

質問に対する回答者推薦手法に用いられるデータの期間についての一検討

眞鍋 雄貴
熊本大学

y-manabe@cs.kumamoto-u.ac.jp

西原 弘樹
熊本大学

c5846@st.cs.kumamoto-u.ac.jp

要旨

ユーザ間で質問回答を行うプラットフォームとして質問回答コミュニティ (CQA) がある。CQA では日々多くの質問に対して、回答が行われる。一方で、回答が行われない質問も多くある。この問題を解決するため、回答者推薦手法が提案されている。特に、複数の手法を組み合わせた回答者推薦手法が多く提案され、特に、トピック分析とネットワーク分析を組み合わせた手法が多くを占める。しかしながら、これらの手法の評価では、推薦に用いるデータの期間が考慮されていないこと、また、新たな質問をテストデータにしていないものがある。これらは、回答者推薦手法がユーザにとって有用であるかを評価するには必要な観点である。そこで、本研究では、このような観点から回答者推薦手法を評価する第一歩として、プログラマ向け CQA である *Stack Overflow* において、質問や回答を抽出する期間を基準となる日の直前 3ヶ月、6ヶ月、9ヶ月、12ヶ月として得られたデータセットを用い、トピック分析とネットワーク分析を組み合わせた回答者推薦手法の結果がどのように変わるのかを調査した。その結果、3ヶ月の場合にのみ正しく推薦を行えた質問があった。また、この質問に対して回答者として推薦されたユーザのうち 3名は基準日直前 2ヶ月間は活動を行っていなかったことが示された。

1. はじめに

ユーザ間で質問、回答を行うプラットフォームを提供するウェブサイトとして質問回答コミュニティ (CQA) がある。CQA には特定のドメインに依存しないもの (Yahoo 知恵袋など) の一方で、特定のドメインに特化した CQA が多く設立されている。特定のドメインに特化した CQA の

うち、プログラマ向け CQA で最大の CQA として *Stack Overflow* がある。*Stack Overflow* では、プログラミングに関する多様なトピックについてユーザ間で質問や回答がなされている。ユーザのアカウント数は 2019 年 2月 11 日時点で、9,663,406 人、投稿された質問の数は 17,167,300、1 日あたりの質問投稿件数が 73000 (2019 年 3月 14 日時点) となっている。

しかし、CQA には大きく二つの制約がある。一つは、CQA では多くの質問に回答が見つからないままである。*Stack Overflow* では 2019 年 3月 14 日時点では 29% の質問に回答がついていないことである¹。もう一つは、CQA はどのようなユーザが質問に回答するかわからないため、回答の品質も様々に変わることである。

これらの問題に対し、その分野の専門知識を持つユーザが質問に回答できるよう、このようなユーザに適切な質問が提示されるように、CQA における新たな質問に対する適切な回答者を推薦する手法が提案されている。近年では、複数の回答者推薦のアプローチを組み合わせた手法が提案されており、特に、トピック分析とネットワーク分析を組み合わせた手法が多く提案されている [1, 2, 3, 4, 5]。しかし、これらの手法の評価では推薦に用いるデータがどのような期間で取得されたかについての検討が不十分である。回答者推薦手法が適切に働き、また、質問を行うユーザや回答を行うユーザにとって有用なものとするには、以下の 2 点について評価を行うのが望ましい。

- どの程度の期間のデータが効率の良い推薦に必要なか。
- 最新の新規の質問に対して、推薦を行うことができるか。

前者は、回答者推薦手法が有効に働くにはどの程度の

¹<https://stackexchange.com/sites?view=list#answers>

データが必要となるかの前提を示している。後者は、回答者推薦手法の目的がどの程度達成されているかを示すものである。

本研究では、CQA を用いるユーザの観点から回答者推薦手法がどの程度有用であるかを評価する最初の一步として、期間が異なる複数のデータセットに対して回答者推薦手法を用いることによって、どのように推薦結果が変化するかを調べた。本研究では、3ヶ月、6ヶ月、9ヶ月、12ヶ月（1年）と期間の異なるデータセットを用意した。また、各データセットのうち、最新1ヶ月間に行われた質問からランダムに100件選び、このうちベストアンサーが設定されていた47件をテスト用のデータとした。回答者推薦の手法としては、トピック分析の手法として Twitter-LDA[6] を、ネットワーク分析の手法として HITS[7] を組み合わせた手法を用いた。結果として、3ヶ月の期間のデータを利用した場合が最も良い精度を示した。

2. 質問回答コミュニティ

質問回答コミュニティ（以後 CQA）は対話的な質問への回答が行われることに基づくコミュニケーションプラットフォームを提供するサービスであり、ユーザが世界中の人々との知識の交換を可能としている [5]。また、数多くの質問や回答を蓄積しているため、ウェブサービスエンジンでは容易に得られない価値のある資源を提供している [8]。現在、多くの CQA が設立されており、多くの CQA をホストしている Stack Exchange では、174 のコミュニティがホストされている。各 CQA では、多くの場合各 CQA では扱う質問の分野が決まっている。例として、Stack Overflow はプログラマの質問に答え、Arqade ではビデオゲーマの質問、Seasoned Advice ではシェフの質問に答えるコミュニティだとしている。

Stack Overflow のスクリーンショットを図 1 に示す。Stack Overflow では質問にタグをつける機能、回答を第三者が編集する機能のほか、その質問もしくは回答に、高評価や低評価をつけたりする機能を提供している。上部には質問の情報が記録されており、上からタイトル、本文、タグ、質問者情報が記載されている。下部には回答の情報が記載されており、上から総回答数、本文、回答者情報、コメントが記載されており、ベストアンサーには左端にチェックマークがついている。

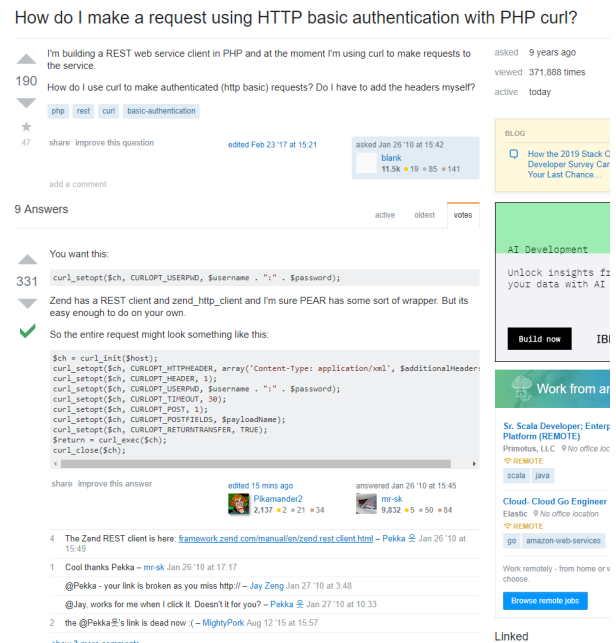


図 1. Stack Overflow のスクリーンショット

3. 本研究で使用するデータ

本研究では、Stack Exchange Explore¹から取得したデータを用いた。Stack Exchange Explore は Stack Exchange がホストする CQA のデータのアーカイブを保存するサイトである。保存されるデータについては、各 CQA の投稿内容だけでなく、メタデータも保存される。

本研究では、Stack Overflow に焦点を当て、そのうち、User に関するデータ 10097976 件が保存されている StackOverflow-Users.xml と、投稿内容に関するデータが保存されている StackOverflow-Posts.xml を用いた。StackOverflow-Users.xml は各ユーザのデータが xml 形式で格納されている。各ユーザのデータが持つ属性を図 1 に示す。

一方、StackOverflow-Posts.xml は質問と回答についてのデータ 43369425 件が xml 形式で格納されている。質問と回答についてのデータが持つ属性を図 2 に示す。属性には、質問と回答共通の属性、質問のみにある属性、回答のみにある属性がある。特に、回答と質問の対応関係については、回答の ParentId を調べる必要がある。一方、タグは質問のみに Tags として記録されるため、回答がどのタグに関するものかを知るには対応する質問の

¹<https://archive.org/download/stackexchange>

属性名	説明
Id	各ユーザの ID
Reputation	ユーザの信用度
CreationDate	アカウントが作成された日時
DisplayName	ユーザ名として表示される文字列
LastAccessDate	アカウントに最後にアクセスした日時
WebsiteUrl	ユーザが登録したウェブサイトの URL
Location	ユーザが登録した場所
AboutMe	ユーザが登録した紹介文
Views	プロフィールの閲覧回数
UpVotes	ユーザが得た高評価の回数
DownVotes	ユーザが得た低評価の回数
ProfileImageUrl	ユーザが登録したユーザの画像
AccountID	ユーザの Stack Exchange Network でのアカウントの ID

表 1. StackOverflow-Users.xml における属性

データを見る必要がある。また、OwnerUserId に着目すると、あるユーザが何度質問もしくは回答を行ったかを調べることができる。

これらのファイルは巨大であり、またユーザや質問回答の数も多いため、データを選択した。その上で、それぞれのファイルを実験で用いることができるように、前処理を行い、データベースへの格納を行った。以下、各ステップについて述べる。

3.1. データの選択

本研究では、[5] での実験設定を踏まえ、'scala' をタグに持つ質問とその回答を用いた。また、使用するデータの期間は、2018 年 12 月 2 日から 1 ヶ月前までの範囲にある質問をテストデータ、2018 年 12 月 2 日から 3 ヶ月前、6 ヶ月前、9 ヶ月前、1 年前までのデータのうち、テストデータの期間を除いた部分を学習データとして用いる。また、テストデータとなる質問は、最新 1 ヶ月の質問のうち 100 の質問を無作為に選んだ。

3.2. データベースへの格納

データの加工の全体像を示す。巨大な XML ファイルを分割し、それを CSV ファイルに変換する。次に CSV ファイルに基づいてデータベースを作成する。最後にそのデータベースから必要な情報を読み出す。

初めに、XML ファイルを 100000 行ごとに分割し、それぞれ CSV ファイルに変換する。次に、各 CSV ファイルをデータベースに格納する。まず、今回使用したタグである 'scala' をタグに持つ質問をデータベースに取り込む。その後、その質問に対する回答を取得し、データベー

属性名	説明
Id	C 各投稿の id
PostTypeId	C 投稿の種類。
	C 主なものは 1:質問, 2:回答。
CreationDate	C 投稿が行われた日時
Score	C 投稿のスコア
Body	C 投稿の本文
OwnerUserId	C 投稿を行ったユーザの ID
OwnerDisplayName	C 投稿を行ったユーザの名前
LastEditorUserId	C 最後に編集したユーザの ID
LastEditorDisplayName	C 最後に編集したユーザの名前
LastEditDate	C 最後に投稿が編集された日時
LastActivityDate	C 最後に投稿に対して変更、 回答などが行われた日時
CommentCount	C 投稿に対して行われたコメントの数
CommunityOwnedDate	C コミュニティ Wiki に 投稿された日時
AcceptedAnswerId	Q 質問を投稿したユーザが ベストアンサーとして マークした回答の ID
ViewCount	Q 質問の閲覧回数
Title	Q 質問のタイトル
Tags	Q 質問につけられたタグ
AnswerCount	Q 質問に対してつけられた回答の数
FavoriteCount	Q 質問に対してユーザが お気に入りにした数
ClosedDate	Q (クローズされた質問について) クローズされた日時
ParentId	A 回答が対応する質問の ID

表 2. StackOverflow-Posts.xml における属性。C: 質問、回答共通属性、Q:質問のみの属性、A:回答のみの属性を意味する

スに追加する。3.1 節に示したとおり、本研究では 'scala' タグがついている質問を用いるが、表 2 で示されている通り、回答には 'scala' タグは付いていない。そのため、回答の ParentId から対応する質問を得て、その上で質問の Tags に 'scala' があるかを見ている。

3.3. 実験用データの生成

次に、作成したデータベースからユーザごとの回答履歴を抽出し、ユーザごとにテキストファイルに書き出す。たとえば、ユーザ ID が 100 のユーザの回答のリストは、100.txt となる。同時に、作成したテキストファイルのリストをテキスト形式で出力している。テストデータとして使用する質問についても、テキストファイルをそれらのファイルパスを記載したファイルリストを作成する。最後に、それらのテキストファイルから、<p>のような HTML タグや、ソースコードの部分を除外する。各データセットにおける回答者数、ユーザ数、回答数、質問数を表 3 に示す。

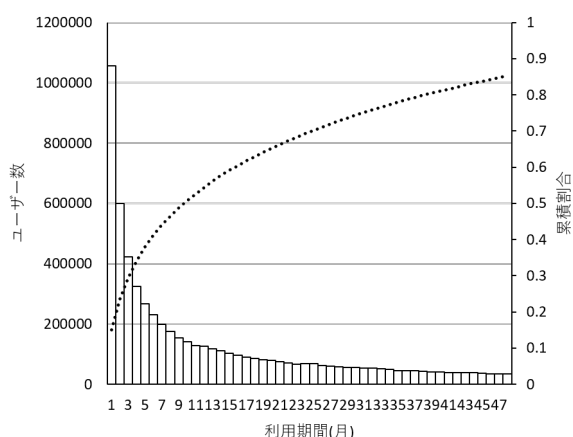


図 2. 利用期間ごとのユーザ数

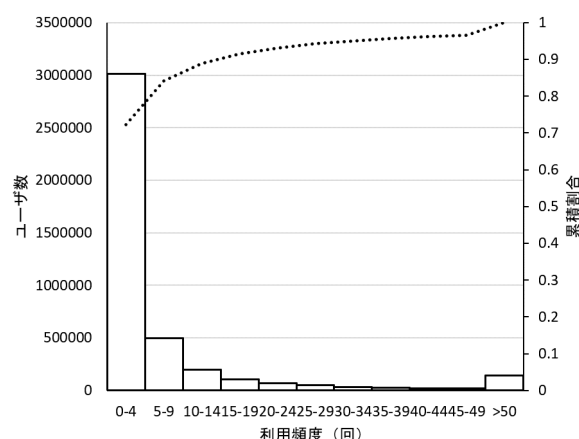


図 3. 利用頻度ごとのユーザ数

4. 予備実験

本研究での問題意識を確認するため、予備実験を行った。予備実験として、Stack Overflow のユーザの利用期間に対する分布を調べた。利用期間は、最後に質問、もしくは回答を行なった日時から、最初にアカウントを作成した日時の差を月単位で表したものとした。利用期間が 48 ヶ月までのユーザの数を図 2 に示す。図 2 は利用期間ごとのユーザ数を表しており、横軸が期間、縦軸がユーザ数である。

図 1 の結果から、ユーザのアカウント利用期間は、3 ヶ月以内が 29.5%、6 ヶ月以内が 41.2%、9 ヶ月以内が 48.8%、12 ヶ月以内が 54.4%となっている。このことから、長期間にわたるデータを用いて推薦を行なったとしてもすでに利用していないユーザを推薦する可能性がある。一方で、1 ヶ月以内のユーザも 15.0%いるため、推薦に用いるデータに存在せず、推薦できないユーザも要る可能性があることを示している。

表 3. 各データセットにおける回答者数、ユーザ数、回答数、質問数

	3months	6months	9months	1year
answerers	784	1659	2545	3257
users	1820	3952	6119	7880
answers	2204	5545	9794	13361
questions	966	966	966	966

これに加え、利用頻度についても調査した。ユーザの利用頻度に対する分布を図 3 に示す。利用頻度は、各ユーザがアカウントを作成してからの質問の回数と回答の回数の合計とした。図 3 は利用頻度ごとのユーザ数を表しており、横軸が質問や回答を行った回数、縦軸がユーザ数である。図は、50 回未満のユーザが 96.6%を占めていること、また、72.2%のユーザは 4 回以下の活動のみ行っていることを示している。また、50 回以上のユーザが行った投稿について調査したところ、質問、回答全体のうち、55.6%を占めていた。これは一部のユーザが大部分の質問や回答を行っていることを意味する。このことが、推薦結果に影響を及ぼす可能性がある。

5. 関連研究

回答者推薦手法に関するサーベイとして、Wang ら [9] のサーベイがある。本サーベイでは、回答者推薦手法を Simple Method, Language Models, Topic Models, Network-Based Methods, Classification Methods, Expertise Probabilistic Models, Collaborative Filtering Methods, Hybrid Methods と分類している。Simple Method は、ユーザに単純な尺度でスコアをつけることによって推薦を行う手法である。尺度の例としては、投票、ベストアンサーの割合、テキストの類似度がある。Language Models は言語モデルによる推薦を行う手法であり、主に query likelihood language model (QLL)[10] とその改良モデルが用いられる。Topic Models は、トピック分析によって推薦を行う手法であり、Probabilic

Latent Semantic Analysis(PLSA)[11] や Latent Dirichlet Allocation(LDA) モデル [12] やその改良が用いられる。Network-Based Methods は、ネットワーク分析による推薦を行う手法であり、PageRank や HITS とその改良が用いられている。Classification Methods は、分類により、エキスパートのクラスと、そうでないクラスにユーザを分割する手法である。Expertise Probabilistic Models は、Dom ら [13] によって提案されているベイジアン確率モデルを使用してユーザの信頼性の事後推定を取得し、特定の質問の専門家を紹介する手法である。Collaborative Filtering Methods は強調フィルタリングを回答者推薦に利用しようとする手法である。HybridMethods は今までに挙げた手法を複数組み合わせる推薦を行うものである。

HybridMethods に属する回答者推薦手法は近年多く提案されており、特に、Topic Models と Network-Based Methods を組み合わせた手法が多い [1, 2, 3, 4, 5]。Zhou ら [1] はトピック分析手法として LDA、ネットワーク分析手法として PageRank を用いた Topical PageRank を提案している。Zhao ら [2] は Topic Model と Network Analysis を組みあわせており、トピック情報を得るために TEL を用いる。TEL は LDA に基づいたモデルで、グラフに基づくリンク分析と内容に基づく意味解析を組み合わせている。Yang ら [3] はトピック分析手法として LDA を、ネットワーク分析手法として HITS を改良した NEWHITS を用いている。Zhou ら [4] はトピック分析手法として LDA を用いるが、質問に対してだけでなく、ユーザに対するトピックを抽出する際にも用いる。また、ネットワーク分析手法としては PageRank を用いる。Li ら [5] の手法がある。Li らは Twitter など短文集合に対応するトピック分析手法である Twitter-LDA[6] を元にした Question-LDA と、ネットワーク分析手法である HITS[7] を改良した NEWHITS を組み合わせた回答者推薦手法を提案している。

本研究では、最も多いトピック分析とネットワーク分析手法の組み合わせが重要な組み合わせであると考え、そのうち、実装が公開され²、短い文章でも有効であるとされる Twitter-LDA[6] をトピック分析手法として用い、ネットワーク分析においては、シンプルな HITS を初期段階の検討対象として選択した。

その他の組み合わせとして、ネットワーク分析とクラスタリングを組み合わせた手法 [14]、言語モデル、トピッ

ク分析、ネットワーク分析を組み合わせた手法 [15] がある。Bougessa ら [14] はエキスパート推薦のためにネットワーク分析とクラスタリングを組み合わせている。本手法では、ユーザ間のネットワークを質問者とその質問にベストアンサーを提示したユーザ間の関係をグラフとし、グラフの重みは始点ノードに対応するユーザが終点ノードに対するユーザの質問へベストアンサーを示した回数である。Liu ら [15] は言語モデル、トピック分析、ネットワーク分析を組み合わせた CQA ランクを提案している。本モデルは言語モデルである QLL と LDA モデルを関連性を分析するために使用し、それに加えユーザの活動やオーソリティの情報が推薦に考慮される。本研究では、これらの組み合わせについては検討の対象外である。

6. 実験

6.1. 本実験で用いる回答者推薦手法

初めに、ユーザごとの回答履歴とテストデータとなる質問を Twitter-LDA で処理し、回答者を推薦したい質問のトピック行列 (行列 WT) と、ユーザの興味を表すトピック行列 (行列 UT) を取得する。

次に、データベースを利用してユーザ間のリンクを取得し、HITS を実行する。これによって、ユーザのオーソリティスコアを計算する。

最後に、あるユーザにおいて行列 WT と行列 UT を掛け合わせる。その結果に、そのユーザのオーソリティスコアをかける。これによって、あるユーザがある質問に対してどれだけ回答する能力を持っているか計算できる。この結果を行列 QR とし、式 (1) に示す。

$$QR = WT \times UT^T \times AuthorityScore \quad (1)$$

これを全てのユーザ、全てのテストデータの質問において繰り返すことで、テストデータの質問ごとの推薦者 5 人を取得する。

6.1.1 トピック行列の算出

トピック行列の算出では、行列 UT と行列 WT を得る。本ステップの入力として、すでに推薦対象の質問の集合と、各ユーザの回答履歴を用い、これらの文書に対してどのトピックがどの程度関連があるかを示す行列を

²<https://github.com/minghui/Twitter-LDA>

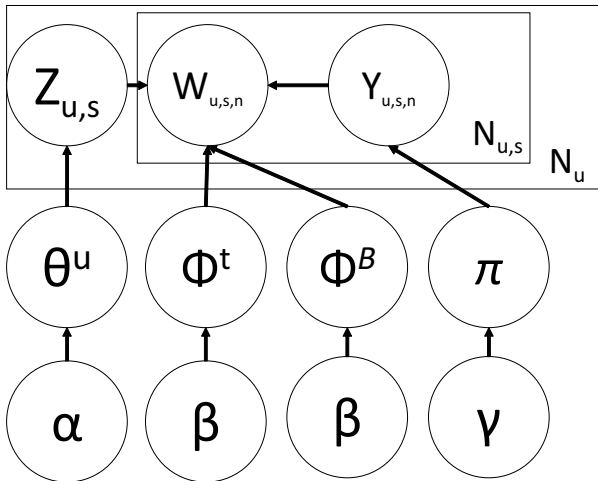


図 4. Twitter-LDA の生成モデル

得る。そして、この行列のうち、推薦対象の質問に対するトピックの対応を示す部分が行列 WT，ユーザの回答履歴に対するトピックの対応を示す部分が行列 UT となる。

推薦対象の質問の数を Q ，推薦対象のユーザの集合を U ，トピックの集合を T とすると，行列 WT は $|Q| \times |T|$ 行列，行列 UT は $|U| \times |T|$ 行列となる。

Twitter-LDA は Blei ら [12] によって考案された LDA を Twitter のような短文から構成される文書集合に対して適用するために改良したトピック分析手法である [6]。従来の LDA が持つ短文の解析の精度という弱点を，トピックの選定には利用しない単語を作成することで処理している。Twitter-LDA の生成モデルを図 4 に示す。

入力にはテキストの文書データが用いられ，出力は各単語の出現確率である。パラメータは，トピックの選択確率を得るためのパラメータである α ，トピックに応じて単語の生成確率を得るためのパラメータである β がある。加えて，Twitter-LDA ではパラメータ γ が導入される。 γ は通常の単語とトピックの選定に利用しない単語のベルヌーイ分布である π を導くために使用される。

6.1.2 Authority Score の算出

HITS とは，ネットワーク構造を測るための手法である [7]。本来はインターネットのページの構造を分析するための手法で，Hub Score と Authority Score という二つの値を結果に持つ。優秀な Hub は優秀なページに対し

て多くリンクを持ち，優秀な Authority は多くのページからリンクを持つ。この考えに基づき，Authority Score は，そのページに対してリンクをしているページのハブスコアの合計値によって求められ，Hub Score は，そのページがリンクを行っているページの Authority Score の合計値によって求められる。グラフの頂点をユーザ，質問したユーザからその質問に回答を行ったユーザへ辺を取ったグラフ (V, E) における Hub Score と，Authority Score の計算式を式 (2) と (3) に示す。

$$HubScore(p) = \sum_{q:(q,p) \in E} AuthorityScore(q) \quad (2)$$

$$AuthorityScore(p) = \sum_{q:(p,q) \in E} HubScore(q) \quad (3)$$

全ての Hub Score，もしくは Authority Score の二乗和で割ることで，この式を正規化し，値が収束するまで繰り返す。

6.1.3 推薦結果の算出

これまでのステップで得られた，行列 WT，UT，AuthorityScore の積を式 4 で取ることにより，ユーザと文書間の対応関係を示す行列 QR が得られる。この QR において，推薦対象となる質問に対応する部分を見ることで，その質問にどの程度各ユーザが適しているかが値で示されることになる。これを，数値が大きい順にユーザをランク付けし，その上位が推薦結果となる。この式から，QR の値は，回答者の推薦を行いたい質問に関連するトピックの質問に多く答え，また，多くのユーザの質問の答えるユーザに対して大きくなりやすいことが示されている。

$$QR_{mn} = A_m \sum_{t \in T} UT_{mt} WT_{nt} \quad (4)$$

6.2. 実験方法

実験では，3ヶ月，6ヶ月，9ヶ月，12ヶ月（1年）分抽出したデータセットに対し，Twitter-LDA と HITS を用いた回答者推薦を行い，精度を評価する。精度の尺度として，precision，recall，F-score を用いる。各尺度の定義を，式 5，6，7 に示す。ただし，正解の予測をして実際に正解だった結果の数を TP，不正解だった結果の数を FP とする。不正解の予測をして，実際に不正解だった結果の数を TN，正解だった結果の数を FN とす

る.⁵ 本実験では、ベストアンサーとなった回答をした回答者を正解の回答者とし、節に示した手法を用いて推薦された結果と比較する。

$$\text{precision} = \frac{TP}{TP + FP} \quad (5)$$

$$\text{recall} = \frac{TP}{TP + FN} \quad (6)$$

$$\text{F-score} = \frac{2\text{recall} \cdot \text{precision}}{\text{recall} + \text{precision}} \quad (7)$$

6.2.1 実験環境

実験のために、Twitter-LDA を入手した。これは、GitHub に共有されていたコードである。Twitter-LDA は、以前述べた通り、[6] によって研究された LDA の派生型で、短文分析もこなすことができる。こちらは、Windows10 および、java、そして java 用の IDE である eclipse を使用した。³ また、先ほど示した実験用のデータセット加工の部分と、次の実験手順で示すネットワーク分析と各ユーザの質問に対するオーソリティスコアの計算の部分については、python3.6 および Ubuntu on Windows⁴ を使用した。これは、Windows 上で Ubuntu のコマンドを利用できるアプリケーションである。

6.2.2 パラメータの設定

LDA のパラメータは [1] に従い、 α を 0.1、 β を 0.01、トピックの数を 50、iteration を 30 とした。また、 γ については、初期値の 0.20 を用いた。

6.3. 結果

以下に実験の結果を示す。テストデータの 100 の質問のうち 53 の質問は、ベストアンサーが指定されていない、回答がついていない、削除されていた、などの理由で無効であった。そのため、有効な推薦は 47 となった。このうち、5 つの推薦結果の中に正解を含んでいたものは、3 ヶ月の期間において 5 つ、それ以外の期間においては 2 つだった。表 4 に、使用データの期間それぞれにおける、precision, recall, F-score を示す。この結果が

ら、本研究で想定したネットワーク分析とトピック分析を組み合わせた手法において、回答者の推薦はできており、また、2 ヶ月分のデータが学習データとして得られていれば一定の精度が得られると言える。

表 4. 期間ごとの precision, recall および F-Score

	3months	6months	9months	1year
precision	0.021	0.009	0.009	0.009
recall	0.106	0.043	0.043	0.043
F-Score	0.035	0.015	0.015	0.015

6.4. 考察

表 5. 質問 53224244 における推薦結果

3months	6months	9months	1year
4993128	2707792	2707792	2707792
6867048	7662670	5880706	4993128
5516172	4204198	6316508	5880706
7662670	4993128	9953316	6316508
2750966	5880706	4204198	7662670

表 6. 質問 53224244 におけるユーザ 5516172 の推薦順位

3months	6months	9months	1year
3	14	24	26

表 5 に 3 ヶ月のデータセットを使ったときのみ正解を推薦した質問 53224244 における推薦されたユーザのリストを示す。それぞれのリストにおいて、重複している id が多く存在する。一方で、正解となる 5516172 は 3 ヶ月のデータセットにのみ存在している。これを深く調べるため、表 7 に各ユーザが期間ごとに何回質問・回答を行ったかを示す。各列は、1 行目に記載されている日から一ヶ月前までの間に各ユーザが行った質問・回答回数を示している。これを見ると、ユーザ 27707792、2750966、4993128、6316508、7662670 はこの 1 年間どの期間においても活動が行われている。また、ユーザ

⁵<http://ibisforest.org/index.php?F> 値

³<https://github.com/minghui/Twitter-LDA>

⁴<https://www.microsoft.com/ja-jp/p/ubuntu/9nblggh4msv6?activetab=pivot:overviewtab>

jp/p/ubuntu/9nblggh4msv6?activetab=pivot:overviewtab

表 7. 推薦されたユーザが期間ごとに行った質問・回答数

Id	11/13- 12/12	10/13- 11/12	9/13- 10/12	8/13- 9/12	7/13- 8/12	6/13- 7/12	5/13- 6/12	4/13- 5/12	3/13- 4/12	2/13- 3/12	1/13- 2/12	12/13- 1/12
2707792	11	18	34	37	60	62	86	49	103	103	120	0
2750966	17	10	13	7	2	3	19	14	15	15	4	0
4204198	0	0	2	10	25	24	6	4	10	10	0	0
4993128	18	27	31	25	16	5	16	28	32	20	32	0
5516172	9	42	20	0	0	0	0	0	0	0	0	0
5880706	0	0	5	10	16	44	51	68	95	95	81	26
6316508	9	21	29	31	20	34	28	33	31	31	22	21
6867048	38	23	37	2	0	0	0	0	1	1	1	2
7662670	20	23	31	34	28	26	22	24	6	6	6	0
9953316	0	2	29	3	50	8	0	0	0	0	0	0

5516172, 6867048, 9953316 に関しては、一部の期間のみに活動が見られる。

一方で、5880706, 9953316 については、テストデータとなる 1 ヶ月間に一度も活動がない。

表 6 に質問 53224244 におけるユーザ 5516172 の推薦順位を示す。3 ヶ月以外のデータでは順位が落ちている。これは、本ユーザが 3 ヶ月以内のみ活動期間があるからと考えられる。

本研究では、トピック分析とネットワーク分析を組み合わせた手法のうちシンプルなものを想定して実験を行っている。しかし、多くの質問に対して回答者推薦を失敗していることから、この手法そのものが実用的とは言えない。しかし、本研究では、ユーザが利用することを想定した評価を行うことを主眼としており、精度そのものの値よりもその変化が重要である。今後、このような観点から新たな回答者推薦手法の評価を行い、より精度の高い手法の構築も重要となる。

7. 結論

本研究では、CQA における回答者推薦手法のうち、複数の手法を組み合わせた手法について、推薦に用いる期間とその推薦結果に関して一検討を行った。実験では、3 ヶ月、6 ヶ月、9 ヶ月、1 年のデータセットを作成し、それぞれについて本研究で想定したトピック分析とネットワーク分析を組み合わせた手法を用いた。その結果として、3 ヶ月のデータセットでのみ正しく推薦された質

問があった。これは、正解となるユーザが直近 3 ヶ月のみ活動していたためである。

今後の課題として、本研究では一つの手法を想定して結果を得ていることから結果の一般化が難しい点がある。そのため、既存の手法をより分析し、回答者推薦手法を構成する要素と推薦結果との関係が使用するデータセットの期間によりどのように変わるかを見る必要がある。また、今回のデータセットの期間は 3 ヶ月区切りとなっている。これをより長期間、また粒度を小さくしたときの変化も重要である。

謝辞 本研究は JSPS 科研費 18H04094 の助成を受けたものです。

参考文献

- [1] Guangyou Zhou, Kang Liu, and Jun Zhao. Topical authority identification in community question answering. In *Pattern Recognition*, pp. 622–629, 2012.
- [2] Tong Zhao, Naiwen Bian, Chunping Li, and Mengya Li. Topic-level expert modeling in community question answering. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pp. 776–784.
- [3] J. Yang, S. Peng, L. Wang, and B. Wu. Finding experts in community question answering based

- on topic-sensitive link analysis. In *Proceedings of 2016 IEEE First International Conference on Data Science in Cyberspace (DSC)*, pp. 54–60, June 2016.
- [4] Guangyou Zhou, Siwei Lai, Kang Liu, and Jun Zhao. Topic-sensitive probabilistic model for expert finding in question answer communities. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM '12*, pp. 1662–1666, 2012.
- [5] Lin Wang, Bin Wu, Juan Yang and Shuang Peng. Personalized recommendation for new questions in community question answering. *ASONAM '16 Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 901–908, 2016.
- [6] Ee Peng LIM, Hongfei YAN, Wayne Xin ZHAO, Jing JIANG, Jianshu WENG, Jing HE and Xiaoming LI. Comparing twitter and traditional media using topic models. *European Conference on Information Retrieval ECIR 2011: Advances in Information Retrieval*, pp. 338–349, 2011.
- [7] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM, Vol. 46, No. 5, September*, pp. 604–632, 1999.
- [8] Fatemeh Riahi, Zainab Zolaktaf, Mahdi Shafiei and Evangelos Milios. Finding expert users in community question answering. *WWW '12 Companion Proceedings of the 21st International Conference on World Wide Web*, pp. 791–798, 2012.
- [9] Xianzhi Wang, Chaoran Huang, Lina Yao, Boualem Benatallah, Manqing Dong. A survey on expert recommendation in community question answering. *Journal of Computer Science and Technology July 2018, Volume 33, Issue 4*, pp. 625–653, 2018.
- [10] David R. H. Miller, Tim Leek, and Richard M. Schwartz. A hidden markov model information retrieval system. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '99*, pp. 214–221, 1999.
- [11] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '99*, pp. 50–57, 1999.
- [12] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research 3*, pp. 993–1022, 2003.
- [13] Dom B, Paranjpe D. A bayesian technique for estimating the credibility of question answerers. In *Proc. SIAM International Conference on Data Mining*, pp. 399–409, 2008.
- [14] Mohamed Bouguessa, Benoît Dumoulin, and Shengrui Wang. Identifying authoritative actors in question-answering forums: The case of yahoo! answers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '08*, pp. 866–874, 2008.
- [15] Liu Yang, Minghui Qiu, Swapna Gottipati, Feida Zhu, Jing Jiang, Huiping Sun, and Zhong Chen. CQArank: Jointly model topics and expertise in community question answering. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, CIKM '13*, pp. 99–108, 2013.