Vol.2020-IFAT-139 No.2
2020/7/31

網羅性を重視した学術論文に対する検索手法

福田 悟志 [1,a)]　冨浦 洋一 [1]

概要：研究の新規性を確認するための学術論文検索では，膨大な論文集合から情報要求に関連する論文を網羅的に発見する必要がある．しかし，検索クエリを工夫するだけで網羅的な検索を達成するには限界がある．本研究では，LDA (Latent Dirichlet Allocation)によるトピック分析を用いることで，検索クエリに関連する論文を網羅的に発見するためのランキング手法を提案した．この手法は，トピック分析結果を介して，抄録および検索クエリ内の各語をトピックに置き換えたトピックレベルでのブーリアン検索に基づいている．そして，LDA に与えるパラメータの設定が異なる複数のトピック分析を行い，それぞれのトピック分析結果を用いてトピックベースのブーリアン検索を実行することで得られた結果を統合することで，論文をランク付けた．さらにこのランキング結果を，従来のクエリベースの情報検索モデルによるランキング結果と統合させた．NTCIR-1, 2 データセットを用いて評価した結果，一定の数の検索結果に含まれる関連論文の数という観点から，2 種類のランキング結果を統合することで，従来の検索モデルと比べて高い再現率を示した．

1. はじめに

学術論文検索では，膨大な論文集合から情報要求に関連する論文を網羅的に発見することが求められる．多くの学術情報検索エンジンでは，ユーザに対して，自身の情報要求が反映された検索クエリの入力を要求しており，システムは，そのクエリと関連する論文から順に提示している．本研究では，以下で述べる 2 つの問題点を考慮した学術論文検索を提案する．

1 つは，検索クエリの考案である．例えば，「言語資源から語の階層関係を自動的に抽出する手法について論じている論文」という情報要求に対して，まず「階層関係 AND 抽出」という検索クエリを考えたとしよう．このクエリに一致する論文は情報要求に関連している可能性は高い．しかし，例えば「階層構造を自動的に獲得する」というように，「階層関係」を「階層構造」，「抽出」を「獲得」として記述している関連論文は，このクエリにはヒットしない．このような論文は，「階層関係 AND 抽出」を「(階層関係 OR 階層構造) AND (抽出 OR 獲得)」と拡張することで収集することができる．しかし，「階層関係」や「階層構造」の代わりに「用語間の関係」といった表現や「上位下位関係」という専門用語を使用している関連論文は，この拡張した検索クエリでもヒットしない．このように，様々な論文の著者が用いている検索語の別表現を網羅的に予測して検索クエリを拡張することには限界がある．

2 つ目は，日常的に行われる情報検索と論文検索における検索の性質の違いである．日常的な検索では，情報要求に関連する情報をいくつか入手し，その後，それらの情報の中からより信憑性の高いものを精査する作業や，情報要求に関連する新たな手掛かりを獲得するための作業が行われる．このとき，入手する情報の数が多くなるほど，ユーザ側にかかる労力は大きくなるため，多くの検索システムでは，ランキング結果の上位数十位以内に情報要求に関連する文書を多くランク付けすることを重視している．実際，ベクトル空間モデル[1][2]やクエリ尤度モデル[3][4]といった情報検索における一般的な検索モデルでは，文書とクエリ間の関連性に基づいてランク付けすることで，情報要求に関連する文書を発見することを目指している．一方で，研究の新規性を確認するための論文検索では，検索結果に情報要求に関連する論文が網羅的に含まれているかどうかが重視され[5][6]，多くの場合，時間をかけて検索結果の確認作業が行われる．このとき，ユーザがチェックできないほどの数が出力されると，ユーザにとっては有用な検索結果にはならない．そのため，一定の数の論文集合(例えば上位 1,000 件の検索結果)に対する関連論文の再現率が重要になる．

本研究では，検索クエリの考案に対する負担を軽減し，かつ関連論文を網羅的に収集するための検索手法を提案する．本手法のアイディアは以下のとおりである．それぞれのアイディアの詳細は 3 節で述べる．

- Griffiths らの LDA (Latent Dirichlet Allocation)[7]によるトピック分析結果を用いたブーリアン検索
- LDA における，Simulated Annealing (SA)法[8]を用いた各単語に対する最適なトピックの確率的な探索
- パラメータの設定が異なる複数のトピック分析結果

1 九州大学システム情報科学研究院
Graduate School and Faculty of Information Science and Electrical Engineering, Kyushu University, 819-0395, Japan
a) fukuda.satoshi.528@m.kyushu-u.ac.jp

によるそれぞれのトピックベースのブーリアン検索の結果を統合して論文をランク付け

- 上記の検索手法と従来の検索モデルによるランキング結果を統合

2. 関連研究

情報要求を端的に表し，かつ関連論文を網羅的に収集できるような検索クエリの考案には，多くの時間や労力を要する．そのため，学術論文検索タスクでは，クエリベースの検索の他に，明示的な検索クエリの入力を必要としない検索システムの構築に関する研究が行われている．このような検索に関する研究では，入力に論文の表題・抄録(または本文)やユーザのプロファイル情報などを利用していることが多い．そして，この情報に基づき，データベース内の各論文との類似度[9][10]やフィルタリング技術[11]，トピックモデル[12]，ディープラーニング[13]などを用いて，ユーザが関心を持つと考えられる論文を探索している．

検索クエリをユーザ自らで考案しないという枠組みで，関連論文を効率的に検索するという研究が多く行われている一方で，Google Scholar や Web of Science など一般的に利用されている論文検索エンジンでは，単語ベースの検索クエリによる検索を採用しており，その利便性の高さは今も根強いといえる．本研究では，単語ベースの検索クエリを入力とした学術論文検索という枠組みで網羅的に関連論文を検索するための手法を提案する．

3. 本研究のアイディア

3.1 LDA によるトピック分析を用いたブーリアン検索

LDA によるトピック分析の特徴として，多くの文書中で共起している単語同士に対して同一のトピックを割り当てる傾向がある．ここで，単語 w と w' にトピックを割り当てることを考える．多くの文書中で，w と共に$w_1, w_2, \ldots$という単語が出現しているとき，これらの単語に対して同一のトピックが付与される．このとき，別の文書中で，w'が$w_1, w_2, \ldots$と共に出現している場合，$w, w_1, w_2, \ldots$に付与されたトピックと同一のトピックが w' にも付与される．これは，多くの文書で$w_1, w_2, \ldots$と共起している語は，$w_1, w_2, \ldots$と同一の文脈上で出現していることから，LDA では同一のトピックから生成される可能性が高いとみなされるためである．

本研究で提案するトピックベースのブーリアン検索では，LDA における上記の性質を利用する．w を検索語とした場合，w と同一のトピックが付与された語 w' は，w と類似した文脈で出現する関連語とみなすことができ，この w' を持つ論文は，情報要求と関連している可能性があると考えることができる．したがって，ユーザが設定した検索条件に従って検索クエリをトピックに置き換え，検索クエリのトピックを含む論文を検出することで，上記のような情報要求と関連する可能性がある論文を検出することができる．

3.2 温度を導入したギブスサンプリングの利用

LDA では，文書は単語の系列であり，各単語はトピックから生成されると仮定している．$\mathbf{w}^{(d)} = (w_1^{(d)}, \cdots, w_{l_d}^{(d)})$を d 番目の文書に対する単語の系列，$\mathbf{z}^{(d)} = (z_1^{(d)}, \cdots, z_{l_d}^{(d)})$を d 番目の文書内の各単語に付与されるトピックの系列，総文書数を D とする．このとき，文書集合 $\mathbf{w}\ (= (\mathbf{w}^{(1)}, \cdots, \mathbf{w}^{(D)}))$が与えられたとき，各単語に付与されたトピックの系列が $\mathbf{z}\ (= (\mathbf{z}^{(1)}, \cdots, \mathbf{z}^{(D)}))$である確率は，以下のように算出される．

$$P(\mathbf{z}|\mathbf{w}, \alpha, \beta) = \frac{P(\boldsymbol{w}, \boldsymbol{z}|\alpha, \beta)}{\sum_{\boldsymbol{z}} P(\boldsymbol{w}, \boldsymbol{z}|\alpha, \beta)}. \quad (1)$$

[7]では，ギブスサンプリングにより，(1)式に従った $\mathbf{z}$ が確率的に生成される．そのため，サンプリングが終了したときに生成される $\mathbf{z}$ は，必ずしも$P(\mathbf{z}|\mathbf{w}, \alpha, \beta)$を最大にするものではない，すなわち，各語に付与されたトピックは，尤もらしいものではない場合がある．文書内で特定のトピックを持つ語が生成される確率や特定のトピック内で語が発生する確率を推定する場合，(1)式に従った $\mathbf{z}$ をサンプリングし，これを利用して事後分布の期待値として推定すれば良い．しかし，トピックベースのブーリアン検索では，各語に付与されたトピックに基づいて関連論文を発見するため，各語に付与されたトピックが正しいもの，すなわち，$P(\mathbf{z}|\mathbf{w}, \alpha, \beta)$を最大にする $\mathbf{z}$ を求める必要がある．

そこで本手法では，SA 法を用いて，$P(\mathbf{z}|\mathbf{w}, \alpha, \beta)$を最大にする $\mathbf{z}$ を近似的に求める．SA 法では温度という概念を導入しており，十分に大きな値の温度から徐々に 0 に近づけることで，最適解あるいは最適解と近似した最良解を導出する．詳細は 4.2 節で述べる．

3.3 複数のトピックベースのブーリアン検索結果を統合したランキング

LDA を実行する際，いくつかのパラメータを事前に設定する必要がある．特に，トピック数および(1)式におけるα, βはトピック分析の結果に大きく影響する．しかし，検索対象となる文書集合に応じてLDA に与える最適なパラメータは異なり，検索時に最適なパラメータを人手で設定することは困難である．一方で，トピックベースのブーリアン検索に対する予備実験を通して，パラメータを細かく調整したとしても，特定の高い確率で同一のトピックから生成される単語グループおよび特定の割合以上で共通のトピックを含む文書グループの大まかな傾向は変わらない可能性があることが分かった．

そこで本研究では，検索対象となる抄録集合ごとに最適な LDA のパラメータを与える代わりの方法として，1 つの抄録集合に対して複数のトピック分析結果を取得し，それぞれの分析結果を用いてトピックベースのブーリアン検索を実行する．そして，それぞれの抄録を，各検索結果に含まれている数に基づいてランク付けする．これによって上位にランク付けられる抄録は，様々なパラメータ設定に対する検索結果に出現する論文であることから，安定的な単

語および抄録間の関係に基づいた関連論文と考えられる．本研究では，このランキング手法を Topic Rank と名付ける．

3.4 2 種類のランキング結果を統合したリランキング

Topic Rank では，複数のトピックベースのブーリアン検索による検索結果に従って論文をランク付けする．そのため，この方法によるランキング結果は，文書からクエリが生成される確率で文書をランキングする従来の検索モデル(クエリ尤度モデル)による結果とは異なる傾向を示す可能性が高い．そこで，実際に，2 種類の検索モデルによるランキング結果を比較し，両モデルにおける論文の出現傾向を調べた．図 1 に，ランキング結果を上位から 5%の区間で分割したときの比較結果を示す．左側の縦軸は，各区間に含まれる論文に対する関連論文の割合を表す．右側の縦軸は，各区間における，各検索モデルによりランク付けされた論文集合間の重複率を表す．比較手法には，本手法と同様に LDA を用いている Wei らのクエリ尤度モデル[4]を用いた．このモデルの詳細は，5.2 節で述べる．なお，重複率はダイス係数により算出した[*1]．

図 1 から，各検索モデルにおけるランキング結果の上位に関連論文が比較的多く出現していることから，それぞれのモデルにおいて高くランク付けられている論文は関連論文である可能性が高いといえる．一方で，論文の重複率は，上位 5%および 5%から 10%においてそれぞれ 0.237, 0.097 と比較的低いことから，それぞれの検索モデルにおけるランキング結果の傾向は大きく異なっている可能性が高い．そのため，2 種類の検索モデルによるランキング結果を統合することで，それぞれのランキング結果とは異なるリランキング結果が得られると考えられる．

4. 複数のトピック分析結果を用いたトピックベースのブーリアン検索結果によるランキング

まず，検索クエリと検索対象となる論文の抄録集合がユーザから与えられる[*2]．その後，ユーザから与えられた抄録集合に対して前処理を行う．次に，前処理された抄録集合に対してトピック分析を行う．このとき，LDA のパラメータの設定が異なる複数のトピック分析を行い，それぞれのトピック分析結果に基づいてトピックベースのブーリアン検索を実行する．最後に，検索結果を統合し，論文をランク付ける．以下では，各手順の詳細を説明する．

4.1 検索クエリの定義および前処理

本手法における検索クエリ Q の形式は以下の通りである．

$$Q_1 \text{ OR } Q_2 \text{ OR } \cdots \text{ OR } Q_k$$

また，Q_kは，以下の形式で表される．

$$(q_1 \text{ OR } q_2 \text{ OR} \cdots q_m) \text{ AND } (q'_1 \text{ OR } q'_2 \text{ OR} \cdots q'_n) \text{ AND } \cdots$$

[*1] データセットには，NTCIR-1 における課題番号("0040")において，正解判定が与えられている 1,909 件の論文データを用いた．

[*2] 例えば，ジャーナルの表題や会議名を複数指定し，それに合致する論文の表題や抄録等の書誌情報を論文データベースから API を用いて収集することで，対象の抄録集合を与えることができる．

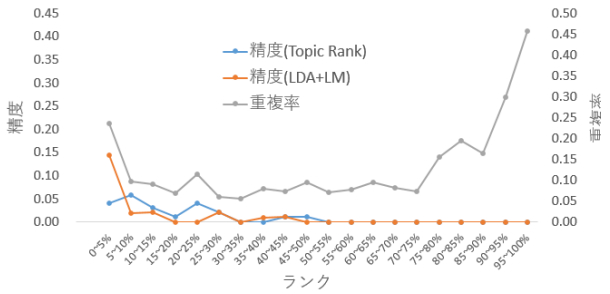

図 1 本研究で提案するランキング手法(Topic Rank)と Wei らのモデル(LDA+LM)による精度と重複率の推移．

ここで，q_mやq'_n は検索語を表し，各括弧内において OR で結合されている語は，同一の検索概念を表す同義語・類似語とする．なお，1 つの情報要求に対して複数の形式のクエリが考案される場合があることを考慮し，上記のように複数のクエリQ_kを OR で結合させることを許容している．

前処理では，Q 内の各括弧を，抄録集合に出現していない特別な記号(シンボル)に置き換える．これは，トピック分析において，OR で列挙された同義語や類似語を表す検索語に同一のトピックが割り当てられるようにするためである．その後，各抄録内に出現する検索語を，その語に割り当てられたシンボルに置き換える．

4.2 LDA によるトピック分析

LDA によるトピック分析を行う際，$P(\mathbf{z}|\mathbf{w},\alpha,\beta)$を最大にする $\mathbf{z}$ を SA 法に基づいて求める．すなわち，ギブスサンプリングにより以下の値に比例する確率で $\mathbf{z}$ をサンプリングすることで求める．

$$P(\mathbf{z}|\mathbf{w},\alpha,\beta)^{\frac{1}{T}}$$

ここで，T は温度を表す．また，サンプリングが行われるごとに，以下のようにTを更新する．

$$T = T * R$$

R はTを更新するためのパラメータである．

T を更新するにあたって，SA 法では，R は十分小さく，また，繰り返し回数は十分大きくする必要がある．6.2 節で，本実験における各パラメータの決定の詳細を述べる．

4.3 トピック分析結果を用いたブーリアン検索

ここでは，2 つのステップから，検索クエリ Q に対するブーリアン検索の検索結果を求める．まず，各Q_kに対する検索結果を獲得する．そして，獲得した各検索結果の和集合を算出し，これを出力する．以下では，Q_kに対する検索結果の獲得について述べる．

4.1 節で述べた前処理の結果，Q_kが以下のように変換されているとする．

$$\text{AND}_{i=1}^{I} SYM_i$$

ここで，I は，4.1 節の前処理において，Q_kに対して生成されたシンボルの数を表し，$\text{AND}_{i=1}^{I}$は，シンボルSYM_iの AND

結合を表す．トピック分析結果に基づいて，このクエリをトピック型の検索クエリに変換するのであるが，トピック分析において一つの語に付与されるトピックは一般に文書によって異なる．ユーザが入力したQ_kと完全一致で検索できる抄録，すなわち，すべての SYM_i $(i = 1, 2, \cdots, I)$ が出現している抄録は情報要求に関連している可能性が高いと仮定し，そのような文書で SYM_i に付与されているトピックをユーザが指定したクエリにおける検索語のトピックとする．つまり，すべての SYM_i $(i = 1, 2, \cdots, I)$ が出現している抄録集合において，SYM_i に付与されているトピックを $t_{i,1}, t_{i,2}, \cdots, t_{i,J_i}$ とすると，シンボル型の検索クエリを以下のトピック型のクエリに変換する．

$$\mathrm{AND}_{i=1}^{I}\left(t_{i,1}\ \mathrm{OR}\ t_{i,2}\ \mathrm{OR}\ \cdots\ \mathrm{OR}\ t_{i,J_i}\right)$$

例えば，クエリが「$(q_1$ OR $q_2)$ AND q'」であり，$(q_1$ OR $q_2)$ が *SYM1*，q'が *SYM2* と置き換えられ，前処理後に *SYM1* と *SYM2* が共に出現している抄録が抄録Aと抄録Bであったとする．また，抄録 A では *SYM1* と *SYM2* にそれぞれ Topic 0 と Topic 1 が，抄録 B では *SYM1* と *SYM2* にそれぞれ Topic 0 と Topic 2 が付与されていたとする．この場合，*SYM1* は Topic 0，*SYM2* は(Topic 1 OR Topic 2)と置き換えられ，「Topic 0 AND (Topic 1 OR Topic 2)」というクエリが構築される．

トピック型の検索クエリを構築した後，このクエリと各抄録が持つトピックを比較し，クエリの条件と完全一致する抄録を持つ論文を検索結果として出力する．例えば，抄録 C における各単語に付与されたトピックが Topic 0，Topic 2，Topic 3 であった場合，この抄録は先のトピック型の検索クエリ「Topic 0 AND (Topic 1 OR Topic 2)」と一致するため，検索結果として出力される．

4.4 複数の検索結果を統合したランキング

ランキングの手順は以下の通りである．まず，LDA に与えるパラメータの設定が異なるトピック分析を複数行い，それぞれのトピック分析の結果において，4.3 節で述べたトピックベースのブーリアン検索を行う．そして，検索結果に含まれている回数に基づき，各論文を降順でソートする．

例えば，パラメータ(α, β, K)を$(0.1, 0.1, 10)$，$(0.5, 0.1, 10)$，$(0.1, 0.5, 10)$と設定し，各検索結果に論文 A, D, 論文 A, B, D, 論文 A, B, C, D が含まれていたとすると，論文 A, B, C, D, E が検索結果に含まれていた回数は，それぞれ 3, 2, 1, 3, 0 回となる．そして最終的に，論文 A と D は 1 位，論文 B は 3 位，論文 C は 4 位，論文 E は 5 位とランク付けされる．

5. 2 種類のランキング手法の統合

5.1 リランキング

本研究におけるリランキング方法は，以下の通りである．

$$\textbf{Hybrid} \quad \tilde{r} = r_1 + r_2 \tag{2}$$

r_1 は Topic Rank により決定された順位，r_2 は既存の検索モデルによりランク付けされた順位を表す．なお，複数の抄録に同一の $\tilde{r}$ が与えられた場合，同率の順位として扱う．

5.2 情報検索における一般的な検索モデル

(2)式におけるr_2を求めるために，本研究では，以下に述べる検索モデルを用いた．なお実験では，検索クエリ Q における各Q_kにおいて，AND で結合されている各括弧内の OR で結合されている検索語から 1 つずつ抽出し，クエリQ_rとして構築する．そして，すべての抽出パターンにおけるQ_rに対してランキング結果を獲得した後，各抄録に対して与えられたランクの総和に基づき昇順でランク付けを行い，その結果を Q に対するランキング結果として獲得した．

(1) ベクトル空間モデル

ベクトル空間モデルは，検索クエリと文書が表すベクトル間の成す角を求めることでクエリとの類似性を測定するモデルであり[1]，コサイン類似度は，ベクトル空間モデルにおける代表的な計算手法である．クエリQ_rおよび各抄録のベクトルを表す方法として，以下の 2 種類を用いた．

- **Bag-of-words (COS(BOW))**

Bag-of-words による方法では，クエリQ_rと抄録に出現する単語を要素としてベクトル化する．各要素に対する重みは，TF-IDF により決定した．

- **Word embedding (COS(WE))**

Word embedding を用いた方法では，学習の際に設定した次元数がベクトルの要素となる．[2]では，以下によりテキスト(クエリQ_rまたは抄録) t をベクトルで表現している．

$$\vec{t} = \frac{\sum_{j=1}^{|V|} \overrightarrow{w_j} \cdot TF\_IDF(w_j, t)}{\sum_{j=1}^{|V|} TF\_IDF(w_j, t)}$$

ここで，$\overrightarrow{w_j}$はw_jの分散表現を表す．なお，分散表現の獲得には，word2vec[14]を用いた．

(2) クエリ尤度モデル

クエリ尤度モデルは，文書 d がクエリQ_rに適合する確率$P(d|Q_r)$を，ベイズの定理を利用して

$$P(d|Q_r) \propto P(Q_r|d) \cdot P(d)$$

とし，さらに$P(d)$を一様とみなして

$$\mathrm{P(d}|Q_r) \propto P(Q_r|d)$$

として，$P(Q_r|d)$に基づいて d をランク付けするモデルである．$P(Q_r|d)$を求める際，d を直接的に用いることができないため，クエリ尤度モデルでは，多項分布を用いて d をユニグラム言語モデルθ_dで表すことが多い．θ_dから Q_rが生成される尤度$P(Q_r|\theta_d)$は以下のように表される．

$$P(Q_r|\theta_d) = \prod_{i=1}^{|Q_r|} P(q_i|\theta_d)$$

上記の式は，Q_r内の単語 $w_j \in V = \{w_1, \ldots, w_{|V|}\}$の出現回数$c(w_j, Q_r)$を考慮して，以下のように変形することができる．

$$P(Q_r|\theta_d) = \prod_{j=1}^{|V|} P(w_j|\theta_d)^{c(w_j, Q_r)}$$

この$P(w_j|\theta_d)$の算出に対して，本研究では，以下に述べる 2 種類の方法を用いた．

- **Zhai らの検索モデル (LM)**

Zhai らは，以下のように，ディリクレスムージングによって，文書 d 中に出現しない語に対して確率値を割り当てるクエリ尤度モデルを提案した[3].

$$P(w_j|\theta_d) = \frac{N_d}{N_d+\mu}P_{ML}(w_j|\theta_d) + \left(1-\frac{N_d}{N_d+\mu}\right)P(w_j|\theta_C)$$

ここで，N_dは d における総語彙語数，$P_{ML}(w_j|\theta_d)$は d におけるw_jの最尤推定，$P(w_j|\theta_C)$は文書集合 C におけるw_jの最尤推定，μはスムージングパラメータである．

- **Wei らの検索モデル (LDA+LM)**

Wei らは，以下のように，LDA による推定結果を用いることで，語に対する同義語や類似語といった関連語を潜在的に考慮したクエリ尤度モデルを提案した[4].

$$P(w_j|\theta_d) = \lambda\left(\frac{N_d}{N_d+\mu}P_{ML}(w_j|\theta_d) + \left(1-\frac{N_d}{N_d+\mu}\right)P(w_j|\theta_C)\right) + (1-\lambda)\left(\sum_{k=1}^{K}P(w_j|t_k)P(t_k|d)\right)$$

ここで, λ, μ はスムージングパラメータである．$P(w_j|t_k)$はトピック t_k から w_j が出現する確率, $P(t_k|d)$は文書 d がt_kを持つ確率であり，ギブスサンプリングにより推定される．

6. 実験

6.1 実験データ

実験には，NTCIR-1, 2 で提供される情報検索用テストコレクションを用いた[15][16]．これらのテストコレクションには，264,153 件の英語論文および 132 件の検索課題が含まれている．また，各検索課題には，約 800 から 5,000 件の学術論文に対して，その課題に対する「高適合」「適合」「部分適合」「不適合」のラベルが付与されている．

本研究では，「高適合」「適合」「部分適合」が付与された論文を関連論文として実験を行った．また，ユーザがチェックできる論文の数の限界を 1,000 件と考え，さらに，ユーザが研究の新規性を確認する場合に，関連論文の数が極端に少ないというケースは稀であると考え，ラベルが付与されている論文が 1,000 件未満または関連論文を 10 件未満しか含まない検索課題を除いた．そしてその中から，一人の被験者が課題内容を解釈することで検索クエリを考案できた 41 件の検索課題を用いた．また，TreeTagger[17]を用いて 264,513 件の抄録内の各単語を原形に戻し, 品詞を解析した．そして，品詞が名詞，動詞，形容詞，数詞以外の単語，1 件の抄録にしか出現しない単語，ストップワードリスト[*3]に含まれる単語を除去した．また，同一のユニグラム分布を持つ抄録が含まれていたため[*4]，これらを除外し，最終的に 259,550 件の英語論文を用いた．

6.2 実験設定

1 節でも述べたように，論文検索では，検索結果に対する再現率が重要になる．そのため，ランキング結果における上位 100, 200, 500, 1,000 件を検索結果として獲得するときに対する再現率で評価した．ここで，同率順位(n 位)の論文集合によって出力される論文の数が設定した件数を超過することを考慮し，以下のように再現率の期待値を測定した．

$$\frac{X+\left(\frac{Y}{Z}\right)\times W}{RP}$$

RP は関連論文の総数, X は$n-1$位までの論文集合に含まれる関連論文の数を表す. Y は n 位の論文集合に含まれる関連論文の数, Z は n 位の論文集合に含まれる論文の数, W は検索結果として獲得する数に達するまで n 位の論文集合から獲得する論文の数を表す.

次に，各手法におけるパラメータ設定について述べる．まず Topic Rank において，LDA に与えるパラメータは，$\alpha, \beta \in${0.01, 0.02, 0.05, 0.1, 0.2, 0.5}，$K \in${6, 7, 8, 9, 10, 11, 12, 13, 14, 15}と設定し，すべての組み合わせにおけるトピック分析結果を用いた．また，実験で用いた 41 件の検索課題においてラベルが付与されていた論文の数が最も多かった課題番号"0004"における 4,780 件の抄録集合を対象に，$\alpha = 0.01$, $\beta = 0.01, K = 15$と設定したときの$P(\mathbf{z}|\mathbf{w},\alpha,\beta)^{\frac{1}{T}}$の推移を観察し，$T$, R, サンプリング回数をそれぞれ 5.0, 0.99990, 30,000 回と設定した．また，LM および LDA+LM では，$\alpha, \beta \in${0.01, 0.02, 0.05, 0.1, 0.2, 0.5}，$K \in${6, 7, 8, 9, 10, 11, 12, 13, 14, 15}，$\lambda \in${0.1, 0.3, 0.5, 0.7, 0.9}，$\mu \in${10, 20, 30, 40, 50}, 設定し，LDA のギブスサンプリングの回数は，30,000 回とした．さらに，COS(WE)において，次元数は 300, 350, 400, ウィンドウサイズは 6, 8, 10 と設定し，word2vec の学習データには 259,550 件の英語論文抄録を用いた．

5.2 節で述べた検索モデルおよび 5.1 節で述べたリランキングでは最適なパラメータを設定する必要があるため，41 分割交差検定により評価した．パラメータの決定方法には grid search を用い，ランキング結果における累積再現率のグラフの面積が最も大きくなるときのパラメータを使用した．

6.3 実験結果

実験結果を表 1 に示す．表 1 では，41 件の検索課題に対する再現率のマクロ平均を示している．表 1 を見ると，検索結果として獲得する論文数を 100 としたとき，従来手法の中で最も高い再現率を示した LDA+LM と比べて，Hybrid (LDA+LM)により 3.88%向上した．また，検索結果として獲得する論文数を 200, 500, 1,000 としたとき, 従来手法の中で最も高い再現率を示している COS(WE)と比べて，Hybrid (COS(WE))により，それぞれ 7.05%, 2.77%, 1.60%向上した．これらの結果から，リンランキング手法を用いて 2 種類のランキング結果を統合することで，関連論文の網羅性を高めることができることが分かった．

[*3] https://www.ranks.nl/stopwords (Default English stopwords list 内の語を用いた．)
[*4] プロシーディング名が異なるのみで, 抄録内容が同一である論文がテストコレクションに存在した．また，同一のユニグラム分布を持つ抄録同士において, 特定の検索課題に対するラベルとして適合と不適合の両方が付与されていたものも存在したため，除外した．

表 1　ランキング結果から検索結果として獲得する論文集合に対する再現率

	検索結果として獲得する論文数			
	100	200	500	1,000
Topic Rank	0.4181	0.5778	0.8086	0.9447
COS(BOW)	0.3943	0.5229	0.7364	0.9006
COS(WE)	0.4221	0.5797	0.8332	0.9503
LM	0.3845	0.5122	0.7135	0.8753
LDA+LM	0.4405	0.5726	0.7964	0.9484
Hybrid (COS(BOW))	0.4791	0.6151	0.8489	0.9545
Hybrid (COS(WE))	0.4686	**0.6502**	**0.8609**	**0.9663**
Hybrid (LM)	0.4745	0.6170	0.8403	0.9515
Hybrid (LDA+LM)	**0.4793**	0.6418	0.8587	0.9616

表 2　Topic Rank において検索クエリと完全一致かつ情報要求と関連する論文のみを用いてトピック型検索クエリを構築した場合の再現率の比較

	検索結果として獲得する論文数			
	100	200	500	1,000
Topic Rank	0.5293	0.6746	0.8651	0.9570
Hybrid (COS(WE))	0.5432	0.7003	0.8884	**0.9671**
Hybrid (LDA+LM)	**0.5498**	**0.7024**	**0.8903**	0.9664

Topic Rank では, ユーザが考案した検索クエリと完全一致する論文集合に基づいてトピック型のクエリを構築し，ブーリアン検索およびランク付けを行っている．しかし実際には，クエリと完全一致する論文の中でユーザの情報要求と関連していないものも少なくない．そこで，検索クエリと完全一致し，かつ「高適合」「適合」「部分適合」のラベルが付与された論文集合のみからトピック型のクエリを構築することで, Topic Rank およびリランキングにどのような影響を与えるか調べた．表 2 に結果を示す[*5]．リランキングに適用する検索モデルとして, COS(WE)と LDA+LM を用いた．表 1 と表 2 の結果を比較すると，情報要求と関連する論文のみを用いることで，全体的に再現率が向上していることが分かる．特に, 検索結果として獲得する論文数を 100, 200, 500 としたとき, 表 2 で示している Topic Rank の再現率は，表 1 で示した Topic Rank と比べて，それぞれ 11.12%, 9.68%, 5.65%と大幅に向上している．また, Hybrid(COS(WE))においても, 表 1 の Hybrid(COS(WE))における再現率と比べて，それぞれ 7.46%, 5.01%, 2.75%向上しており，Hybrid(LDA+LM)でも，それぞれ 7.05%, 6.06%, 3.16%向上している．これらの結果から，Topic Rank において，トピック型のクエリを構築する前のプロセスとして，クエリと完全一致する論文集合に対して適合・不適合の判定する作業を導入することで，さらなる検索性能の向上が見込まれる．

7. おわりに

本研究では，ユーザにより作成された検索クエリおよび LDA に与えるパラメータの設定が異なる複数のトピック分析結果を用いて，論文をランク付けるための手法を提案した．さらに，この手法と情報検索タスクで広く利用されている検索モデルによる 2 種類のランキング結果を統合し，リランキングを行った．そして，NTCIR-1, 2 データセットを用いた実験から，ランキング結果を統合することによる全体的な検索性能の向上を確認した．

[*5] 41 件の各検索課題において, 考案された検索クエリと完全一致で検索される論文集合には 1 件以上の関連論文が含まれており，表 1 と同様の評価を表 2 では行った．

謝辞　この研究は科研費 JP15H01721 および JP19K20629 の助成を受けたものである．

参考文献

[1] Salton, G. and McGill, M. (Eds.): Introduction to Modern Information Retrieval, McGraw-Hill (1983).
[2] Kosmopoulos, A., Androutsopoulos, I., and Paliouras, G.: Biomedical Semantic Indexing Using Dense Word Vectors in BioASQ, Biomedical Semantics (2015).
[3] Zhai, C. and Lafferty, J.: A Study of Smoothing Methods for Language Models Applied to Information Retrieval, ACM Transactions on Information Systems, Vol. 22, No. 2, pp. 179–214 (2004).
[4] Wei, X. and Croft, W.B.: LDA-based Document Models for Ad-hoc Retrieval, Proc. SIGIR, pp.178–185 (2006).
[5] Kim, Y., Seo, J., and Croft, W.B.: Automatic Boolean Query Suggestion for Professional Search, Proc. SIGIR, pp. 825–834 (2011).
[6] Verberne, S., Sappelli, M., and Kraaij, W.: Query Term Suggestion in Academic Search, Proc. ECIR, pp.560–566 (2014).
[7] Griffiths, T.L. and Steyvers, M.: Finding Scientific Topics, Proc. National Academy of Sciences, pp. 5228–5253 (2004).
[8] Kirkpatrick, S., Gelatt Jr, C.D., and Vecchi, M.P.: Optimization by Simulated Annealing. Science, Vol. 220, No. 4598, pp. 671–680 (1983).
[9] Takaku, M. and Egusa, Y.: Simple Document-by-Document Search Tool “Fuwatto Search” Using Web API, Proc. ICADL, pp. 312–319 (2014).
[10] Nascimento, C., Laender, A.H.F., Silva, A.S., and Gonçalves, M.A.: A Source Independent Framework for Research Paper Recommendation, Proc. JCDL, pp. 297–306 (2011).
[11] Alzoghbi, A., Ayala, V.A.A., Fischer, P.M., and Lausen, G.: Learning-to-Rank in Research Paper CBF Recommendation: Leveraging Irrelevant Papers, Proc. RecSys, pp. 43–46 (2016).
[12] Amami, M., Pasi, G., Stella, F., and Faiz, R.: An LDA-based Approach to Scientific Paper Recommendation, Proc. Natural Language to Information Systems, pp. 200–210 (2016).
[13] Hassan, H.A.M.: Personalized Research Paper Recommendation Using Deep Learning, Proc. UMAP, pp. 327–330 (2017).
[14] Mikolov, T., Chen, K., Corrado, G., and Dean, J.: Efficient Estimation of Word Representations in Vector Space, Proc. ICLR Workshop (2013).
[15] Kando, N., Kuriyama, K., Nozue, T., Eguchi, K., Kato, H., Hidaka, S., and Adachi, J.: The NTCIR Workshop: The First Evaluation Workshop on Japanese Text Retrieval and Cross-lingual Information Retrieval, Proc. Information Retrieval with Asian Languages Workshop (1999).
[16] Kando, N.: Overview of the Second NTCIR Workshop, Proc. NTCIR Workshop, pp. 35–43 (2001).
[17] TreeTagger Homepage, http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/