2Q-1

スーパーノードと木構造を用いた自律分散協調ネットワーク上の全文検索システム

成毛源樹*　　山口和紀†

1 導入

本研究では、P2Pネットワーク中の各ノードが持つ全ての文書に対して、効率的かつ柔軟な全文検索を行えるようなシステムを実現する手法を提案する。

P2Pにおける探索手法で最も単純なものはGnutella[1]などで使われているFloodingという手法である。Floodingには、各ノードがネットワークの局地的な情報(自分の周囲のノードのIP)を知っているだけで探索が可能である、という利点があるが、いつまでも目的のもの(ノード、ファイル等)が見つからない場合、通信量が爆発的に増加してしまうという問題がある[2]。この問題により、Floodingだけを使ってネットワーク全体の文書に対して、効率的に欲しい文書を得られるような全文検索システムを実現することは難しいと思われる。

近年注目されている探索手法として分散ハッシュテーブル(以下DHT)が挙げられる。DHTでは一意のハッシュ関数を用いてノードやファイルをネットワーク上にマッピングし、探索時には同じハッシュ関数を用いることで探索を効率的に行う。DHTを使うことでスケーラビリティや通信量の問題は解決できるが、ハッシュ値算出時に使用するキー(ノードのIP、ファイルの名前等)を完全に知っていないと目的のもの(ノード、ファイル等)を発見できないという欠点がある。DHTを使った全文検索システムを考えた場合、最も単純なものは、通常の全文検索システムで使われているような転置インデックスの単語毎の情報(出現文書、出現位置)を、DHT上のノードに登録するものであるが、この場合もDHTの仕様により、登録された単語を完全に知っていなければその単語に関する情報を登録したノードを発見できない。これはつまり、部分検索等ができないということである。n-gram単位の転置インデックスを使うことで部分検索等を実現することが出来るが、インデックスサイズが増大してしまい、登録時の通信量、各ノードが保持するデータ量も増大してしまう。本研究ではこのような問題を解決するため、DHT上に完全な出現位置情報を登録した転置インデックスを分散させるのではなく、tf-idfスコアのみを登録した粗いインデックスを分散させ、このインデックスを利用して検索語のスコアが高い文書を持つノードにだけクエリをFloodingする、という手法を提案する。

---

* Motoki Naruke. Graduate School of Arts and sciences, The University of Tokyo. motoki@graco.c.u-tokyo.ac.jp

† Kazunori Yamaguchi. Information Technology Center, The University of Tokyo.

2 システム概要

2.1 登録と更新

本手法では2gram単位の転置インデックスを使う。DHTとしてChord[3]を使い、hash(IP)で決定されるIDがhash(2gram)に最も近いノードに、全ノードに関する2gramのtf-idfスコアを登録する。2gramのスコアをDHT上の対応ノードに登録する時は、2gram毎にバラバラに登録していくのではなく、各ノードが持つDHTの経路表を使ってアンバランス木を構築、利用してまとめて登録する。インデックス更新時は、各ノードがスコアを再登録するのではなく、ネットワーク中にスーパーノードと呼ばれる、サーバのように働くノードを動的に決定し、定期的に情報を集約することで行う(図1.)。

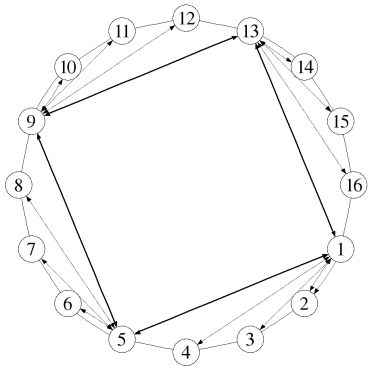

図1. スーパーノードが定期的に情報を集約

2.2 検索

検索は以下の流れで行う

- 検索語を2gramに分解
- 分解した全2gramに関して、それぞれの情報を保持するDHT上のノード(ID≒hash(2gram))に問い合わせ、高スコアのノードを選出
- 2gramの平均スコアが高いノードを選出、スコアをキーとした木を構築、クエリをFlooding

3 実験結果

本研究の手法で問題となるのは、ある単語を構成する2gramのスコアが高い文書が、本当に単語のスコアの高い文書であるか、ということである。

以下では http://www.gutenberg.org/ より取得した文書(総文書数:：10598、総容量：4136MB)について、文書中の単語のスコアと単語を構成する2gramのスコアの相関と、2gramのスコアを使って単語のスコアが高い文書を選出できるかどうかについて実験を行った。

3.1 単語のスコアと2gramのスコアの相関

全文書に関して、21217単語のスコアと各単語を構成する2gramの平均スコアの相関係数を算出した。単語を構成する2gram全てを含んでいても、実際には単語を含まない文書は除き、純粋な相関係数を算出した。

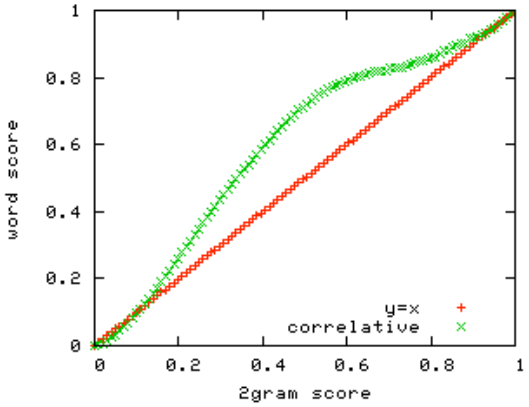

図 2. 単語のスコアと2gramの平均スコアの相関の平均

図 2. は全単語の平均スコアと各単語を構成する全2gramの平均スコアを0-1で正規化したものをプロットした図で、横軸は単語を構成する2gramの平均スコア、縦軸は単語のスコアである。相関係数の平均は0.983となった。単語の中には2gramのスコアとの相関係数が-1となるものがいくつかあったが、これは非常にレアな単語(含む文書が極端に少ない)で、その次に相関係数が低い単語で0.3以上、以降は相関係数0.5以上となり、総じて高い相関を示すと言える。

以上の実験により、単語のスコアと単語を構成する2gramの平均スコアには高い相関があることがわかる。

3.2 2gramのスコアによる文書の選出

21217単語に関して、単語のスコアが上位5%以内であるような文書のうち80%を選出することを考えたとき、単語を構成する2gram全てを含む文書から、スコアが上位何%のものまでを選出すればよいかを調べた。同時に、選出された文書が実際に単語を含む確率も算出した。

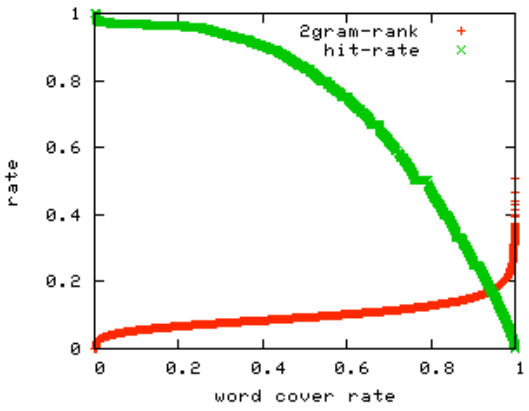

図 3. 単語スコア上位5%を80%取得

図 3. 上昇線：縦軸は、単語を含む主な文書(単語のスコアが上位5%以内であるような文書のうち80%)を選出するために、単語を構成する2gram全てを含む文書から、スコアが上位何%のものまでを選出すればよいかを示す。横軸は単語のカバー率である。例えば2gramの平均スコアが上位10%となるような文書を選出することで、80%程度の単語で主な文書を選出することができる、ということがわかる。

図 3. 下降線：縦軸は2gramの平均スコアが上昇線の基準を満たすような文書を選出したときに、それらの文書が実際に単語を含んでいる確率である。横軸は単語のカバー率である。例えば上昇線に従って、単語を含む主なファイルを選出できるだけの、2gramの平均スコア上位の文書を選出すると、全単語の50%以上でそれら文書の80%以上が実際に単語を含んでいることがわかる。

以上の実験により、2gramの平均スコアが高い文書を選出することで、単語のスコアが高い文書を選出できることがわかった。これにより、2gramのtf-idfスコアを登録した転置インデックスを使って、単語のスコアが高い文書、さらにはクエリを送出するノードを選出することが可能になる、ということがわかる。

4 まとめ

本論文では単語のスコアと2gramのスコアの相関性を示し、2gramのスコアを使うことで単語のスコアが高い文書を選出できることを示した。これにより転置インデックスにtf-idfスコアのみを登録することでデータ量を削減し、この転置インデックスを利用して検索語のスコアが高い文書を持つノードにだけクエリを送出することで検索の効率化を図るという、本研究の手法が有効に働くものと思われる。

今後は分散環境下の様々な状況(高トラフィック時等)においても、実際にインデックス登録、更新、検索が可能であることを検証する。インデックス更新に関しては、スーパーノードを動的に、適切に決定することが検索精度やパフォーマンスに大きく影響することになるので、特に仔細に検証したい。

参考文献

[1] Gnutella http://www.gnutella.com/

[2] Jordan Ritter, “Why Gnutella Can't Scale. No, Really,” http://www.darkridge.com/~jpr5/doc/gnutella.html

[3] Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek, and Hari Balakrishnan, “Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications,”ACM SIGCOMM 2001