4U-02 **協調サーチエンジンにおけるインデックス更新時間の短縮化**

山本 崇† 佐藤 永欣† 西田 喜裕† 上原 稔† 森 秀樹†

†東洋大学工学部情報工学科

1 はじめに

我々は我々が開発している協調サーチエンジン (Cooperative Search Engine、以下 CSE) においてインデックス更新時間の短縮化により、更新間隔を短縮した運用が可能なサーチエンジンを目指している。CSE とは複数の局所サーチエンジンを構築し、これらを協調動作させ検索作業や、インデックスファイルの更新作業を行なうものである。

本稿では、インデックスファイルの更新作業時に、処理に時間のかかる局所サーチエンジンが作業の一部を他の局所サーチエンジンに依頼することにより、CSE のインデックス更新作業の所要時間を短縮させるシステムについて述べる。

2 関連研究

分散型検索エンジンに関する研究は、各所で行なわれている。検索対象となる文書の収集部分の分散化は JEIDA[1] 等で行われている。収集部分の分散化は、多数の文書を取得するためにかかる時間の短縮を主な目的にしている。google[2][3] では、URL のリストを送信する URL Server、取得した情報を保存する Store Server、HTML 文書を取得する crawler を用いて文書の収集を行なっている。この crawler を分散させることで、分散収集を行なっている。また、収集した文書から、Indexer、URL Resolver、sorter の各構成要素が協調動作し、検索用のデータベースの作成を行っている。

3 CSE のインデックス作成時の動作

3.1 CSE の構成

CSE には、検索動作や、インデックスファイルの更新作業を行う Local Meta Search Engine(LMSE)、CSE 内の各 LMSE に関する情報を管理する Location Server(LS)、検索時に LMSE からの要求を受け、LS や LMSE との通信を行ない検索結果を取得、キャッシュする、Cache Server(CS) の三つの構成要素がある。

Reduce of processing time of information update in Cooperative Search Engine
Takashi Yamamoto
Department of Information and Computer Sciences, Faculty of Engineering,Toyo University

3.2 インデックス更新の過程

CSE におけるインデックスの更新作業の概略図を図 1 に示す。

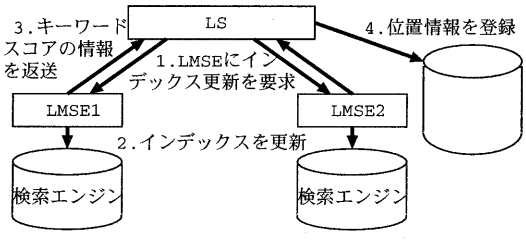

図 1: CSE の検索時の動作

更新作業は LS が主体となり処理を行なう。これは CSE 全体を制御するのに都合がよいからである。更新作業は LS と LMSE が処理を行なう。更新作業は以下のようにして行なわれる。

1. LS は LMSE にインデックスの更新とインデックスに関する情報の更新を要求する。
2. 要求を受けた LMSE はインデックス更新作業を行ない、キーワードとスコアの情報を抽出する。
3. LMSE はインデックスの更新が終ったら、抽出した情報を LS へ送信する。
4. LS は送信されてきた情報を元に、位置情報を更新する。

LMSE が構築されている計算機の性能や、検索の対象とする文書の量の違いがあるため各 LMSE の更新作業の所要時間には、ばらつきがある。CSE 全体としてのインデックスの更新時間は、最も処理に時間のかかった LMSE のインデックス更新の所要時間に依存している。そこで、インデックス更新の処理が終了していない LMSE の中で、最も処理に時間がかかるであろうと思われる LMSE が、処理の終了した LMSE へ作業の一部の処理を依頼することにより、各 LMSE の更新所要時間の差を縮め、CSE 全体の更新所要時間の短縮を目指すシステムを開発した。

先に挙げた各 LMSE が行うインデックス更新作業を (a) インデックス更新作業の対象となる文書を収集する、文書収集作業と、(b) 検索対象とする文書に対し処理を行ない、インデックスファイルへ情報を書き

出すインデックス作成作業、の二つに分類する。他のLMSEへ処理の依頼をするのはインデックスファイルの作成の処理である。

4 LMSE間の処理の依頼

LMSEの処理の依頼は最も処理に時間がかかると思われるLMSEが他のLMSEに対して行う。各LMSEの更新所要時間の推定は、更新作業の対象とする文書のファイルサイズと、以前の更新作業時のファイルサイズと所要時間に基づいて行う。更新作業終了時に、その所要時間($time$)と処理を行なった総ファイルサイズ($filesize$)から$P = \frac{time}{filesize}$で求められる$P$を記録しておき、これから処理を行なう総ファイルサイズ($filesize$)から、推定所要時間(T)を$T = P \times filesize$で求める。処理の対象となる文書のファイルサイズと、処理の所要時間との関係は、あまり密接なものではないが、インデックス作成作業の前や、その途中で容易に測定でき、測定の所要時間も短いという利点を考え、今回は、このような方法を用いた。この所要時間の推定の方法は、今後考慮していかなければならないだろう。

LMSEから他のLMSEへのインデックス更新処理の依頼は以下のように行われる。

- LSは各LMSEのPの値を知っているものとする。また、LSは各LMSEがインデックス更新作業を行っているか行っていないかを把握しているものとする。
- インデックス更新作業を行うLMSEは、更新作業の対象となる文書の総ファイルサイズをLSに通知する。
- LSは、最も処理に時間がかかるであろうLMSEへ、処理の依頼を行なうよう通知する。依頼先のLMSEは、インデックス更新作業を行っていない(既に終了した)LMSEである。
- 通知を受けたLMSEは通知に従い、他のLMSEへ処理の依頼を行う。その結果、自分自身が処理を行なう総ファイルサイズが変化するので、再計算し、LSへ通知する。
- 処理が終了したLMSEは、所要時間と処理を行った総ファイルサイズをLSへ通知する。LSは、そのLMSEに関するPの値を更新する。

5 評価

今回作成したシステムを用いて実験を行なった結果を示す。実験にはlute,mutsuki,heliosの計算機上にLMSEを構築し、測定を行なった。今回は、各LMSEで同じ文書を対象に処理を行なわせ、その所要時間を測定した。ファイル数1400個のHTML文書から作成されたインデックスファイルに、ファイル数200個、総ファイルサイズ1055[kbyte]のHTML文書の情報の追加、更新作業をそれぞれの計算機で行なった。その結果を表.1に示す。

表 1: 処理依頼を行なわない場合の所要時間

	lute	mutsuki	helios
所要時間 [min:sec]	2:20	3:20	6:29

表 2: 処理依頼を行なった場合の所要時間

	lute	mutsuki	helios
所要時間 [min:sec]	2:33	3:22	3:59

次に、同様の条件で、インデックス作成を行ない、処理依頼を行なうようにした場合の結果を表.2に示す。この所要時間は、各LMSEが担当する文書の処理の所要時間である。lute,mutsuki上のLMSEは、自分が担当する文書の処理の終了後はheliosからの処理依頼を受け作業を行なった。その結果、heliosの更新作業の所要時間が短縮された。

6 まとめと今後の課題

本稿では、CSEの中で処理に時間がかかるLMSEが処理の一部を他のLMSEに依頼することにより、そのLMSEの処理の所要時間を短縮させ、CSE全体のインデックス更新所要時間の短縮を可能にするシステムについて述べた。今後の課題を挙げると、LMSEの処理時間の推定方法の改善が挙げられる。また、今回は自分自身が担当する処理を終えたLMSEのみが、他のLMSEの処理を引き受けたが、自分自身の処理と並列に、他のLMSEの処理を引き受ければ、CSE全体の更新時間の短縮につながると思われる。各LMSEの処理の分担方法について考える必要があるだろう。

参考文献

[1] 『次世代分散型情報検索システムに関する調査研究報告書』 http://www.jeida.or.jp/committee/jisedai/top.html (1997)

[2] *Google* http://www.google.com

[3] Sergey Brin,Lawrence Page 『*The Anatomy of a Large-Scale Hypertextual Web Search Engine*』, Seventh International World Wide Web Conference http://www7.scu.edu.au/programme/fullpapers/1921/com1921.htm

[4] 高林哲 『全文検索システム *Namazu*』 http://openlab.ring.gr.jp/Nnamazu/