Verbose-Agentによる情報検索の考察

5S－5

山本英雄　梅村恭司
豊橋技術科学大学

1 はじめに

近年、新聞記事などの電子化などにより、大量のデータを対象とした検索が行なわれるようになった。新聞記事検索などでは、欲しい情報を手に入れるためには膨大なデータを検索する必要があり、複数の記事が候補として得られる。我々はそのような検索を、検索範囲が無限大、検索結果も複数、途中結果も有効であるような問題と考えており、検索には、途中結果の報告と検索の打ちきりが有効であると考えている。人が大量の情報に対して手分けして検索を行なう場合、リーダー的存在の者が仲間に検索を指示し、その検索が終了するまでに数日かかるような場合、少なくともその日のうちにはリーダーに何らかの報告をするであろうし、リーダーも何らかの情報を欲するはずである。すべての検索が終了するまで結果が分からないというのは、多くの時間を必要とし、無駄な労力と言わざるを得ない。そして、報告を聞いたリーダーは検索の中止を命じたり、新たな検索を命じたりして欲しい情報を得るであろう。これは人間ならば実際に行なっている検索法だが、コンピュータシステムで行なわれることは少ない方法である。

新聞記事検索は、そのリソースの大きさからも候補の記事が複数存在することからも上記の検索問題に合致する。そこで我々は、オンラインの新聞記事検索システムを構築するために、途中結果の報告の機能を備えた分散システムフレームワークを作成し、それを用いて新聞記事検索システムを構築した。

2 分散システムフレームワーク

2.1 フレームワーク構成

本フレームワークは、ワーカー(Worker), エージェント(Agnet), ジョブ(Job), 報告条件(ReportCondition)の4つを中心として構成されている。ワーカーはエージェントに実行の環境を提供する場であり、エージェントは実行させたい処理を格納する器である。そしてジョブはワーカーで実行させたい処理であり、報告条件は、エージェントがユーザに報告を行なう条件である。例えば、ホストAに移動して検索を行なうと共に、検索実行中には10秒おきに閾値以上のスコアの解の候補のみを報告させたい場合には、我々のフレームワークでは、ホストAに移動するジョブと検索を行なうジョブ、そして10秒おきに閾値以上のスコアのみを報告するという報告条件を用意し、それぞれをエージェントにセットすればよい。以下は上記の処理を行なうサンプルコードである。

```
Worker_Proxy homeworker
        = new Worker_Proxy("horb://userhost");
ReportConditon cond = new MyReportCondition();
Agent agent = new Agent(homeworker, cond);
agent.add(new Move());
agent.add(new Search());
homeworker.startJob(agent);
```

図 1: サンプルコード

3 性能評価のためのモデル化とそのシミュレーション

途中結果の報告と検索の打ち切りを導入した検索の性能評価のために、条件を満たす結果が得られるまで報告をしない単報告検索と、途中結果を報告する複報告検索による比較を行なった。図2は検索過程における次の解を発見するまでの間隔のモデルである。まず、報告の処理のために生じる遅延をパラメータとして、単報告検索によって結果が報告される以前に、複報告検索で何らかの報告がなされていたら複報告フレームワークを勝ちと

The information retrieval using Verbose-Agents
Hideo Yamamoto, Kyoji Umemura
Toyohashi University of Technology

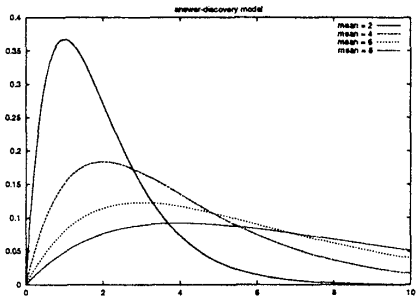

図 2: 解発見モデル

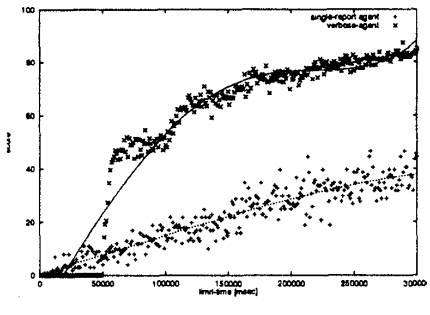

図 3: 打ち切り時間内に得られるスコア平均

したときの複報告検索の勝敗をシミュレートした。その結果、表1に示されるような、勝敗が5分5分となる遅延時間の大きさが得られた。

表 1: 勝ち負けが5分5分となる遅延時間

解発見の平均間隔	相対遅延時間
2	5.681
4	5.551
6	5.612
8	6.033

次に、打ち切り時間をパラメータとして、打ち切り時間内に得られるスコアの期待値をシミュレートした。報告によって生じる遅延は最初のシミュレーションにおいて得られた勝敗が5分5分となる遅延時間を使用している。得られる解のスコアは一様分布に従っており、単報告検索では、99以上のスコアの解を得たときに報告するとする。図3に、解発見モデルの平均8秒、遅延時間を48秒としたときのシミュレーション結果を示す。

グラフからもわかるとおり、単報告検索で条件を満たす結果を得るには多くの時間を要することが予測できるが、複報告検索の場合、打ち切り時間内でも、途中結果を報告することによって単報告検索の場合よりスコアの期待値が高くなっていることがわかる。

4 応用システム

本フレームワークを利用したシステムとして、新聞記事検索システムを構築した。このシステムでは、記事検索プログラムとして、欲しい記事に関するあいまいな入力からでも、入力に類似した内容の記事を検索できるプログラムを利用している。この検索プログラムは、あいまいな入力からでも結果を得られるという特徴をもつ反面、計算には多くの時間を要し、唯一の解というものは存在しない。しかしながら、新聞記事データを複数のホストに分散させることで比較的簡単に検索時間を減らすことができる。

現在、PentiumII300MHz クラスのマシン4台による分散システムでは、毎日新聞2年分の記事約200MBのリソースを検索するのに要する時間は8分程度で、かつ候補の記事が見つかり次第報告するようになっているので、すべてのリソースを検索し終るまでユーザは待つ必要がなく、逐次、候補記事を参照することができるシステムが構築できている。

5 まとめ

本稿では、途中結果報告と検索の打ち切りという、実際に人間が行なっている検索法をコンピュータシステムで行なう方法の提案と性能評価、そしてそれを実現するための分散フレームワークの構築について述べた。また、応用システムとして新聞記事検索システムを示した。

参考文献

[1] Yasuyuki Tahara, " Plangent – An Intelligent Multiagent System for Network Computing " ,ICMAS–96,p460,1996

[2] 津田 宏, 山崎重一朗, " Telescript 言語入門 " ,ASCII Books,Inc.,1996