B-14

オープンソース開発支援用メール検索システムの試作
Mailing-lists Archive Search System for Open-Source Software Development

高尾祐治† 石川武志‡ 松下誠† 井上克郎†
Yuji Takao† Takeshi Ishikawa‡ Makoto Matsushita† Katsuro Inoue†

1 まえがき

世界中に分散した多数の開発者が並行して開発を行う，オープンソース開発(以降 OSD と記す)と呼ばれる開発手法がある[1]．OSD では，意志疎通の手段としてメーリングリスト(以降 ML と記す)が用いられる．ML でやりとりされた全てのメールはアーカイブとしてまとめられ，問題点とその解決方法といった，開発を行う上で有益な情報が多数蓄積される．このため，OSD に携わる開発者は，疑問点，問題点が生じるたびにアーカイブを検索して過去の議論を参照することになる．しかし，既存の検索システムは全文検索という単純な方法を使っているため，問題解決に役立つ情報を検索できない，あるいは，必要のない余分な情報を検索することがある．

そこで本研究では，メールを使った議論の単位であるスレッドに注目し，スレッド間の結び付きを使ったオープンソース開発支援用メール検索システムの設計と実装を行う．また，実際の OSD で用いられている ML を使って評価を行い，その有効性について考察する．

2 スレッド間の結び付きを使った検索

アーカイブとは，ML でやりとりされた全てのメールが含まれるメールの集合である．アーカイブの検索に多く用いられる全文検索は，多数のメールの中からある文字列が含まれるメールを検索するという検索手法である．そのため，あるメールに開発者の抱える問題を解決させることのできる情報が含まれていても，検索に使った文字列がメール中に含まれない限りそのメールを検索することはできない．本節では，全文検索には検索できない情報を検索するため，スレッド間の結び付きを使ったメール検索システムを設計する．

2.1 スレッドとその結び付き

ML での議論は，一般的に，ある開発者が送信したメールに他の開発者が返信し，意見を述べ合うことで行われる．そのため，議論がなされた 2 通のメールに着目すると，一方のメールが他方のメールに返信されたという返信関係が存在する．本研究では，議論の発端となるメールと返信関係にあるメールを再帰的に求めて得られるメールの集合をスレッドと定義する．スレッドには，問題提起から，様々なコメントが寄せられ解決に至るまでの議論が含まれるため，スレッドは議論の単位であると考えることができる．

スレッドでは特定のトピックについて議論が行われる．ところが，他のスレッドでも同じトピックについての議論が行われる場合がある．このように，アーカイブには互いに関連のあるスレッドが存在する．本研究ではスレッド間の関連を，スレッド間の結び付きとして 3 種類定義する．

A) 似たトピックについて議論している
B) 同じファイルについて議論している
C) 同じ開発者たちが議論している

これらの結び付きにあるスレッドを検索することで，関連のある議論を検索することができる．

アーカイブ中のスレッドとそれらの間に存在する結び付きを図 1 に示す．

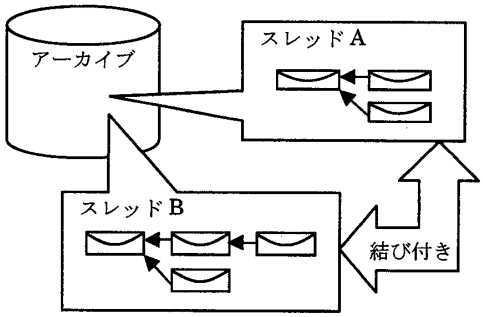

図 1 スレッドとその結び付き

2.2 結び付きにあるスレッドの検索

あるスレッドが与えられたとき，そのスレッドと結び付きにあるスレッドを計算するために，3 種類の文字列を考える．これらの文字列は，3 種類の結び付きに対応する．

A) スレッドから抽出したキーワード
B) メールに書かれたファイルパス
C) 議論している複数の開発者のメールアドレス

それぞれの文字列を使ってアーカイブの全文検索を行い，検索結果のメール群からスレッドを求める．これらのスレッドが，結び付きにあるスレッドである．以上の手順で，あるスレッドが与えられたとき，そのスレッドと結び付きにあるスレッドを計算することができる．

なお，スレッドからキーワードを抽出するための計算手順は以下の通りである．

1. メール中の文章を単語単位に分割する
2. 単語の出現回数と出現した箇所から重みを計算する
3. 前置詞，助詞等のキーワードとして不適切な単語を除外する
4. 重みの大きい単語をキーワードとする

2 において，重みを計算する際に次の 2 点を考慮する．

- Subject はメールの内容を代表する文章であるため，本文中に出現した場合より重みを大きく計算する

† 大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology, Osaka University
‡ 富士通株式会社
FUJITSU LIMITED

- コンピュータ用語は単語の頭文字を取った言葉が多いため，大文字だけの単語は重みを大きく計算する

以上の計算を行うことで，議論で中心的に使われる単語及び，OSD の議論で多く用いられる用語をキーワードとして抽出することができる．

2.3 アーカイブの検索

スレッド間の結び付きを使ったアーカイブの検索は以下の手順で行う．

1. アーカイブの全文検索を行い，メールの集合を取得する
2. それぞれのメールが含まれるスレッドを求める．一般に複数のスレッドが求まる
3. 2 で求めた全文検索結果のスレッドそれぞれについて，これまでに述べた手法で結び付きにあるスレッドの検索を行う

以上の計算を行うことで，全文検索で取得したスレッドよりも多くのスレッドを取得することができる．全文検索で求めることのできないスレッドも検索できるため，より多くの情報を検索することが可能となる．

3 実装

本システムは，以下に示す 3 つのプログラムから構成される．システム構成を図 2 に示す．

- ユーザーインターフェース
  利用者とのインターフェースであり，全文検索エンジンへの検索依頼，結果表示を行う
- スレッド検索プログラム
  全文検索エンジンと連携し，2 章で述べたスレッド間の結び付きから関連スレッドを検索する
- メールデータベース作成プログラム
  アーカイブを検索可能な形式に展開し，メールからスレッドを求めるための，メール情報データベースを作成する

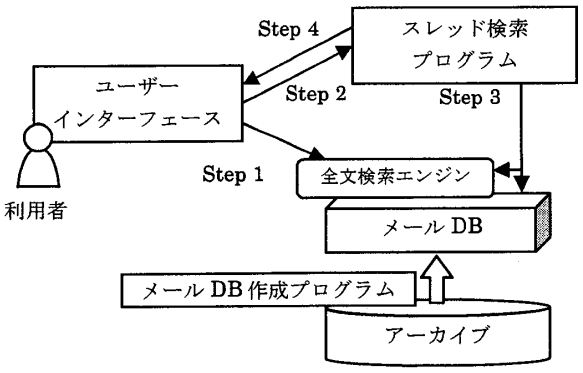

図 2　システム構成

本システムにおける検索過程は次の通りである．

Step 1. 利用者はキーワードを使って全文検索を行い，検索結果を得る
Step 2. 利用者は，全文検索結果から興味を持ったスレッドを選ぶ
Step 3. 利用者が選んだスレッドを基に，スレッド検索プログラムが関連のあるスレッドを検索する
Step 4. 検索結果が利用者に返される

なお，全文検索エンジンには Namazu[3]を用いた．

4 評価実験・考察

FreeBSD[2]の開発で実際に使用されている，freebsd-stable ML を対象に以下の実験を行った．

2002 年 1 月にやりとりされたメール 1263 通の中から，FreeBSD 4.5-RELEASE に関する問題を，試作したシステムと，Namazu を使った一般的なアーカイブ検索システムを使って検索した．そして，検索結果のスレッドの再現率，適合率の平均値を求め，f 値による比較を行った．なお，再現率とは，必要な情報のうち，実際に検索された情報の割合であり，適合率とは，実際に検索された情報のうち，必要な情報の割合である．f 値とは再現率と適合率の調和平均として定義され，情報検索の精度を測るための指標として用いられる．f 値が大きいほど，情報検索の精度は高いといえる．

実験結果を表 1 に示す．

表 1:　既存システムとの比較

	再現率	適合率	f 値
試作したシステム	26%	40%	0.32
Namazu	15%	50%	0.23

実験の結果，f 値が大きくなり，本システムによって検索の精度が上がったことが示された．また，再現率は 11%高くなり，適合率は 10%低くなった．再現率が高くなるのは，全文検索に使われたキーワードが含まれないメールも検索できたためであり，スレッド間の結び付きを使った本検索システムの有効性が示された．また，適合率が 10%低くなることは，キーワードの抽出に失敗し，関連のないスレッドも検索したためと考えられる．

5 まとめ

本研究では，メールを使った議論の単位であるスレッドに注目し，スレッド間の結び付きを使ったオープンソース開発支援用メール検索システムの設計と実装を行った．さらに，実際の開発で用いられた ML を使って評価を行った結果，既存のシステムと比べて検索の精度が向上し，本検索システムの有効性が示された．今後の課題としては，キーワード抽出を正確に行い，検索精度をさらに向上させることが挙げられる．

参考文献

[1] Eric S. Raymond, “The Cathedral & the Bazaar”, O’REILLY, 1999.
[2] The FreeBSD Project, The FreeBSD Project, http://www.freebsd.org/
[3] Namazu Project, 全文検索システム Namazu, http://www.namazu.org/