

3S-8

モバイルエージェントを用いた 検索システムの連携における一検討

國頭 吾郎 相澤 清晴

東京大学大学院 新領域創成科学研究科

東京大学大学院 工学系研究科

1 はじめに

近年ディスク容量の増大に伴い、メールやニュースといったプライベートな情報の量も増大している。例えば自分が過去に受信したメールの中からあるメールを探し出したいようなときなど、一つ一つ目でみながら探すことは非常に困難になりつつある。このような場合に全文検索システムがあると非常に便利である [2]。

筆者らはこのようなプライベートユースの小規模検索システムを単にローカルとして使うだけでなく、グローバルなシステムの窓口として使うための小規模検索システムの連携を提案してきている [1]。本稿では連携検索の際の再帰検索の継続決定法について検討する。

2 再帰検索の問題

複数の検索サーバが連携して検索を行うと、再帰的な検索が発生する。この再帰的検索においては次のような問題点が考えられる。

- 再帰的検索の深さ
- 検索遅延

検索サーバ間が連携する場合には、一つの検索サーバに検索をかけた場合に他のサーバへの検索が実行される可能性があり、さらにそのほかのサーバへの検索が実行される可能性がある。このような再帰的検索の場合に、どの深さまで実行するか、どの深さでうち切るかが問題である。さらに、検索の深さにも関連するが、再帰的に検索を行うことによって遅延が大きくなってしまいう可能性がある。ユーザによっては遅くてもいいからなるべく多くの検索結果がほしい場合や、多少少なくてもいいか

ら素早い検索結果がほしい場合等が考えられる。これらの問題の対策の一つとして、ユーザから階層またはタイムアウトを指示するという方法が考えられる。それぞれの利点・欠点は以下の通りである。

1. タイムアウト指定

指定されたタイムアウトにしたがって検索を打ち切るとすると、ユーザへのレスポンスの向上を期待できる。

2. 階層の深さの指定

ユーザによって階層の深さを指定することで、検索による負荷のかかるホストを限定することができ、また再帰的検索のトラフィックの発生する範囲を限定することができる。しかしながら階層の深さを指定しても必ずしもユーザへのレスポンスに反映されるとは限らない。

以上の理由から、再帰的検索においてユーザまたはユーザエージェントが検索の条件を指定する場合には、階層の深さ、及びタイムアウトの両方に関する指定をする必要があるといえる。

3 再帰的検索に要する時間

図1のように、ユーザはユーザエージェントを通して検索サーバAに検索要求を出し、検索サーバAは検索サーバB,Cに再帰的に検索を行うとする。このとき、2パスのサーバ間連携プロトコル [1] を用いるものとする。

ここで、User Agent、検索サーバA,B,Cをそれぞれ U, A, B, C と表わす。図2のように T_{trans} はデータを転送にかかる時間で、 T_{trans}^{AC} はサーバAからCへの転送時間である。検索、スコアの再計算、結果のマージにかかる時間をそれぞれ T_{search} , T_{recalc} , T_{merge} と表わす。また、サーバAが連携プロトコルパス2のためのキーワード出現頻度の総和、全文書数の総和を計算する時間を T_{calcN}^A 、ユーザエージェントに結果を返すまでの時間

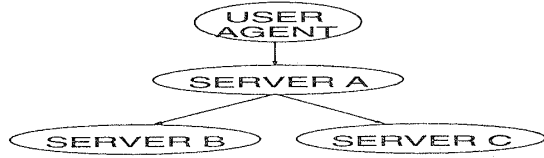


図 1: A Hierarchy Search System

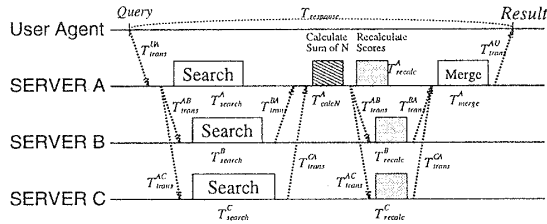


図 2: 再帰的検索の流れ

を $T^D_{response}$ (再帰的検索をしない場合), $T^H_{response}$ (再帰的検索する場合) とする。

もしサーバ A がサーバ B, C に再帰的検索を行わない場合は

$$T^D_{response} = T^{UA}_{trans} + T^A_{search} + T^{AU}_{trans} \quad (1)$$

である。サーバ B, C に再帰的検索を行うとすると

$$\begin{aligned} T^H_{response} = & T^{UA}_{trans} \\ & + \max [T^A_{search}, (T^{AB}_{trans} + T^B_{search} + T^{BA}_{trans}), \\ & (T^{AB}_{trans} + T^B_{search} + T^{BA}_{trans})] \\ & + T^A_{calcN} \\ & + \max [T^A_{recalc}, (T^{AB}_{trans} + T^B_{recalc} + T^{BA}_{trans}), \\ & (T^{AB}_{trans} + T^B_{recalc} + T^{BA}_{trans})] \\ & + T^A_{merge} + T^{AU}_{trans} \end{aligned} \quad (2)$$

となる。ここで、 $T^A_{search} = T^B_{search} = T^C_{search} = T_{search}$, $T^A_{recalc} = T^B_{recalc} = T^C_{recalc} = T_{recalc}$, $T^{AB}_{trans} = T^{BA}_{trans}$, $T^{AC}_{trans} = T^{CA}_{trans}$, $T^{AU}_{trans} = T^{UA}_{trans}$ と仮定すると、

$$\begin{aligned} T^H_{response} = & 2T^{UA}_{trans} + T_{search} + 4 \max [T^{AB}_{trans}, T^{AC}_{trans}] \\ & + T^A_{calcN} + T_{recalc} + T^A_{merge} \end{aligned} \quad (3)$$

となる。1 階層再帰的検索を行うごとににかかる時間は

$$\Delta_{response} = T^H_{response} - T^D_{response} \quad (4)$$

のようになる。

4 再帰的検索打ち切り決定法

図 1 において、検索サーバ A が検索要求を受けたとき、再帰的検索を行わない場合には結果を返すまでにかかる時間は 1 式で与えられる。また、1 階層再帰検索によって増加する処理時間は 4 式である。ここで Aglets を用いて実装を行うと、メッセージのやり取りのイベントや外部プロセスの起動に時間がかかるので、 T^A_{calcN} , T_{recalc} は内部計算処理だけであることから

$$T^{AB}_{trans}, T^{AC}_{trans} \gg T^A_{calcN}, T_{recalc}, T_{merge} \quad (5)$$

であり、4 式は

$$\Delta_{response} \simeq 4 \max [T^{AB}_{trans}, T^{AC}_{trans}] \quad (6)$$

となる。ここで、6 式の $\Delta_{response}$ はあらかじめ測定しておくか、過去の値を利用することが可能である。

検索サーバ A が検索サーバ B, C に再帰的検索を実行するのは次の条件がそろった場合である。

- ユーザから指定された再帰階層の深さ以下
- $T^H_{response}$ が最大待ち時間以内

過去の検索での T^{UA}_{trans} , T^A_{search} , T^{AU}_{trans} を元に $T^H_{response}$ を計算し、この値が最大待ち時間以内であればサーバ B, C へ検索要求をする。

以上のようにして再帰検索を行う。

5 おわりに

パーソナルな情報は今後とも増大し、検索システムが求められるような状況が今後ますます増えるであろう。

筆者らは個人が持っているような小さなデータベースを、個人的に使うだけでなく他の小さなデータベースと連携させていこうという方向を目指している。本稿ではユーザへのレスポンスを所定の時間内に制限するために、再帰検索の際の所要時間の見積もりに関する検討を行った。この検討をもとに実装を進めていく予定である。

参考文献

- [1] 國頭吾郎, 相澤清晴, 羽鳥光俊. モバイルエージェントを用いた WWW 検索システムに関する一検討. 信学技報 MVE98-93, 電子情報通信学会, Feb 1999.
- [2] 馬場肇. 日本語全文検索システムの構築と活用. ソフトバンク, Sep. 1998.