2 -1
d1, d2 -1 d1, d2 1
d1, d2 -1 d1, d2 -1
dn-1, dn 1 dn-1, dn 1
100
101
模型训练
M(q, di, dj)
查询q
d1 ?
d2 ?
d3 ?
dn ?
测试数据
评分函数
F(q, di)
102
103
查询q
d1 F(q, d1)
d2 F(q, d2)
d3 F(q, d3)
dn F(q, dn)
排序

图 1

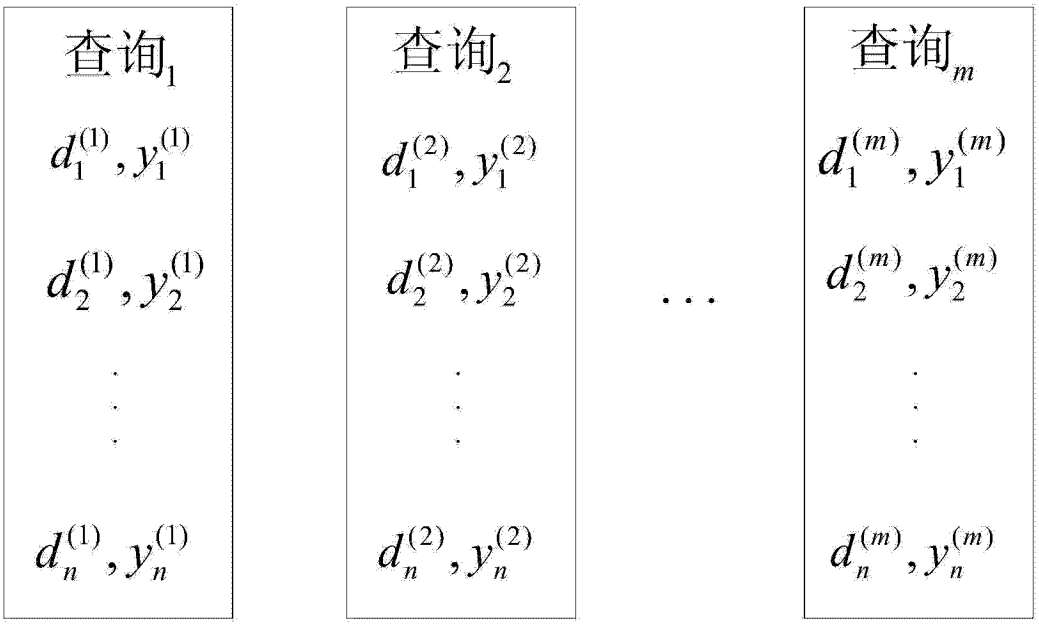

查询1
查询2
查询m

图 2

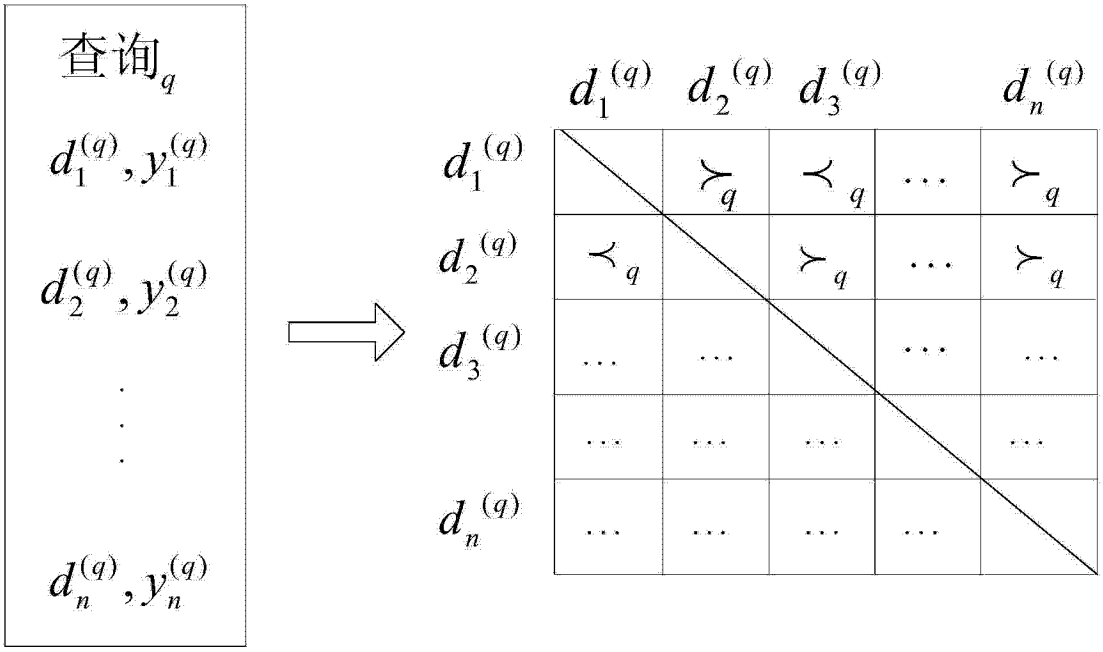

查询$_q$
$d_1^{(q)}, y_1^{(q)}$
$d_2^{(q)}, y_2^{(q)}$
$d_n^{(q)}, y_n^{(q)}$
$d_1^{(q)}$ $d_2^{(q)}$ $d_3^{(q)}$ $d_n^{(q)}$
$d_1^{(q)}$
$d_2^{(q)}$
$d_3^{(q)}$
$d_n^{(q)}$
$\succ_q$
…

图 3

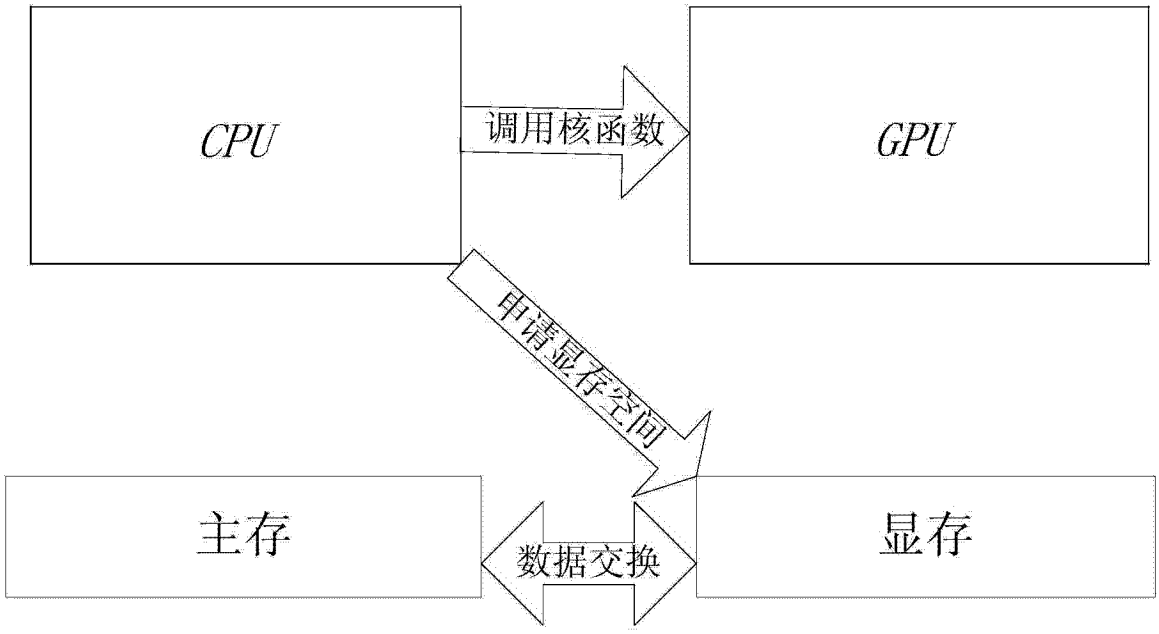

CPU
调用核函数
GPU
申请显存空间
主存
数据交换
显存

图 4

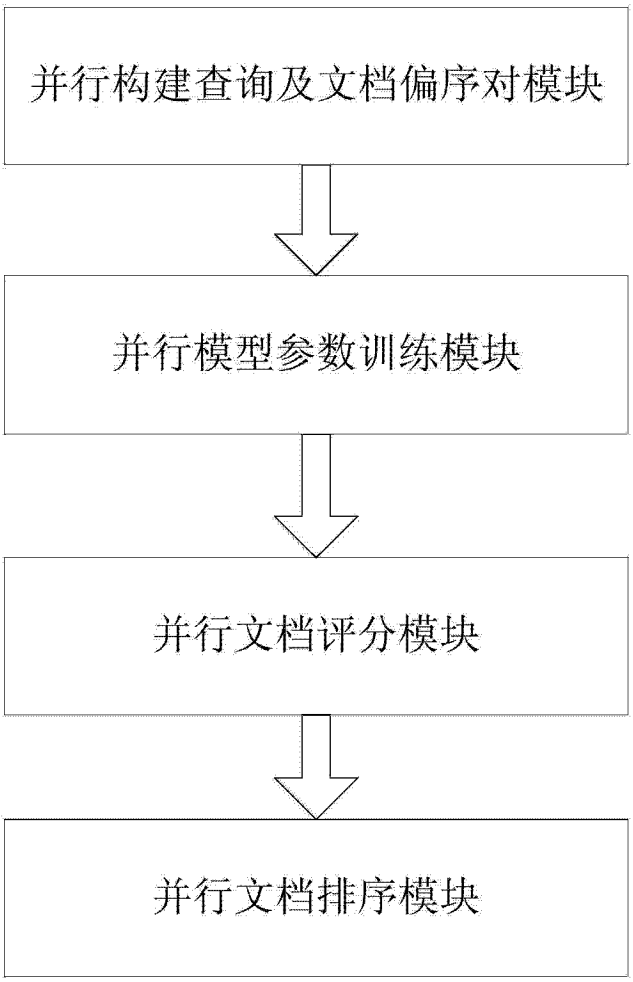


图 5

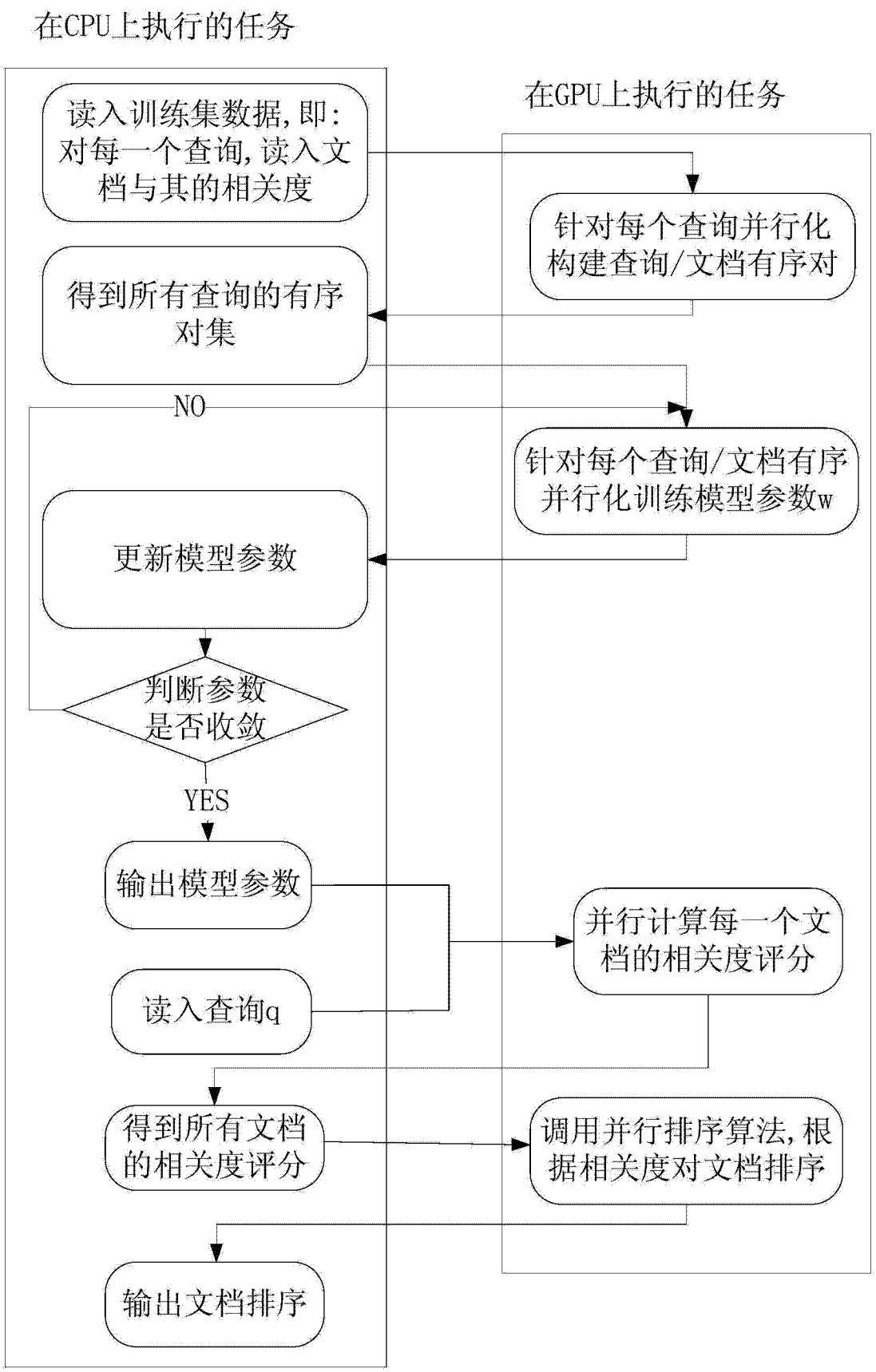

在CPU上执行的任务
在GPU上执行的任务
读入训练集数据,即:对每一个查询,读入文档与其的相关度
针对每个查询并行化构建查询/文档有序对
得到所有查询的有序对集
NO
针对每个查询/文档有序并行化训练模型参数w
更新模型参数
判断参数是否收敛
YES
输出模型参数
并行计算每一个文档的相关度评分
读入查询q
得到所有文档的相关度评分
调用并行排序算法,根据相关度对文档排序
输出文档排序

图 6

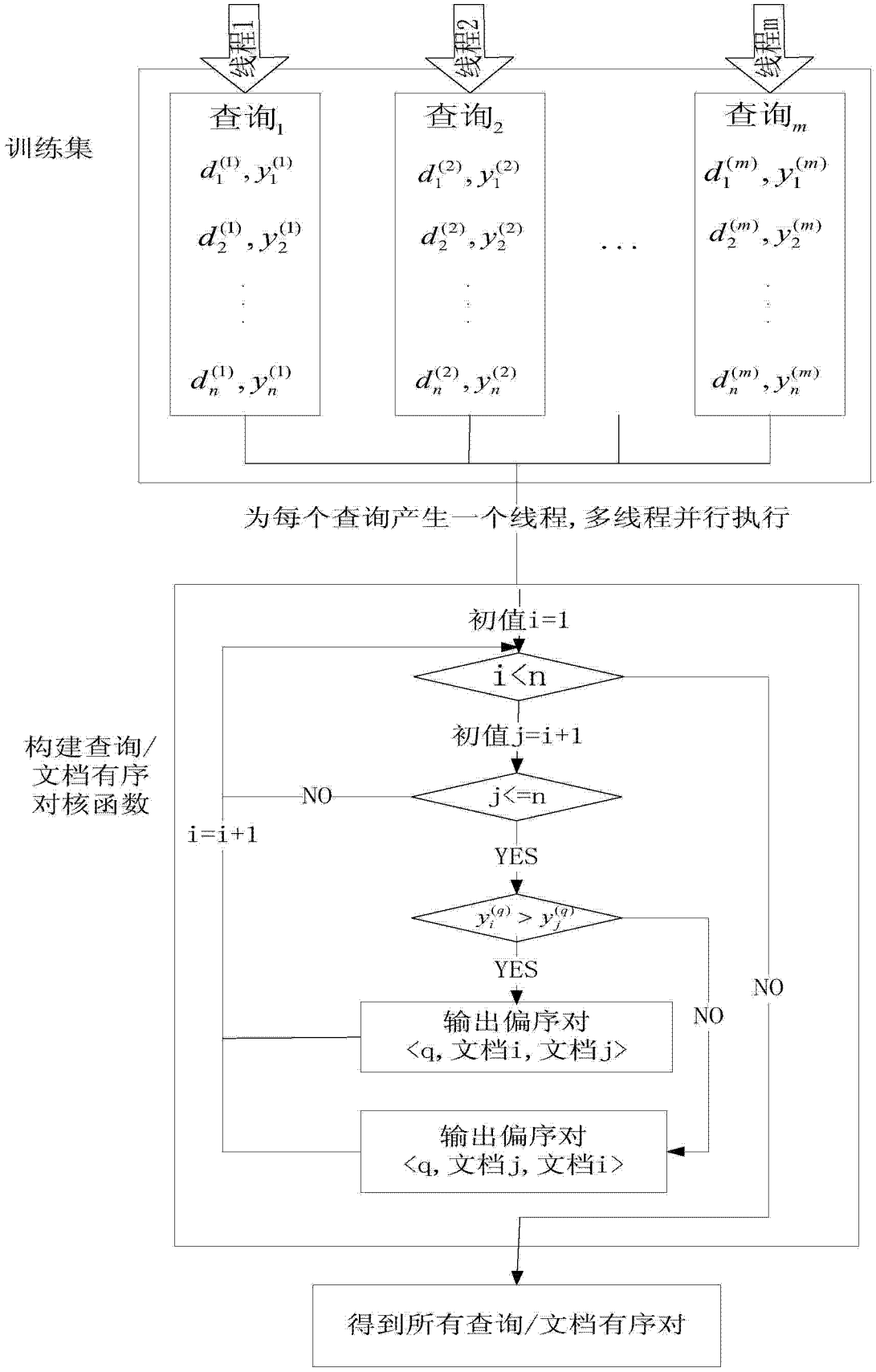

训练集
线程1
线程2
线程m
查询1
查询2
查询m
为每个查询产生一个线程，多线程并行执行
构建查询/文档有序对核函数
初值i=1
i<n
初值j=i+1
NO
j<=n
i=i+1
YES
YES
NO
NO
输出偏序对
<q, 文档i, 文档j>
输出偏序对
<q, 文档j, 文档i>
得到所有查询/文档有序对

图 7

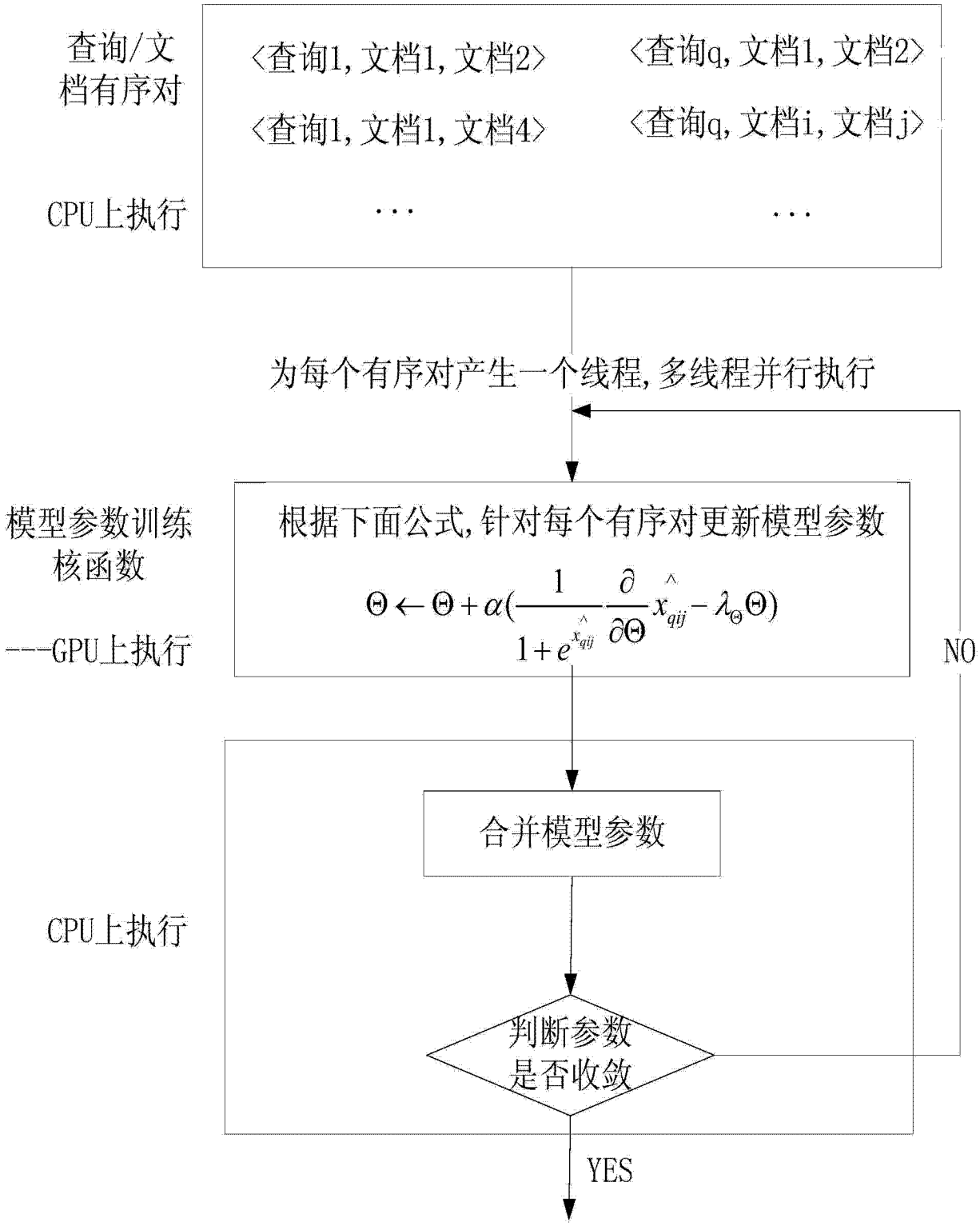

查询/文档有序对
〈查询1, 文档1, 文档2〉
〈查询q, 文档1, 文档2〉
〈查询1, 文档1, 文档4〉
〈查询q, 文档i, 文档j〉
CPU上执行
为每个有序对产生一个线程, 多线程并行执行
模型参数训练
核函数
---GPU上执行
根据下面公式, 针对每个有序对更新模型参数
$\Theta \leftarrow \Theta + \alpha(\frac{1}{1+e^{\hat{x}_{qij}}}\frac{\partial}{\partial\Theta}\hat{x}_{qij} - \lambda_{\Theta}\Theta)$
NO
合并模型参数
CPU上执行
判断参数是否收敛
YES

图 8

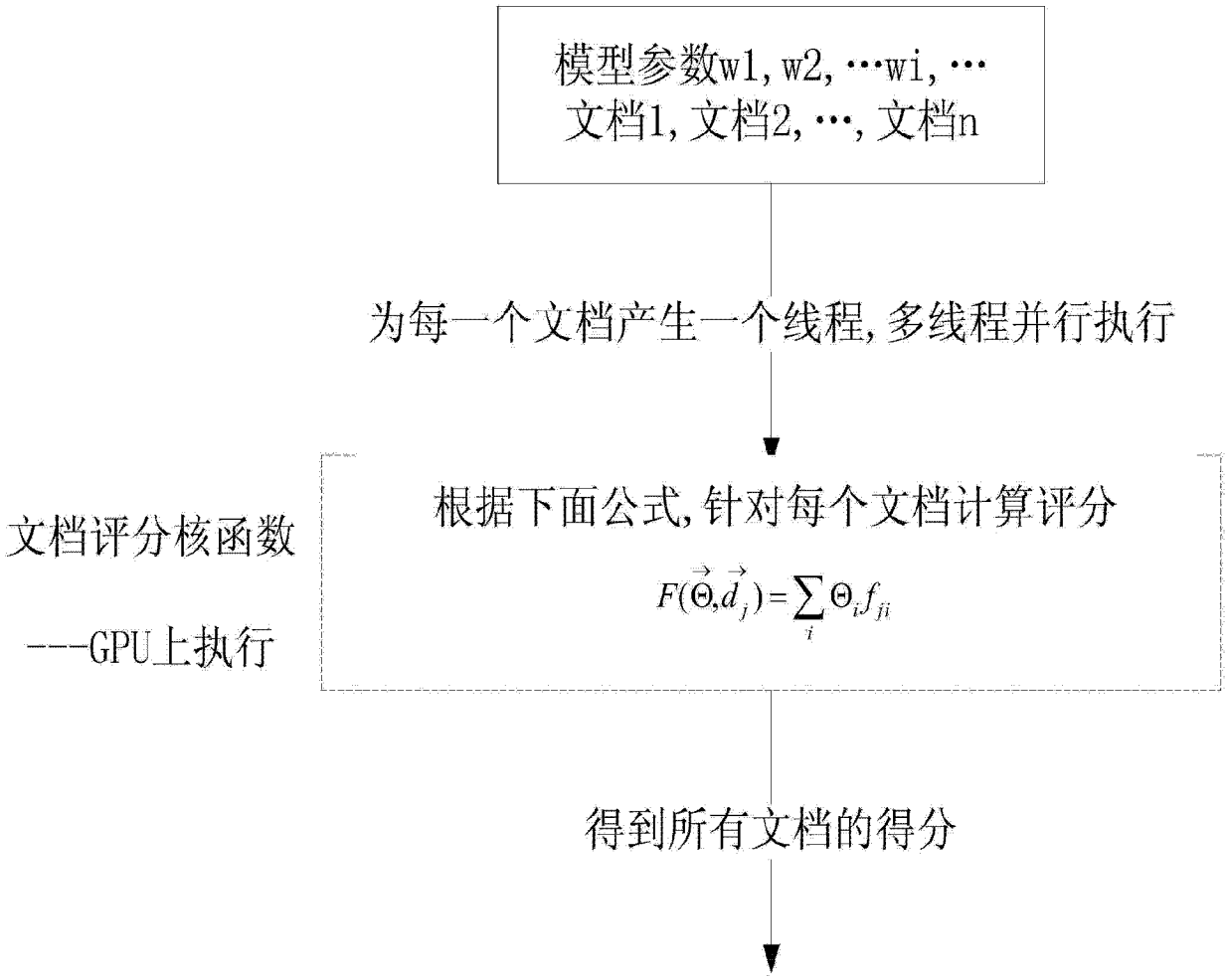

模型参数w1, w2, …wi, …
文档1, 文档2, …, 文档n
为每一个文档产生一个线程, 多线程并行执行
文档评分核函数
---GPU上执行
根据下面公式, 针对每个文档计算评分
$F(\vec{\Theta},\vec{d}_j)=\sum_i \Theta_i f_{ji}$
得到所有文档的得分

图 9

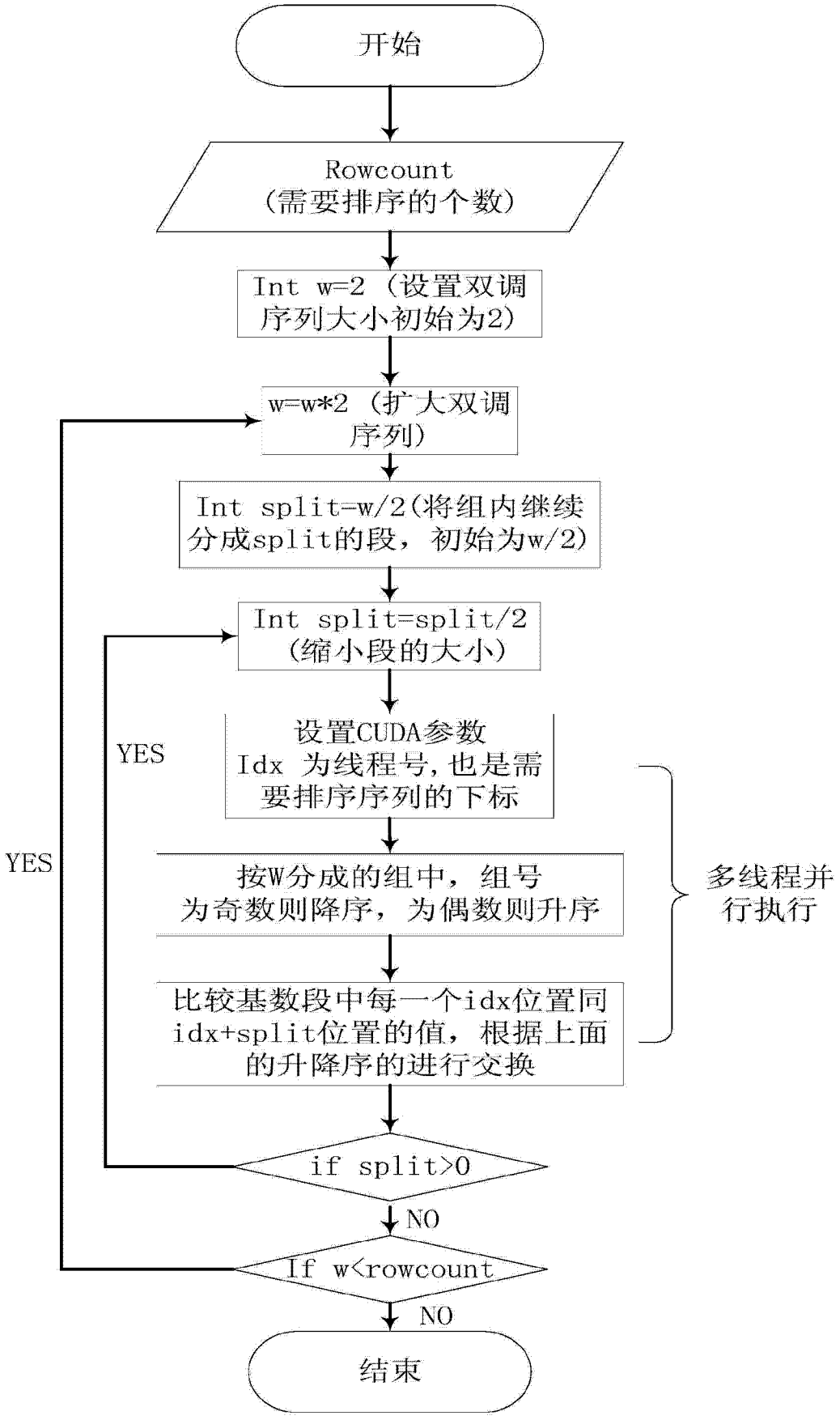

开始
Rowcount
（需要排序的个数）
Int w=2 （设置双调序列大小初始为2）
w=w*2 （扩大双调序列）
Int split=w/2(将组内继续分成split的段，初始为w/2)
Int split=split/2
(缩小段的大小)
设置CUDA参数
Idx 为线程号,也是需要排序序列的下标
按W分成的组中，组号为奇数则降序，为偶数则升序
比较基数段中每一个idx位置同idx+split位置的值，根据上面的升降序的进行交换
多线程并行执行
if split>0
YES
NO
If w<rowcount
YES
NO
结束

图 10