グラフ型データベースを用いた Wikipedia における不適切リンクの修正手法

大政 飛鳥† 井上 潮‡

†東京電機大学 工学研究科 情報通信工学専攻 〒120-8551 東京都足立区千住旭町 5 番
‡東京電機大学 工学部 情報通信工学科 〒120-8551 東京都足立区千住旭町 5 番
E-mail: †18kmc07@ms.dendai.ac.jp, ‡inoueu@mail.dendai.ac.jp

あらまし Wikipedia は誰でも自由に記事の編集や閲覧が行えるインターネット百科事典であり, 記事には編集者による関連記事へのリンクが存在する. 多くの利用者による共同編集が記事の網羅性や即時性に寄与する一方, リンク先記事を間違えたりその内容が他の編集者によって大幅に変更または移動されたりした結果, リンク先が不適切なものが生じる. 本論文では, 利用者の利便性および記事間リンク構造解析研究の精度向上を目的として,不適切なリンクの修正案を利用者に提示する手法を提案する. 既存研究で問題となっていた機械学習のための不適切リンクのサンプル収集に対し, 提案手法ではその発生パターンを基にグラフ型データベースを用いて擬似的にその生成および特徴量の抽出を行う. 日本語版 Wikipedia に対して提案手法を適用し修正精度の評価による有効性の検証を行った.

キーワード Wikipedia , Web 構造マイニング, グラフ型データベース, ランキング学習

1. はじめに

Wikipedia は誰でも自由に記事の編集や閲覧が行えるインターネット百科事典である. Wikipedia は既存の百科事典と比較して次の 3 つの特徴を持つ. Wikipedia の統計[1]によれば、2019 年現在, 日本語版の記事数は 110 万件を超えており, 月に 4 千件ほどのペースで増えている. この膨大な記事数は他の百科事典にはない大きな特徴のひとつである. 英語版に至っては 590 万件を超えておりブリタニカ百科事典の 12 万件と比較してもその数は遥かに多く, 幅広いトピックを網羅している. また, 紙の百科事典と比べて随時更新される Wikipedia は情報の即時性も高い. 2 つ目は多数のボランティアによる共同編集である. Web ブラウザを使って誰でも簡単に記事の編集に参加できる Wikipedia では, 1 つの記事が複数の利用者によって編集されていることは珍しくない. 記事によっては 1 文単位での細かな情報の追加や訂正も頻繁に行われており, 多くの編集者の存在が前述の網羅性および即時性を支えている. Wikipedia 日本語版では 1 万 3 千人以上のアクティブユーザーが記事の編集に参加しており, 記事数の多さでは 300 以上の言語で運営されている Wikipedia の中で 13 番目であるのに対し編集者の数では 5 番目と特に多い. 3 つ目は関連する記事同士がリンクで繋がれている点である. 紙の百科事典においてもその辞典に記事がある言葉に印が付けられているものもあるが, Web を使った Wikipedia では記事に編集者が作成した他の記事へのハイパーリンクが存在し, 読者はそれをクリックするだけで容易に他の記事へ移動することができる.

しかし誰でも編集できることから, Wikipedia の信頼性は古くから議論の的になってきた[2]. Wikipedia の信頼性はブリタニカ百科事典と同等であるという 2005 年に公開された Nature による調査[3]がある一方で, 質の低い記事が存在することもまた事実であり, 荒らし行為との戦いも常に続いている. 数々の課題がある中でも本研究が扱うのは記事中に存在する不適切リンクの問題である.

ここで言う不適切なリンクとはリンク先の記事として適当ではない記事が指定されているものを言い, その原因は編集者同士の連携不足からリンク作成時の単純なミスまで種々考えられるが, どちらの場合も多義性がある単語についての記事が関係していることが多い. 不適切なリンクの存在は利用者の利便性や Wikipedia への信頼性を損なうだけでなく, Wikipedia のリンク構造解析を行う研究[4,5]にも悪影響を与える. また, 日々更新される膨大な記事を人手でチェックしていくのは現実的ではなく, 不適切なリンクの自動的な修正手法が求められる. そこで本論文では不適切なリンクの修正案を機械的に作成し, 利用者に提示する手法を提案する.

既存研究[6,7]では不適切なリンクのデータセット作成が大きな課題となっていた. それに対し本研究では不適切リンクの発生パターンをもとに擬似的にその作成を行う. 疑似データの作成にはネットワーク構造のデータの処理に適したグラフ型データベースを使用し, それをランキング学習モデルで学習させた.

本論文の構成は以下の通りである. 2 章では不適切リンクの修正に関する既存研究について述べる. 3 章

では本研究が用いるグラフ型データベースとその構築方法について説明し，4 章で不適切リンクの例と対応する発生パターンについて述べる．5 章ではそのパターンに基づいた疑似訓練データの作成方法を示し，6 章でその学習方法について述べる．最後に 7 章でまとめを行う．

2. 関連研究

不適切リンクの修正に関する研究は，語義の曖昧性解消やエンティティリンキングに関する研究，またはリンク予測の研究と深い関わりがある．不適切なリンクは多義性を持つ語について意図する意味とは異なる別の用法についての記事へリンクしてしまったものが多い．よって不適切リンクの修正は語義の曖昧性を解消することと考えることができる．語義の曖昧性解消と関連するエンティティリンキングではテキスト中の単語を Wikipedia の記事などにマッピングするタスクであり，不適切リンクが持つアンカーテキストについてのエンティティリンキングと見なすこともできる．また，間違いが含まれるリンク構造から，適切なリンク構造を予測するリンク先予測としても考えられる．このように不適切リンクの修正は様々な見方をすることができるが，ここでは不適切なリンクの修正に目的を絞っている 2 つの研究について述べる．

Pateman ら[6]は記事間リンク構造をもとに不適切リンクの発見および修正を行う手法を提案している．また，論文の中では不適切リンクのサンプル収集の困難さおよび重要性にも触れており，効率的に探す方法として，過去の Wikipedia のスナップショットと比較してリンクの変化を調べる方法について検討している．

Wang ら[7]は不適切な可能性のあるリンクを抽出するための LinkRank アルゴリズムを提案した．また，その中から実際に不適切であるリンクの検出および修正を行うために，記事間リンク構造と本文から得られる特徴量を用いてペアワイズ手法によるランキング学習モデルを学習させている．しかし，そのモデルの学習には人手によるラベル付けされたデータセットが必要であり，その作成は容易ではない．

本研究では不適切リンク修正のためのランキング学習モデルを人力で作成したデータではなく，その発生パターンをもとにグラフ型データベースを用いて作成した疑似訓練データで学習させる手法を提案する．

3. グラフ型データベースとその構築

本研究では訓練データの作成にグラフ型データベースの Neo4j[8]を用いた．グラフ型データベースはグラフ理論に基づいたデータベースで，データそのものだけでなくデータ間の関係もリレーションシップ（エッジ）として予め静的に格納しているので，リンクを主体とするネットワーク構造のデータを効率的に探索することができる．Neo4j はオープンソースのグラフ型データベースで，最短経路探索や全文検索機能を備えており，本研究では特徴量の抽出に利用する．

データベースを構築するために，Wikipedia が公開している日本語版の全ページを含んだ XML ファイル（pages-articles.xml, 2019 年 6 月 1 日取得）から記事とカテゴリ，そしてそれらを繋ぐリンクを自作のパーサーを用いて抽出した．記事およびカテゴリはその名称を，記事間リンクについてはアンカーテキストやそのリンクが関連項目の中にあるかどうか，箇条書きのリストの項目であるかといった情報も合わせて取得している．記事の中には表記ゆれや同義語等に対応するためのリダイレクト記事が存在し，そこへアクセスすると実際に存在する記事へと自動的に転送される仕組みになっているが，リンク先がリダイレクト記事であるリンクの参照先はリダイレクト先の記事に書き換えた上でデータベースに格納している．また，カテゴリの中には記事の内容に基づかない，メンテナンスのための「隠しカテゴリ」が存在するが，インポートするデータからは取り除いた．

図 1 はデータベースにインポートしたデータの一部を Neo4j で表示した様子である．青色のノードは通常の記事を表し、橙色はカテゴリ，紫色はリダイレクト記事を表している．これらのノード間にはリレーションシップ（リンク）が予め定義されていることがわかる．また，構築したデータベースの概要を表 1 に示す．

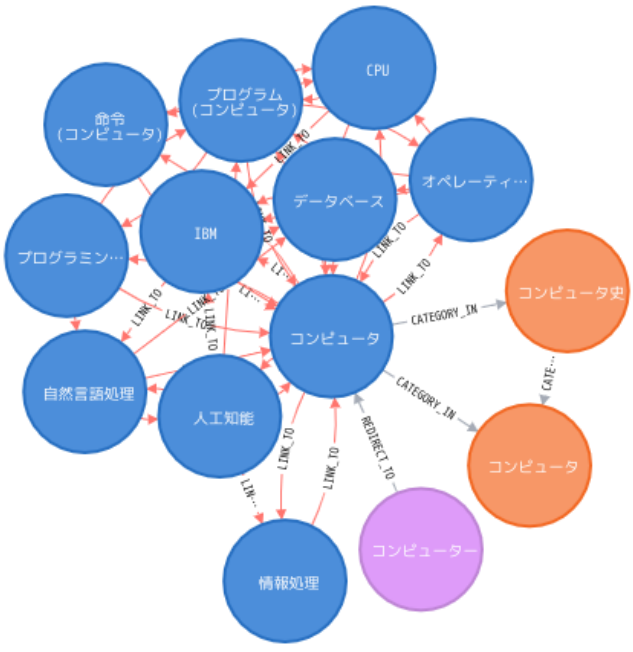


図 1. 記事「コンピュータ」まわりの構造

表 1. 構築したデータベース

種類	件数
記事	1,153,615
カテゴリ	217,150
記事間リンク	48,997,047
カテゴリ間リンク	5,468,287

4. 不適切リンクの例とその発生パターン

ここでは不適切なリンクの具体例とその発生パターンをリンク作成時に発生する場合と作成後に不適切なリンクになってしまう場合の 2 つに分けて説明する．不適切なリンクが発生する背景には多義性を持つ言葉の存在がある．Wikipedia は 1 つの概念が 1 つの記事に対応しており，例えば「モデル」という言葉については「モデル (職業)」や「モデル (自然科学)」のように用法別に記事が作成されている．モデルという言葉は他にも「模型」という意味で使われる場合や職業としてのモデルの中でも「ファッションモデル」や「読者モデル」等を指す場合があり，単に「モデル」という記事にアクセスすると図 2 のように，これらの用法別の記事を列挙した曖昧さ回避ページが表示される．曖昧さ回避のページへのリンクは適切な用法別の記事へのリンクに張り替えることが推奨されており [9]，本研究ではこの曖昧さ回避のページへのリンクも不適切なリンクとして扱う．

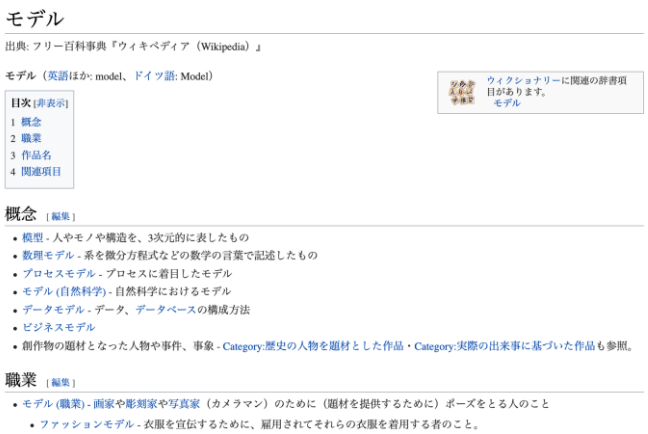


図 2. 曖昧さ回避ページ

4.1. リンク作成時に発生する場合

2019 年現在，記事「飛行機」の中には「…排気反力に加え、タービンで大径ファンを駆動し…」という一文があるが，ファンというリンクをクリックすると愛好者の意味でのファンついて解説している記事へと飛んでしまう．これは記事の編集者がリンク先記事の内容を確認しないままリンクを作成したことが原因であると考えられる．

4.2. リンク作成後に不適切なリンクになる場合

リンク作成時は適切なリンクだったものの，後に不適切なリンクになってしまったものも存在する．例えばリンク作成時点でのリンク先記事が複数ある用法のひとつのみを扱っていたとする．後に他の編集者によって別の用法についての記事が作成された等の理由でリンク先記事が曖昧さ回避ページに変更され，もともとの記事が別名の記事に移動された場合，リンク先を変更しなければ不適切なリンクとなってしまう．基本的には記事の統合や移動が発生する際にはリンク先が正しく引き継がれるよう修正されるが，そのまま放置されてしまうこともある．

5. 訓練データの作成

6 章で述べる不適切リンクの修正に用いるランキング学習モデルのためのデータセットについて述べる．

5.1. 不適切リンクの疑似生成

膨大な記事数とリンク数が存在する Wikipedia において不適切リンクとその修正先のサンプルを収集するのは容易なことではない．ただし 5 章で述べた通り，不適切リンクが生まれる原因には単語の多義性が関係している．そこで各用法がまとめられた曖昧さ回避ページをグラフ型データベースで探索し不適切リンクを擬似的に生成する手順について説明する．

はじめに，1 つ以上の記事からリンクされている曖昧さ回避のページを収集し，多義性を持つ単語の各用法についての記事を取得する．ある記事が曖昧さ回避ページかどうかの判断はカテゴリ「曖昧さ回避」に属しているかどうかで判断することができる．次に，集めた曖昧さ回避ページの中に含まれる記事（リンク）を各用法として抽出する．このとき，抽出するリンクを次の 3 つの条件を満たすものに限っている．

① リストの項目の 1 つになっている
② 関連項目の中にあるリンクではない
③ リンク先が曖昧さ回避ページではない

これは曖昧さ回避のページでは各用法を箇条書きで一覧にすることが多いためである．ここまでの処理を行う Neo4j へのクエリは以下のようになる．

```
MATCH (a:Article)-[:CATEGORY_IN]->(b:Category)
WHERE b.title = "曖昧さ回避"
WITH a, b
MATCH (a)-[l:LINK_TO]->(c:Article)
WHERE l.isItem AND NOT (c)-[:CATEGORY_IN]->(b)
      AND NOT l.isInSeeAlso
RETURN a, b, collect(c)
```

変数 a には曖昧さ回避ページが，変数 b にはカテゴリ「曖昧さ回避」が，変数 c には用法別の記事がそれぞれ代入される．曖昧さ回避ページ「アーケード」について同様のクエリを実行すると図 3 の結果が返る．

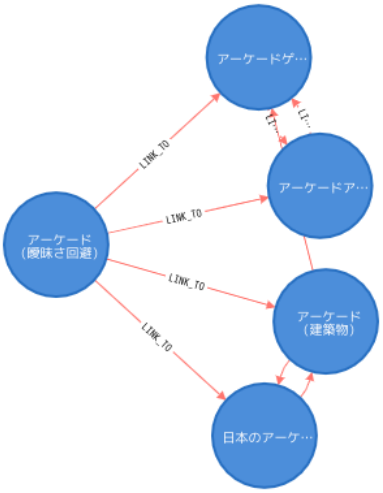


図 3. アーケードの各用法

次に抽出した各用法の記事それぞれに対して被リンクを取得する．ここで，実際に存在する各用法の記事へのリンクは適切なリンクと見なす．一方，不適切リンクはその適切なリンクの参照先だけを別の用法の記事へと書き換えることで生成する．

5.2. 特徴量の作成

機械学習モデルへの入力としてリンク 1 件に対し以下の 7 つの特徴量を作成した．

- **リンク先記事名とアンカーテキストの類似度(w1)**

  リンク先記事名とアンカーテキストの類似度を Python3 の difflib.SequenceMatcher で計算する．
- **アンカーテキストに対するリンク先記事の出現確率(w2)**

  特徴量を計算するリンクが持つアンカーテキストと同じアンカーテキストを持つリンクを全文検索で抽出し，その宛先に対するリンク先記事の出現確率を計算する．
- **リンク先記事の被リンクにおけるアンカーテキストの出現確率(w3)**

  リンク先記事の被リンクを取得し，そのアンカーテキストとして特徴量を作成するリンクが持つアンカーテキストが使われている割合を計算する．
- **リンク元記事とリンク先記事間のカテゴリリンク構造における最短距離(w4)**

  リンク元記事とリンク先記事間のカテゴリリンク構造において最短経路探索を行いその最短距離を計算する．ただし計算コストを考慮し，距離が 4 以上のものは探索をやめて全て 6 としている．
- **リンク先記事の被リンク数(w5)**

  リンク先記事の質が反映されることを期待して特徴量に組み込んだ．
- **リンク元記事からリンク先記事への記事間リンク構造における 2 ホップ以内のパス数(w6)**
- **リンク元記事とリンク先記事のどちらにもリンクを作成している記事数(w7)**

  どちらも 2 つの記事同士の関連の強さが反映されることを期待して特徴量に組み込んだ．

5.3. 不適切リンクの修正

不適切なリンクを修正するためには，複数の正しいリンク先記事の候補の中から最適な記事を選ぶ必要があり通常の分類問題や回帰問題とは性質が異なる．複数の候補をリンク先としてふさわしい順に並び替え，利用者に最適な修正案を提案するためにランキング学習を用いる．使用するモデルには勾配ブースティングベースのペアワイズ手法によるランキング学習モデル LambdaMART[10]を選び，実際の学習にはそれを実装した Microsoft が公開する LightGBM[11]を用いた．LightGBM は短い学習時間で高い精度が期待でき，大規模なデータセットにも対応できる．

未知の不適切リンクの修正は次の手順で行う．まず，不適切リンクとそのリンク先記事を候補記事に書き換えたものについて疑似訓練データと同様に特徴量を計算する．次に，作成されたデータをランキング学習モデルに入力し各リンクについてモデルによるスコア値を求める．最後にそのスコア値が最も高い候補記事を修正案とし，利用者に提示を行う．なお，最も高いスコア値の記事がもとのリンク先記事であった場合はそのリンクは不適切ではなかったと判断する．

6. 評価実験

提案手法を評価するために，実際に Wikipedia 日本語版のリンク構造が格納されたグラフ型データベースを用いて疑似訓練データの作成を行い，ランキング学習モデルを学習させる．そして同様に作成した評価用データについて学習済みモデルによる不適切リンクの修正を行い，その精度を評価する．

6.1. 実験内容

まず，疑似訓練データの作成を行う．実験に用いたデータベースには 1 つ以上の記事からリンクされている曖昧さ回避ページは 31,696 件存在するが，その内の無作為に選んだ 2 万件をもとに実際に疑似訓練データを作成する．その際，生成できる擬似的な不適切リンクの数は適切なリンクの数よりも遥かに多いので，適切なリンク 1 件につき不適切リンクは最大 3 件のみ生成するようにした．このとき，最終的なデータ総量を調整するために 1 つの用法別の記事に対して適切なリンクは最大 5 件までとしている．こうして作成された

疑似訓練データでランキング学習モデルを学習させる．

次に，実際の不適切リンクについてリンク先記事の修正を行う．修正する不適切リンクは，疑似訓練データの作成に使わなかった残りの曖昧さ回避ページへのリンクの中から無作為に選んだ 100 件とし，リンク先候補記事はその曖昧さ回避ページに存在する全てのリンクのリンク先記事とする．疑似訓練データと同様にこの評価用データについても特徴量を計算し，学習済みのランキング学習モデルを用いて不適切リンクの修正を行う．

最終的に正しい修正案の提示が行えたかを Wikipedia 上の実際の記事を見て判断し，その割合を算出した．

6.2. 実験環境

実験は表 2 の環境で行った．また，データベースのデータは表 1 のものを使用した．

表 2. 実験環境

項目	値
OS	CentOS 7.6
CPU	Core i7-3770
メモリ	24GB
ストレージ	SSD 128GB
データベース	Neo4j Community Edition 3.5.2

6.3. 結果と考察

作成された疑似訓練データは 762,939 件あり，ランキング学習モデルで学習させる際には学習データとテストデータが 8:2 の割合になるよう分割した．このデータを使い，ランキング学習モデルを学習した際の各特徴量の重要度は図 4 のようになった．

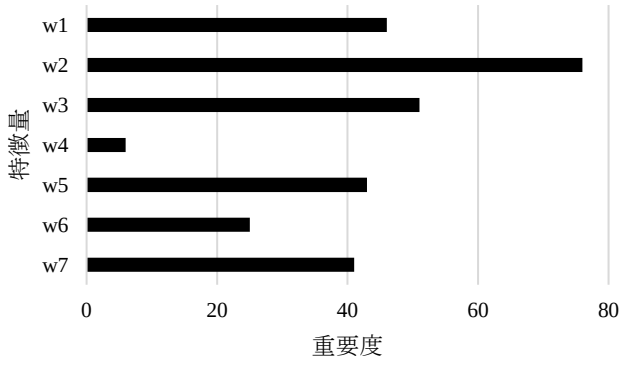


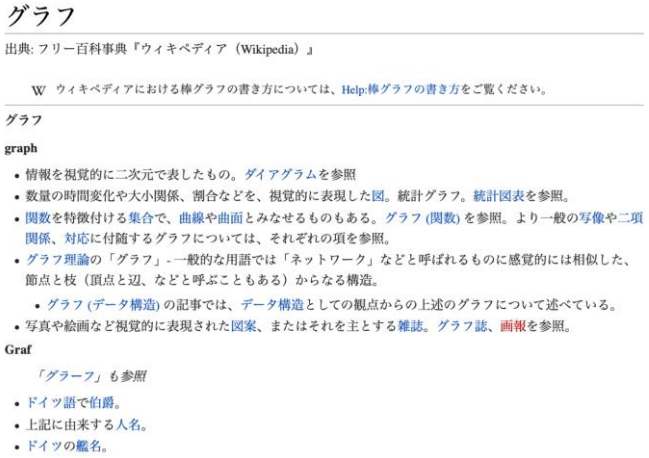
図 4. 各特徴量の重要度

不適切リンクの修正精度については疑似訓練データ内のテストデータでは 98.9%と非常に高かった．実際の不適切リンク 100 件に対して評価した結果は表 3 の通りである．

表 3. 実験結果

評価リンク数	正しく修正できた数	修正成功率
100	60	60%

疑似訓練データ内のテストデータにおける修正成功率と比べ，実際の不適切リンク 100 件において正しく修正できた割合は 60%と大きく下回った．図 5 のような用法別の記事について，その意味が言葉で説明がされている曖昧さ回避ページの場合，本研究の手法では各用法別の記事が正しく抽出されず生成される不適切リンクの実態は実際のものとかけ離れてしまう．また，訓練データと実際のデータについて適切なリンクと不適切なリンクの分布も異なる．これらが修正精度に悪影響を与えた原因のひとつとして考えられる．


グラフ

出典: フリー百科事典『ウィキペディア（Wikipedia）』

ウィキペディアにおける棒グラフの書き方については、Help:棒グラフの書き方をご覧ください。

グラフ

graph

- 情報を視覚的に二次元で表したもの。ダイアグラムを参照
- 数量の時間変化や大小関係、割合などを、視覚的に表現した図。統計グラフ。統計図表を参照。
- 関数を特徴付ける集合で、曲線や曲面とみなせるものもある。グラフ (関数) を参照。より一般の写像や二項関係、対応に付随するグラフについては、それぞれの項を参照。
- グラフ理論の「グラフ」- 一般的な用語では「ネットワーク」などと呼ばれるものに感覚的には相似した、節点と枝（頂点と辺、などと呼ぶこともある）からなる構造。
  - グラフ (データ構造) の記事では、データ構造としての観点からの上述のグラフについて述べている。
- 写真や絵画など視覚的に表現された図案、またはそれを主とする雑誌。グラフ誌、画報を参照。

Graf

「グラーフ」も参照

- ドイツ語で伯爵。
- 上記に由来する人名。
- ドイツの艦名。

図 5. 用法別の記事が上手く抽出できない例

ただし，修正を試みた実際の不適切リンク 100 件の中には Wikipedia 日本語版に適切なリンク先記事が存在しないものが 18 件含まれた．適切なリンク先記事が存在する場合に限った場合の修正成功率は 73.1%となり疑似訓練データであっても一定の有効性があると言える．このような適切なリンク先記事が存在しない場合，修正しない判断をすることや他の情報源から目的の情報を探すなどの判断を行うことが望ましい．そこで正しく修正できた場合と間違えた場合，適切なリンク先記事が存在しない場合でモデルが出力するスコアの最高値について調べた結果が表 4 である．

表 4. 各ケースにおける最高スコアの平均

ケース	平均最高スコア
正しく修正できた場合	0.25
修正案が不適切な場合	-0.34
適切なリンク先記事がない場合	-0.4

正しく修正が行えなかった場合や適切なリンク先記事が Wikipedia に存在しない場合には全てのリンク先候補記事においてスコアが低い傾向が見られた. 適切な閾値を設定することで上述したような処理の切り替えが可能であると考えられる.

7. まとめ

本論文では Wikipedia のリンク構造を格納したグラフ型データベースを用いて不適切リンクの疑似生成と特徴量の作成を行い, その疑似訓練データで学習させたランキング学習モデルによって不適切リンクの修正を行う手法を提案した. 評価実験により疑似訓練データの有用性を示したが, さらなる精度向上のためにより実態に即した不適切リンク生成手法の検討や特徴量の吟味を行うことが今後の課題として挙げられる.

参 考 文 献

[1] Wikipedia: 全言語版の統計 - Wikipedia, https://ja.wikipedia.org/wiki/Wikipedia:全言語版の統計

[2] 鈴木優, “Wikipedia における情報の質”, 情報処理学会論文誌データベース Vol6, No4, pp. 46-58, 2013.

[3] Giles J. “Internet encyclopaedias go head to head”, Nature 438, pp. 900-901, 2005.

[4] D. Milne, “Computing semantic relatedness using wikipedia link structure”, Proc. of the New Zealand Computer Science Research Student Conference, pp. 1-8, 2007.

[5] 大政飛鳥, 井上潮, “グラフ型データベースを用いた Wikipedia からの関連概念抽出手法”, 第 11 回データ工学と情報マネジメントに関するフォーラム（DEIM Forum 2019）, F5-4, 2019.

[6] B. M. Pateman and C. Johnson, “Using the Wikipedia link structure to correct the Wikipedia link structure”, In Proc of 2nd Workshop on Collaboratively Constructed Semantic Resources, pp. 10-18, 2010.

[7] C. Wang, R. Zhang, X. He, and A. Zhou, “Error Link Detection and Correction in Wikipedia”, In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM ’16), pp. 307–316, 2016.

[8] Neo4j, https://neo4j.com/

[9] Wikipedia:曖昧さ回避,https://ja.wikipedia.org/wiki/Wikipedia:曖昧さ回避

[10] C. Burges, “From ranknet to lambdarank to lambdamart: An overview”, Microsoft Research Technical Report MSR-TR-2010-82, 2010.

[11] LightGBM, https://github.com/microsoft/LightGBM