

他者のクエリログを用いた論文検索支援

伊藤和馬[†] 黄 宏軒[†] 川越恭二[†]

[†]立命館大学 情報理工学部

1 はじめに

一般に、適切な検索クエリを指定することは研究者が求める論文を検索する際に重要である。しかし、研究を始めたばかりの初学者である場合、専門的な知識が乏しく、適切な検索クエリの指定は容易ではない。その結果、研究者が求める適切な論文を取得するのが困難となる。この問題を解決するために、論文検索を支援する研究が行われている。荒木ら[1]は、二つの検索クエリ間の関連を複数の観点の類似性を用いて算出し、その関連クエリによる適合論文を推薦する手法を提案している。この手法では、研究者が適切な検索クエリを選択できない場合でも、最初に頭に思い浮かんだ検索クエリを入力することで、関連クエリや関連論文の提示が可能な手法である。しかし、この手法は個々の検索クエリ間の類似性しか考慮しておらず、研究者は適切な論文が取得できるまで検索クエリの編集を繰り返す必要があることから、論文取得まで入力する検索クエリの集合間の類似性を用いる方が、研究者が求める論文を推薦しやすいと考えられる。

そこで本研究では、他者の連続した検索クエリログを用いて研究者の論文検索を支援する手法を提案する。利用者と他の研究者の用いた連続した検索クエリログ間の類似度を算出することで、適切な検索クエリを得ることができ、その結果利用者が求める論文を取得しやすいようにする。

2 他者の検索クエリログを用いた論文検索支援手法

2.1 論文検索支援手法の流れ

本研究では、検索クエリは一つ以上の単語から構成されていることを想定する。連続した検索クエリログは、求めている論文を取得するまでに単語の追加や、削除などの編集を繰り返し入力された検索クエリの集合

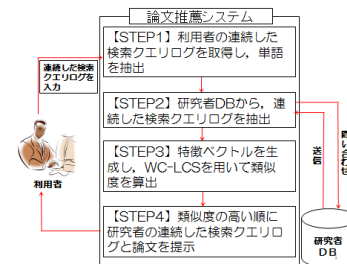


図 1: 提案手法の流れ

と定義する。

論文検索支援手法の流れを図1に示す。

STEP1 利用者が入力した連続した検索クエリログから、検索クエリを構成する単語を抽出する。

STEP2 研究者 DB から、STEP1 で抽出した単語を含む複数の連続した検索クエリログを取得する。ここで、研究者 DB とは、複数の研究者の連続した検索クエリログが格納されているデータベースである。

STEP3 利用者が入力した連続した検索クエリログと STEP2 で取得された連続した検索クエリログから各々の特徴ベクトルを生成する。この連続した検索クエリの集合間の類似度を、コサイン類似度を用いた改良型 LCS(Longest Common Subsequence)[2]WC-LCS(後述)を用いて算出する。

STEP4 算出結果から類似度の高い順に研究者の連続した検索クエリログと、論文を利用者に提示する。

2.2 特徴ベクトルの生成

本研究では、連続した検索クエリログを、ベクトル空間を用いて表現する。あらかじめ連続した検索クエリログを統一するために特徴ベクトルを生成する。利用者の連続した検索クエリログは特徴ベクトル $K_i = (K_{i1} \cdots K_{im})^t$ で示す。ここで単語が存在する場合は1、単語が存在しない場合は0とする。連続した検索クエリログ S は、以下の式(1)で表すことができる。

Research Paper Retrieval Using Search Query Logs of Other Users
Kazuma ITO[†], Hung-Hsuan HUANG[†] and Kyoji KAWAGOE[†]

[†]College of Information Science and Engineering, Ritsumeikan University.

[†]kito@coms.ics.ritsumeikan.ac.jp

[†]huang@fc.ritsumeikan.ac.jp

[†]kawagoe@is.ritsumeikan.ac.jp

$$S_l[S_{l1}, \dots, S_{lm}] = \left\langle \begin{pmatrix} K_{l11} \\ \vdots \\ K_{l1n} \end{pmatrix} \cdots \begin{pmatrix} K_{lm1} \\ \vdots \\ K_{lmn} \end{pmatrix} \right\rangle \quad (1)$$

2.3 WC-LCS(WeightingCosineSimilarity-LongestCommonSubsequence)

連続した検索クエリログ間の類似度を，特徴ベクトルから算出する．

LCS は二つの記号列間で，最長の共通部分列を求める手法である．共通部分列が長いと，二つの記号列の類似性が高いことを表す．LCS では，比較対象である二つの記号が完全一致の場合のみ一致していると判断する．完全一致では，比較する二つの検索クエリの中に同じ単語が含まれていても，一つでも違う単語が存在していれば不一致とみなされる．しかし利用者が初学者の場合，研究者 DB に格納されている検索クエリと完全に一致するような検索クエリを選択することが困難であるため，共通部分列を求めることができない．そこで，LCS で最長共通部分列を求めるときにコサイン類似度を用いて，閾値 α 以上か以下かで一致の判断を行う．コサイン類似度を用いた場合の LCS を，C-LCS(CosineSimilarity-LongestCommonSubsequence) と定義する．これにより，完全一致でない場合でも共通部分列を求めることが可能になる．

長さ x の連続した検索クエリログ a と，長さ y の連続した検索クエリログ b 間の C-LCS の長さは，以下の式 (2) で表す．

$$C-LCS(x, y) = \max \begin{cases} C-LCS(x-1, y-1) + Sim(a[x], b[y]) \\ C-LCS(x-1, y) \\ C-LCS(x, y-1) \end{cases} \quad (2)$$

ただし，

$$Sim(a[x], b[y]) = \begin{cases} Cos(a[x], b[y]) & (Cos(a[x], b[y]) \geq \alpha) \\ 0 & (otherwise) \end{cases}$$

$a[x]$ と $b[y]$ が一致しているかを判断する際に，コサイン類似度を算出し，閾値 α 以上ならばその類似度を足すことで LCS を求める．なお， $C-LCS(x, 0) = C-LCS(0, y) = 0$ である．

閾値 α 以上の類似度の共通部分列が連続している場合，それに応じて記号列による重み付けを行う．この重み付けを行った C-LCS を WC-LCS とする．具体的には，C-LCS を算出した後，共通部分列が連続する際

S_1	S_2
情報推薦	情報推薦 協調フィルタリング
情報推薦 協調フィルタリング	情報推薦 TF-IDF
情報推薦 TF-IDF 適合性フィードバック	情報推薦 コサイン類似度
協調フィルタリング 適合性フィードバック	情報推薦 TF-IDF 協調フィルタリング
協調フィルタリング TF-IDF	情報推薦 TF-IDF コサイン類似度

表 1: S_1

表 2: S_2

S_1/S_2	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0.7	0.7	0.7	0.7	0.7
2	0	1.0	1.0	1.0	1.5	1.5
3	0	1.0	1.8	1.8	1.8	1.8
4	0	1.0	1.8	1.8	1.8	1.8
5	0	1.0	1.8	1.8	2.6	2.6

表 3: 類似度計算例

に，以下に示す式 (3) の重み付けを行い，WC-LCS を算出する．

$$WC-LCS(x, y) = w(z) + C-LCS(x, y) \quad (3)$$

ここで， z は連続した共通部分の個数を表し， $w(z)$ は連続した共通部分の個数 z に依存した関数である．共通部分が連続であるということは，利用者と研究者が検索クエリを同じように編集しているということを意味する．WC-LCS により，連続している個数が多いほど，類似性は高くすることが可能となる．

説明を簡単にするために $w(z) = 0$ にしたときの WC-LCS による類似度計算例を，表 3 に示す．例として，連続した検索クエリログ S_1 を表 1， S_2 を表 2 に示す．

類似度計算例より，連続した検索クエリの集合 S_1 と S_2 の類似度は 2.6 となる．

3 おわりに

利用者と他者の連続したクエリログの類似性を用いて，論文検索を支援する手法を提案した．今後は，利用者に個々のクエリの類似度を算出し，論文を推薦される結果と，提案手法を用いて推薦される結果とを比較し，本研究の有効性を検証する予定である．

参考文献

- [1] 荒木他：”情報検索支援のための RWR による関連クエリ抽出および関連文献提示手法”，WebDB Forum 2009 論文集,(2009).
- [2] L.Bergroth, et al.: ”A survey of Longest Common Subsequence Algorithms”, SPIRE 2000.(2000).