ウェブ検索クエリに対する教師なしエンティティリンキング

齋藤 智輝　豊田 樹生　夜久 真也　岩澤 宏希
ヤフー株式会社
{tosaito, itoyota, syaku, hiwasawa}@yahoo-corp.jp

1 はじめに

近年, 商用検索エンジンの進化によってユーザが検索結果に求める情報は高度化している. 例えば, 検索結果に表示されるリンクを辿り目的のウェブページに遷移する前に, 図 1のようにウェブ検索クエリ (以下, クエリ) からユーザの意図するエンティティ(実存する概念) を解釈し, そのエンティティの構造化された情報を表示することでユーザの満足度を満たすことが知られている.[1] そのため, ウェブページからの情報抽出やユーザの求めている情報のさらなる理解を目指し, ウェブ上のブログやニュース記事などの自然文に表れるエンティティを表す固有表現を知識ベース (Knowledge Base, KB) 上のレコードと結びつけるエンティティリンキングタスクに関する研究が広く行われている.

図 1: クエリ「増上寺」に対して掲載するエンティティパネル

そこで, ユーザの満足度を満たすために, 入力されたクエリに対してエンティティリンキングタスクを解き, エンティティパネルに表示するエンティティを特定する必要がある. しかし, ウェブページやニュース記事内でのエンティティリンキングタスクとは異なり, ウェブ検索クエリは文としては短い. そのため, エンティティを特定するための文脈情報などが取得しにくいことや, 入力ミス, 打ち間違いなど, 既存の手法を適用する際に様々な問題について考える必要がある.

さらに, エンティティリンキングを行う際に対象となる固有表現が多義語である場合, 複数のエンティティが候補として挙げられる. (例えば Wikipedia のページタイトルにおいて, 「さくら」という固有表現に該当するページは植物や曲, 新幹線など 10 個以上のページが存在している.)

多義語に対する曖昧性解消問題について, ERD'14[2] をはじめとするコンテストや, どのエンティティが最もその固有表現に結びつくべきなのかについての研究も行われている. しかし, 検索エンジンへの入力クエリに対する適用, さらに日本語を扱う研究となるとその数も極端に少なくなる.

本研究では, 日本語を含むクエリに対するエンティティリンキングタスクについて考える. その中でも特に多義クエリに対する曖昧性解消問題について取り組む. 具体的には, Wikipedia 内のアンカーテキスト情報及びウェブ検索ログから「あるクエリがどのエンティティの意図として入力されやすいか」という統計量を算出し, 2 つの統計量から混合モデルを構築することで多義クエリの曖昧性を解消する. また, Wikipedia の記事本文及び BM25 を用いて曖昧性を解消する手法をベースラインとし, 混合モデルによる提案手法を商用検索エンジンに実際に入力されたクエリを用いて評価・比較することで, 提案する手法の優位性を示す.

2 関連研究

特定の固有表現に対する曖昧性解消問題に取り組んでいる研究について, クエリを対象としている研究, 日本語を扱っている研究をそれぞれ紹介する.

クエリを対象にしている研究として, Roi ら [3] は, 入力クエリ q とエンティティ e のペアに対するスコア $P(\mathbf{e}|\mathbf{q})$ を算出して曖昧性解消問題に取り組んでいる. ウェブ検索ログと Wikipedia の内部リンクの向き先から, クエリ q に対して最もスコアを高くするようなエンティティ $\mathbf{e}$ を特定している.

日本語を扱っている研究として, 石川ら [4] は, ニュース記事を対象に文中に表れる固有表現に対して曖昧性解消を行っている. Wikipedia の「曖昧さ回避ページ」のタイトルになっている表記 s_t を曖昧さを持つ多義語とし, それらの表記に対するエンティティ e_t を特定する. その際, Wikipedia の内部リンクとウェブ検索ログを用いた統計量, さらに, 入力テキストから得られる文脈情報 c_t を用いてスコア $Score(e_{ti}|s_t, c_t)$ を算出し, 曖昧性解消を行っている.

他にも, Wikipedia を知識ベースとして, 文中の固有表現に対して Wikipedia のレコード (エンティティ) を割りあてる wikification のツールも松田ら [5] によって作成されている.

以上で, 多義語表記の曖昧性解消問題に取り組む 2 つの研究を挙げた. 本研究では, 日本語のウェブ検索クエリを対象に曖昧性解消問題に取り組む.

3 曖昧性解消混合モデル

本研究では, Roi ら [3] と同様に Wikipedia の内部リンクとウェブ検索ログから統計量を計算し, それらの混合モデルとして曖昧性解消混合モデルを作成した. 本章では, Wikipedia とウェブ検索ログの 2 つの情報ソース (c_w, c_q) から統計量の算出, 算出した統計量から曖昧性解消混合モデルを作成する手法について述べる.

曖昧性解消混合モデルにおいて, 曖昧性解消の対象となる表記 s が与えられたときに, s がエンティティ e を意図する確率 $P(e|s)$ は Roi ら [3] の式 (1)-式 (6) で定義される. ここで, a_s はある表記 s がエンティティへのエイリアスかどうか ($a_s = 0$: テキスト, $a_s = 1$: エイリアス), $n(s, c)$, $n(e, c)$ はそれぞれ, あるソース c における表記 s が生起する回数, あるソース c においてエンティティ e が生起する回数を表す.

$$P(a_s = 0|c, s) = 1 - P(a_s = 1|c, s) \quad (1)$$

$$P(a_s = 1|c, s) = \frac{\sum_{s:a_s=1} n(s, c)}{n(s, c)} \quad (2)$$

$$P(e|c) = \frac{n(e, c) + 1}{|E| + \sum_{e \in E} n(e, c)} \quad (3)$$

$$P(e|a_s, c, s) = \frac{\sum_{s:a_s=1} n(s, c) + \mu_c \cdot P(e|c)}{\mu_c + \sum_{s:a_s=1} n(s, c)} \quad (4)$$

$$P(e|c, s) = \sum_{a_s=\{0,1\}} P(a_s|c, s) P(e|a_s, c, s) \quad (5)$$

$$P(e|s) = \sum_{c \in \{c_q, c_w\}} P(c|s) P(e|c, s) \quad (6)$$

3.1 Wikipedia の内部リンク

Wikipedia の文中では, ある固有表現に対応するページが Wikipedia 内に存在する場合, その固有表現を該当ページへのリンク (アンカーテキスト) として記述できる. 例として, 「バリスタ」という固有表現について, あるページでは「バリスタ (職業)」, 他のページでは「バリスタ (電子部品)」への内部リンクが記述されている. そこで, Wikipedia の文中において表記 s がアンカーテキストとして登場するときを $a_s = 1$, それ以外を $a_s = 0$ として, Wikipedia の記事が表すエンティティ e に対する確率 $P(e|c_w, s)$ を計算する.

3.2 ウェブ検索ログ

本研究でのウェブ検索ログとは, ユーザが検索エンジンに入力したクエリ q と検索結果に表示されるページのうち, ユーザが実際に遷移した Wikipedia URL(url) のペア (q, url) を指す. 例えば, クエリ「関ヶ原」について, 「関ヶ原の戦い」や「関ヶ原 (映画)」, 「関ヶ原 (小説)」など, 多義語クエリに対しては, 同じ表記が異なる Wikipedia のページに遷移することが考えられる. この情報から, クエリ表記 s がエンティティ e を示す確率 $P(e|c_q, s)$ を計算する.

3.3 情報ソースの信頼度

本研究では, 混合モデルを構築する際に各情報ソースの信頼度として以下に示す確率 $P(c|s)$ を用いる.

あるソース c において表記 s が生起する回数を $n(s, c)$ としたとき, 表記 s が情報ソース c で生起する確率 $P(c|s)$ は式 (7) のように表せる.

表 1: 使用したデータと取得時期

入力データ	データ取得時期
クエリログ	2017/12/1-12/19
ウェブ検索ログ	2015/10-2016/09
Wikipedia ダンプファイル	2017/11/20
Wikidata ダンプファイル	2017/11/20

$$P(c|s) = \frac{n(s,c)}{\sum_{c'} n(s,c')} \tag{7}$$

以上のように, 各情報ソース内でそれらの表記が生起する頻度から算出した確率を用いて, 曖昧さ解消の混合モデルを構築する. また, 混合モデルを構築する際, 各確率値に対してはラプラス法によりスムージングを行うこととする.

4 実験

本研究では, 知識ベースとして Wikidata[1] の持つエンティティのうち, 日本語 Wikipedia ページを持つものから構築したものを用いる.[2] その知識ベース及び Yahoo!検索に入力されたクエリと検索数のペアからなるクエリログを用いて評価用クエリセットを作成する.

作成した評価用クエリセットに対して各手法で曖昧性解消を行い, 評価指標としては, ウェブ検索への適用を想定して, 各手法毎に最も類似度が高いエンティティ e とクエリのペア (e, q) に対する精度 Precision at 1(P@1) を用いる.

本研究で使用したデータ及び取得時期を表 1に示す.

4.1 評価用クエリセットの作成

対象とするクエリは, 取得したクエリログのうち, タブやスペースなどの空白文字で分割した際の単語数が 1 のクエリを対象とした. また, 曖昧性解消を行う必要のある多義語クエリを抽出するため, 1で示したクエリログのクエリ表記と知識ベース内のエンティティが持つ複数の日本語表記 (Japanese Label と Also Known as) を完全一致させ, マッチしたエンティティが複数個あるものを対象とした. なお, Wikipedia の曖昧さ回避ページからなるエンティティを正解としてしまうことを避けるため, Wikimedia disambiguation page のインスタンス[3] であるか否かの情報を基にそのようなエンティティはあらかじめ除外した.

以上で抽出したクエリについて, 検索数が上位 100 件の Head クエリと, 検索数の下位 100 件の Tail クエリについて, ERD'14[2] の Annotation Guideline に準じて人手でアノテーションを行い評価用クエリセットを作成した.

[1] https://www.wikidata.org

[2] この知識ベースは著者らの所属団体で運用されている非公開のものである

表 2: P@1 での評価結果

評価手法	P@1	
	Head	Tail
BM25	0.70	0.71
提案手法	0.87	0.79

4.2 比較手法

本研究では, Wikipedia の記事本文を単一の情報ソースとして, 教師なしの手法において広く使われている BM25[6, 7] を用いて曖昧性を解消する手法をベースラインとして用意する.

具体的には, 以下の式 (8), (9) を用いて, 対象クエリ q と Wikipedia ページ D の組み合わせに対して算出された類似度スコア $Score(D, q)$ が最も高いペア (D, q) を選出, 選出されたドキュメントが表すエンティティ e とクエリのペア (e, q) を曖昧性解消の最終出力とする. ここで, N,$n(q)$, $f(q, D)$,$avgdl$ はそれぞれ Wikipedia 全体のページ数, クエリ q が生起する Wikipedia ページの数, Wikipedia ページ D 内でのクエリ q の生起回数, 平均ページ長とし, ハイパーパラメータの k, b はそれぞれ 1.2, 0.75 とした. また, Wikipedia の記事本文を入力として用いる際にトークナイズ等は特に行わない.

$$IDF(q) \quad = \quad log\frac{N - n(q) + 0.5}{n(q) + 0.5} \tag{8}$$

$$Score(D, q) = IDF(q) \cdot \frac{f(q, D) \cdot (k + 1)}{f(q, D) + k \cdot (1 - b + b \cdot \frac{|D|}{avgdl})} \tag{9}$$

4.3 結果・考察

各手法を P@1 で比較した結果を表 2に示す.

[3] https://www.wikidata.org/wiki/Q4167410

表 3: 提案手法による改善例

クエリ	Wikipedia のタイトル	
	BM25	提案手法
アンドロイド	人造人間	Android
モーニング娘。	モーニング娘。の歴史	モーニング娘。
ニット帽	帽子	ニット帽

表 4: 提案手法による改悪例

クエリ	Wikipedia のタイトル	
	BM25	提案手法
いかけ屋	いかけ屋	鋳掛屋
コチャン郡	コチャン郡	居昌郡

表より, Head クエリと Tail クエリ共に提案手法の P@1 の値が比較手法よりも上回っていることが確認できる. また, 表 3に提案手法によって改善した例を示す. クエリに対してどのエンティティがリンキングされたのかをクエリとエンティティの持つ Wikipedia URL のタイトルで表現しているが, いずれもクエリが意図していると考えられるエンティティにリンキングされている. 例えば, 「モーニング娘。」というクエリに対して, アイドルグループ自体を指すエンティティとそのグループの歴史を示すエンティティが存在する場合に提案手法では前者をリンキングすることが出来ている.

表 4に提案手法によって改悪してしまった例を示す.

ここで挙げた例では, クエリ「いかけ屋」に対して正解とは異なる「鋳掛屋」を表すエンティティにリンキングしてしまっている. 表 4のクエリはいずれも Tail クエリであり, Wikipedia の内部リンクにおいても今回正解とアノテーションされたエンティティとは異なるエンティティに結びついている回数が多かった. そのため, Wikipedia の内部リンクの影響を改善できるだけのウェブ検索ログ側の統計量が十分に得られず, 最終出力も正解とアノテーションされた結果と異なってしまった.

5 おわりに

本研究では, 日本語を含むクエリを対象に, エンティティリンキングタスクでの曖昧性の解消に取り組んだ.

Wikipedia の内部リンク及びウェブ検索ログから構築した混合モデルを用いた手法を提案し, これをよりシンプルな Wikipedia の記事本文及び BM25 を用いた曖昧性解消手法と比較することで, 提案手法が高い精度を実現することを示した.

本研究では, 単語数が 1 つからなる文として非常に短いクエリを評価対象としたが, 検索数の上位と下位の 100 件ずつの小規模なものであった. 今後, より広い範囲のクエリに対しても提案手法が有効であるか, クラウドソーシングなどにより作成した大規模な評価クエリを用いることで検証を行いたい. また, 本手法は単語を 2 つ以上含むクエリにおいて, 対象となる単語の周辺単語を利用できていない. 今後はこの周辺単語を利用したクエリの曖昧性解消の手法についても検討していく.

参考文献

[1] Horatiu Bota, Ke Zhou, and Joemon M. Jose. Playing Your Cards Right: The Effect of Entity Cards on Search Behaviour and Workload. In *Proceedings of the 2016 ACM on Conference on Human Information Interaction and Retrieval*, CHIIR '16, pp. 131–140, New York, NY, USA, 2016. ACM.

[2] David Carmel, Ming-Wei Chang, Evgeniy Gabrilovich, Bo-June Paul Hsu, and Kuansan Wang. ERD'14: Entity recognition and disambiguation challenge. In *ACM SIGIR Forum*, Vol. 48, pp. 63–77. ACM, 2014.

[3] Roi Blanco, Giuseppe Ottaviano, and Edgar Meij. Fast and Space-Efficient Entity Linking for Queries. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, WSDM '15, pp. 179–188, New York, NY, USA, 2015. ACM.

[4] 石川裕貴, 小林健, 長田誠也. ウェブ検索ログと Wikipedia 内部リンクを用いたエンティティの曖昧性解消. 言語処理学会第 21 回年次大会, pp. 696–699, 2015.

[5] 松田耕史, 岡崎直観, 乾健太郎. 日本語 wikification ツールキット: jawikify. 言語処理学会第 23 回年次大会, pp. 250–253, 2017.

[6] Stephen Robertson, Hugo Zaragoza, et al. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, Vol. 3, No. 4, pp. 333–389, 2009.

[7] Hasibi Faegheh, Nikolaev Fedor, Xiong Chenyan, Balog Krisztian, ErikBratsberg Svein, Kotov Alexander, and Callan Jamie. DBpedia-Entity v2: A Test Collection for Entity Search. In *ACM SIGIR Forum*. ACM, 2017.