

# 潜在関係検索の対称性および相補語の期待性を用いたスコアに基づくランキング法

Ranking Method based on Symmetry and Complementary Words of Latent Relational Search

後藤 友和\*<sup>1</sup>  
Tomokazu Goto

Nguyen Tuan Duc\*<sup>1</sup>

Danushka Bollegala\*<sup>1</sup>

石塚 満\*<sup>1</sup>  
Mitsuru Ishizuka

\*<sup>1</sup> 東京大学大学院 情報理工学系研究科

Graduate School of Information Science and Technology, The University of Tokyo

Relational search is a novel paradigm of search which focuses on the similarity between semantic relations. Given three words ( $A$ ,  $B$ ,  $C$ ) as the query, a relational search engine retrieves a ranked list of words  $\mathcal{D}$ , where a word  $D \in \mathcal{D}$  is assigned a high rank if the relation between  $A$  and  $B$  is highly similar to that between  $C$  and  $D$ . However, if  $C$  and  $D$  has numerous co-occurrences, then  $D$  is retrieved by existing relational search engines irrespective of the relation between  $A$  and  $B$ . To overcome this problem, we exploit the symmetry in relational similarity to rank the result set  $\mathcal{D}$ . To evaluate the proposed ranking method, we use a benchmark dataset of Scholastic Aptitude Test (SAT) word analogy questions. Our experiments show that the proposed ranking method improves the accuracy in answering SAT word analogy questions, thereby demonstrating its usefulness in practical applications.

## 1. はじめに

検索エンジンを用いて情報を探るとき、目的の情報を得るためには検索エンジン利用者の知識を必要とすることが多い。例えば、検索エンジンの方式がクエリマッチングの場合、利用者は、探したい情報が載っているページが含むような語をクエリとして与えなければならない。しかし、探したい情報が曖昧である場合や適切なクエリが思いつかない場合、目的の情報を得ることができない。

Relational Search と呼ばれる検索システムは、関係を明示する必要なく、目的の情報を得ることができるシステムである。Relational Search は 3 つの入力クエリ ( $A, B, C$ ) を持ち、 $A$  と  $B$  が持つ関係と、 $C$  と  $D$  が持つ関係が等しくなるような  $D$  を結果として返す。このクエリを、 $(A, B), (C, ?)$  と表すことにする。例えば、Relational Search に対してクエリ  $\{(ostrich, bird), (lion, ?)\}$  を入力したとき、結果として  $cat$  という語が期待できる。何故なら、 $”?"$ に  $cat$  という語を代入した場合、2 つのワードペア間の関係は「 $X$  is a large  $Y$ 」となるからである ( $X$  と  $Y$  は任意の語)。つまり、それぞれ「 $ostrich$  is a large  $bird$ 」、「 $lion$  is a large  $cat$ 」という関係が成り立っている。このように Relational Search は、 $”?"$ に語を入れたときに、与えられたワードペア間での関係が似る語を返すシステムである。

後藤ら [後藤 10] は、SAT データセットにより、Relational Search の精度を測った。このとき、対称性は 2 つしか用いていない。また、候補語のスコアリングに関しては、ランキング内の順位を用いているため、Relational Search による正解が複数含まれている場合には、SAT での正解であっても必然的に順位が下がり、結果が悪くなるということがあった。本研究では、使用する対称性を 6 つに増やし、対称性を増やすことの影響を見る。また、スコアリングに関してはそのままスコアを用いることで精度のさらなる向上を目指す。

## 2. 手法

### 2.1 Relational Search の実装

Relational Search システムを実装するためには、与えられたワードペア間の関係を取得する必要がある。そこで、我々はワードペア間の関係を得るために Web 検索エンジン\*<sup>1</sup>を用いた。ここでは、Web 検索エンジンを用いた Relational Search の実装方法について述べる。

Relational Search のクエリを  $\{(A, B), (C, ?)\}$  とする。我々は  $A, B$  間の関係を抽出するために、検索結果のスニペットを用いた。スニペットとは、検索で得られた Web 文書内の、検索クエリが含まれている箇所を提示する短い文章である。我々は  $A$  と  $B$  を含むスニペットを得るために検索エンジンに対し “ $A * * B$ ” というクエリを投げた (“\*” も含む)。\* は 1 つ、もしくは 0 の単語にマッチする。このクエリでは  $A$  と  $B$  の間に 3 単語以内の単語列を含んだスニペットを得ることができる。スニペット内で、 $A$  と  $B$  に囲まれる単語列 (パターン) の集合を  $\mathfrak{P}_{A, B}$  とする。 $A, B$  およびパターン  $P_{A, B}$  ( $P_{A, B} \in \mathfrak{P}_{A, B}$ ) を含む文字列 “ $A P_{A, B} B$ ” に対して、 $A$  の部分を  $C$  に置き換え、 $B$  の部分を \* に置き換えた、“ $C P_{A, B} *$ ” を新たなクエリとする。そして、このクエリを再び検索エンジンに投げて再度スニペットを取得する。このスニペットから、\* に相当する語を抽出して得られた語の集合を検索結果の候補語集合と呼ぶ。この候補語集合  $\mathcal{D}$  に含まれる候補語  $D_i$  ( $D_i \in \mathcal{D}$ ) に対し、式 1 によるスコア付けを行い、降順に並べることでランキングを作成する。

$$\text{score}(D_i) = \frac{\sum_{P_{C, D_i} \subseteq \mathfrak{P}_{C, D_i}} (\text{freq}("C P_{C, D_i} D_i"))}{\text{freq}("C * * D_i")}. \quad (1)$$

ここで、 $\mathfrak{P}_{C, D_i}$  は  $C, D_i$  と共に現れたパターンの集合であり ( $\mathfrak{P}_{C, D_i} \in \mathfrak{P}_{A, B}$ )、 $\text{freq}(query)$  はクエリ文字列  $query$  を投げたときの検索ヒット件数である。式 1 は、 $C$  と  $D_i$  と  $P_{C, D_i}$  の結びつきの強さを表している。

図 1 は Relational Search の実行の一連の流れの例を示している。 $\{(ostrich, bird), (lion, ?)\}$  という Relational Search のクエリに対し、“ $ostrich * * bird$ ” というクエリを検索エンジンに投

連絡先: 後藤友和, 東京大学大学院情報理工学系研究科, 東京都文京区本郷 7-3-1, goto@mi.ci.i.u-tokyo.ac.jp

\*<sup>1</sup> <http://developer.yahoo.com/search/boss/>

げる．そこから得られたスニペット，“the ostrich is the largest bird in size and weight on earth...”から，ostrich と bird に囲まれた単語列である “is the largest” を得る．そして，この単語列と {(lion, ?)} を組み合わせ，“lion is the largest \*” というクエリを作成し，再び検索エンジンに投げる．そこから得られるスニペット，“the lion is the largest cat, while some...” から\*に相当する語を得ることで候補語 cat を得ることができる．このとき，lion, cat と共に現れたパターンは “is the largest” である．なお，例には登場しないが，他にも “is the big” などパターンとして出現することがある．このとき，式 1 では， $\mathfrak{P}_{lion,cat} = \{“is the largest”, “is the big”\}$  である．

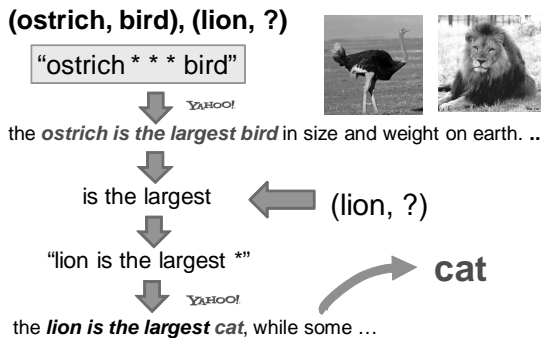


図 1: クエリ {(ostrich, bird),(lion, ?)} に対する Relational Search の実行例

## 2.2 相補語スコアと対称性

式 1 では，検索エンジンのヒット件数を用いている．しかし，検索エンジンのヒット件数は正確な値でない場合もあり [FUNAHASHI 08, FUNAHASHI 09]，特にヒット件数が多いときは下数桁が丸められ，それが顕著に見られる．また，単純に，強く共起する語に対してはスコアが高く出てしまうこともある [Bunescu 07]．これらの問題を解決するために，相補語のスコアと対称性を用いる．

Relational Search のシステムに対し，何らかのクエリ， $\{(A, B), (C, ?)\}$  を投げ，得られた語  $D_i$  を “?” の位置に代入し，他の  $A, B$  または  $C$  を “?” に置き換える．このとき，“?” に置き換わった語を “?” の相補語と呼ぶ．また，“?” に置き換えた後にそのクエリを再び Relational Search システムに対して投げたときに得られるスコアを “?” の相補語スコアと呼ぶ．今回扱うのは “?” に関してのみなので，“?” の相補語，および “?” の相補語スコアのことをそれぞれ単に相補語，相補語スコアと呼ぶことにする．例えば， $\{(A, B), (C, ?)\}$  というクエリにおいて，相補語を  $C$  とすると，相補語スコアを求めるためのクエリは  $\{(A, B), (? , D_i)\}$  となる．このとき，相補語スコアは， $C$  のスコアになる．

$\{(A, B), (C, ?)\}$  から得られる語と  $\{(B, A), (? , C)\}$  から得られる語は共通であることが期待できる．例えば， $\{(ostrich, bird), (lion, ?)\}$  というクエリにおいて，ostrich と bird の関係を表す単語列として，“ostrich is a bird” などがある．逆に， $\{(bird, ostrich), (? , lion)\}$  としたとき，bird と ostrich の関係を表す単語列として，“bird such as ostrich” などがある．このとき，どちらも  $\{(ostrich, bird), (lion, ?)\}$  から cat を導くことができる．これが Relational Search の対称性である．

$\{(A, B), (C, ?)\}$  から， $D_i \in \mathcal{D}$  が得られたとき，対称性を用いることで，表 1 に示す 6 種類の相補語スコアを用いることが

できる．これら 6 種類の相補語スコアを平均化し，最初に得られた  $D_i$  のスコアとの和，または積を取ることで，最終的なスコアとする．あるクエリ文字列 query の相補語スコアを， $comp(query)$  と定義すると，最終的なスコア (final\_score) は式 2 で表せる．

$$final\_score(D_i) = score(D_i) (+ or *) \frac{\sum_{q \subseteq Q} (comp(q))}{len(Q)} \quad (2)$$

ここで， $Q$  は，表 1 に記した 6 種類のクエリ文字列の集合であり， $len(Q)$  は， $Q$  が含むクエリ文字列の個数を表し， $score(D_i)$  は式 1 で示した  $D_i$  のスコアである．

表 1: 相補語スコアと対称性

相補語スコアを求める時のクエリ	求める相補語スコア
$\{(A, B), (? , D_i)\}$	$C$ のスコア
$\{(B, A), (D_i, ?)\}$	$C$ のスコア
$\{(C, D_i), (A, ?)\}$	$B$ のスコア
$\{(C, D_i), (? , B)\}$	$A$ のスコア
$\{(D_i, C), (? , A)\}$	$B$ のスコア
$\{(D_i, C), (B, ?)\}$	$A$ のスコア

## 3. 実験

本手法の精度を測るために，本研究では Scholastic Aptitude Test (SAT) データセットを用いた．SAT データセットは SAT の問題のうち，アナロジー問題のみを 374 件集めたものである．Bollegala ら [Bollegala 09] や Turney [Turney 06] の手法においても関係類似度を測るためのベンチマークとして使われてきた．正解率だけを見れば良いため，容易に手法の比較ができる．この SAT データセットには問題となるワードペアがあり，その関係と同じものを探するための選択肢となるペアが各問題に対して 5 件出現する．例えば，問題として，(ostrich, bird)，選択肢として，1.(lion, cat), 2.(goose, flock), 3.(ewe, sheep), 4.(cub, bear), 5.(primate, monkey) があり，この場合は 1.(lion, cat) が正解となる．ostrich は大きな bird であり，lion もまた大きな cat であるからである．本論文では，この SAT データセットを用いて達成できる正解率を見ることで，システムの有効性，スコアリングの有効性，およびランキング手法の評価を行った．

Relational Search を用いてアナロジー問題を解くために，問題単語対と，選択肢ペアをクエリとした．そして，選択肢ペアのうち一つを “?” とし，Relational Search に投げた．例えば問題：(ostrich, bird)，選択肢の一つ：(lion, cat) の場合， $\{(ostrich, bird), (lion, ?)\}$  と  $\{(ostrich, bird), (? , cat)\}$  がクエリとなる．この場合，得られたランキングにおける cat と lion の順位の平均がこの選択肢のスコアとなる．このスコアを他の選択肢と比較し，最も低いものを正解候補した．そして，この正解候補と実際の正解を比較し，正解数を記録した．また，正解数を測る際に，6 つの相補スコアを初期スコアに対して加算する場合と，乗算する場合の 2 通りで比較した (式 2 の加算または乗算を使用)．結果を表 2 に示す．ここで，初期スコアは，式 1 により計算されたものを示す．

## 4. 考察

表 2 を見ると，初期スコアに対して最も精度があがったのは，初期スコアに対して相補スコアの平均を乗じたものであっ

表 2: SAT データセットにおける正解率

指標	精度	改善度
初期スコア	95 / 343 = 27.7%	-
初期スコア + 6 つの相補スコア	106 / 343 = 31.0%	+3.3%
初期スコア × 6 つの相補スコア	115 / 343 = 33.5%	+5.8%

た。これは、加算する場合に比べて、乗算する場合の方が相補スコアの影響力が大きく、初期スコアに比べて相補スコアを重視する方が良いことが分かる。また、初期スコアに対して、加算する方も乗算する方も総じて精度があがっていることから、相補スコアの有用性が言える。関係類似度に関する他の研究と比較した場合、最高は Turney ら [Turney 06] による 56.1% が最高であり、それに比べて非常に悪い結果である。しかし、関係類似度の研究では、“?”に相当する語を探すわけではなく、最初から全ての単語が与えられている。Relational Search で SAT を解く場合、“?”に相当する語を探す必要あるため、そもそも見つけれない場合がある。さらに、見つけれなかった場合でも、他の不適当な語がたくさん見つかった場合、適切な語に関する情報があまり得られなくなる。このことは、Relational Search を SAT に適用する際に大きな不利を受ける。なぜならば、SAT は一つの答えしか認められないからである。例えば、{(bird, ostrich), (cat, ?)} としたときに、答えの候補として jaguar, lion, cheetah, siberian tiger などの結果を得ることができる。これらは全て大きな猫であり、Relational Search の結果としては適切であるといえる。しかし、SAT の問題としてみると、適切なのは lion のみである。もし、他の語に関する情報のみが得られた場合、Relational Search の結果としては良いのに、SAT の結果としては悪くなる場合がある。このように、既存の関係類似度の研究と比較したとき、精度が悪くなる場合が多い。しかしながら、Relational Search の結果が悪いのに SAT の結果が良いということは考えにくい。Relational Search の精度を測る指標として SAT は有用であると考えられる。また、Relational Search の結果を見たときの解釈は様々に考えられるため、人間が評価しなければならない。しかし、SAT の場合、答えが一つに定まっているため、自動で評価を行えるという利点がある。SAT を用いて Relational Search の評価をした関連研究はないが、今後、自動的に行える客観的な評価指標として利用可能である。

## 5. まとめ

Relational Search の精度を上げるための手法として、対称性と相補語を用いたものを提案し、SAT データセットを用いて実験を行った。結果、相補スコアの影響力が大きい方が精度が高かった。このことから、相補スコアの有用性を示せた。

## 参考文献

- [Bollegala 09] Bollegala, D., Matsuo, Y., and Ishizuka, M.: Measuring the similarity between implicit semantic relations from the web, in *Proc. of WWW'09*, pp. 651–660 (2009)
- [Bunescu 07] Bunescu, R. C. and Mooney, R.: Learning to extract relations from the web using minimal supervision, in *Proc. of ACL'07*, pp. 576–583 (2007)

[FUNAHASHI 08] FUNAHASHI, T., UEDA, T., HIRATE, Y., and YAMANA, H.: Reliability Verification of Search Engines' Hit Count, *IPSJ SIG Notes*, Vol. 2008, No. 88, pp. 139–144 (2008)

[FUNAHASHI 09] FUNAHASHI, T., SONE, H., and YAMANA, H.: Reliability Verification of Search Engines' Hit Count using Multi Query, *IEICE technical report. Data engineering*, Vol. 109, No. 153, pp. 19–24 (2009)

[Turney 06] Turney, P.: Similarity of semantic relations, *Computational Linguistics*, Vol. 32, No. 3, pp. 379–416 (2006)

[後藤 10] 後藤 友和, Duc, N. T., Bollegala, D., 石塚 満: 潜在的関係を利用する検索システムの対称性による候補語リランキング法, 情報処理学会第 72 回全国大会講演論文集 (2010)