論　文

階層的自動タグ付けによるエキスパート検索エンジンの研究・開発

大平　倫宏*1)　　富山　真一*1)

Research and development of expert search system using hierarchical automatic tagging

Norihiro Ohira*1)，　Shinichi Tomiyama*1)

For solving problems inside an enterprise or research groups, finding the expert persons who have the pertinent knowledge is a key for solving the problems. In recent years, several expert search systems have been devised for finding experts. However, it has been pointed out that the conventional methods used in such systems may not be capable of properly finding experts in case where the search keywords have some synonyms or an expert has multiple areas of specialization. In this study, applying a hierarchical tagging technique, an expert finding system has been developed which can output more appropriate rankings in such cases.

キーワード：専門家検索，階層構造，タグ，情報検索
Keywords：Expert finding, Hierarchical structure, Tag, Information retrieval

1.　はじめに

企業内や研究グループ内等での問題解決では，特定の専門的な知識を利用することが重要である。適切な問題解決を行うためには，そのグループ内で，適切な知識を持った専門家をみつけることが重要である。グループの規模が小さいうちは，人手で専門家をみつけることは比較的容易であるが，組織が数百人を超える規模になると発見が困難になってくる。この問題に対応するため，近年では，キーワードから専門家を検索可能なエキスパート検索システムが考案されている(1)。しかし，これまでに提案されているシステムは，検索語が類義語を持つ場合や，一人の専門家が複数の専門分野を持つ場合において，正しく機能しないことが指摘されている(2)。

本研究は，上述したような問題点を解決することを目的とした。エキスパート検索は，検索そのものが目的ではなく，検索後に専門家から情報を得るためにより多くの時間を費やす必要があることから，検索結果として適切でない専門家が得られた場合の人的・時間的コストが他の検索システムに比べて非常に高いという特徴を持つ。このため，少しでも正確な検索結果が望まれる。本研究では，具体的なデータ構造として，階層的タグ付け構造を利用した。階層的タグ付け構造を利用することで，検索語が類義語を持つ場合でも，正確なランキングを生成できる。更に，階層的タグと検索語の関連度から，一人の従業員が複数の専門分野を持つ場合でも，正確に検索を行える。

事業名　平成 25 年度　基盤研究
*1) 情報技術グループ

2.　エキスパート検索方法

2. 1　従来法　　従来，エキスパート検索によく利用されている方法として，図 1 のように，専門家に対するタグを列挙して，そのタグとの関連性を基に検索を行う方法がある。図 1 では，専門家と対応するタグが記されており，括弧内の数値は専門家とタグとの関連度を表している。ここでは，数値が高いタグがより関連性が高いとしている。

2. 2　提案法　　本研究では，図 2 のような階層的な構造を用いた。図 2 では，専門家の名前もタグの一つとして扱い，タグ間の関連をリンクで表す。リンクに記された数値は，各タグ間の距離を表している。ここでは，類義語となるタグの間を横方向のリンクで繋げている。また，「検索」の中の「言語処理」等のように抽象性が異なるタグ間の繋がりを縦方向のリンクで繋げている。

専門家名	A さん
タグ	言語処理(2.0)，類義語(1.8)， 検索エンジン(2.5)

専門家名	B さん
タグ	超音波(4.0)，探傷試験(2.0)， 毒性物質(2.2)，害虫駆除(1.0)

専門家名	C さん
タグ	害虫駆除(2.8)，超音波(1.2)，毒性物質(2.1)

図 1.　従来の専門家に対する自動タグ付けの例

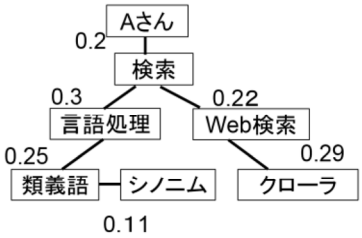

図 2. 今回提案する階層的な自動タグ付けの例

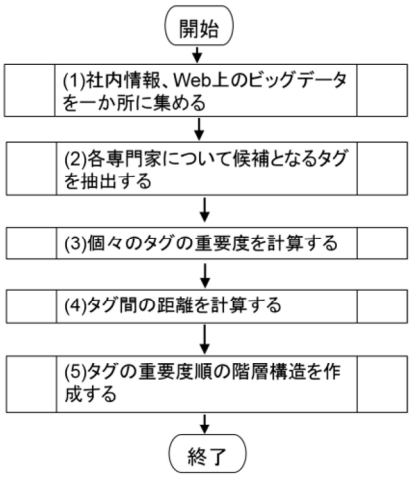

図 3. 階層的自動タグ付け方法のフローチャート

3. 提案法の階層構造作成方法

図 2 のような階層構造を自動的に作成するために，図 3 のフローチャートにしたがって，計算を行う手法を開発した。以下，図 3 のそれぞれの手順について説明をする。

3. 1 社内情報，Web 上のビッグデータを一か所に集める方法 Web クローラを利用して，専門家名等をキーワードに Web 上を検索し，関連する PDF ファイルや HTML ファイルを取得する。同様に，社内情報については，専門家が作成したドキュメント等のファイルを収集する。以下では，この手順で得られたファイルやドキュメントのことを，その専門家に関する文書として扱う。専門家ごとに，関連する文書を記録しておく。

3. 2 各専門家について候補となるタグを抽出する方法 3. 1 で収集した文書に対して形態素解析を行い，各文書に出現する専門用語の集計を行う。ある文書に一定回数以上出現する専門用語をその文書に関するタグとする。また，そのようにして得られたタグを，その文書に関連する専門家のタグの候補とする。

3. 3 個々のタグの重要度の計算方法 ある文書で最も出現回数の多いタグを，その文書のトピックとして扱う。トピックの総数をNとして，各タグtに対して，トピック$topic_k(k = 1,2,\cdots,N)$を持つ文書に対するタグ$t$の出現確率を$p(topic_k)$，タグの情報量を

$$H(t) = \sum_{k=1}^{N} p(topic_k)\log p(topic_k) \quad \cdots\cdots (1)$$

で定義する。また，$C(t)$を全文書中でのタグtの総出現数（タグtの数），$D(t)$を全文書中でのタグtの出現する文書数（文書の数)として，タグの重要度を

$$I(t) = H(t)\cdot D(t)\cdot(\log C(t) + 1) \quad \cdots\cdots (2)$$

で定義する。

3. 4 タグ間の距離の計算方法 i番目のタグt_iとj番目のタグt_jとの間の距離を以下の四つの特徴量からなるベクトルを用いて定義する。

(i) 頻度共起$Coo(t_i, t_j)$: 同じ文書中にタグt_iとタグt_jが同時に出現した場合の，その文書の総数。

(ii) リンク共起性$L(t_i, t_j)$: タグt_iの出現する文章からタグt_jが出現する文書へのリンクの総数と，逆に，タグt_jの出現する文章からタグt_iの出現する文書へのリンクの総数を足し合わせて 2 で割った数。ここでは，参考文献による引用を，文書間でのリンクとして定義している。

(iii) 時間的共起性$T(t_i, t_j)$: タグt_iの出現する文章の平均作成日から，タグt_jの出現する文章の平均作成日を引いた日数の絶対値。

(iv) トピックに対する特徴量$h(t_{ik})$: i番目のタグt_iにおけるk番目のトピック$(k = 1,2,\cdots,N)$に対する情報量

$$h(t_{ik}) = p(topic_k)\log p(topic_k) \quad \cdots\cdots (3)$$

をタグのN次元の特徴量とする。

これら 4 つの特徴量を用いて，タグ間の距離を，

$$d(t_i, t_j) = 1/Coo(t_i, t_j) + 1/L(t_i, t_j) + T(t_i, t_j) + \sum_{k=1}^{N}(h(t_{ik}) - h(t_{jk}))^2 \quad \cdots\cdots (4)$$

と定義する。

3. 5 タグの重要度順の階層構造の作成方法 (2) 式から，出現頻度が高く，多くのトピックに関連があり抽象度が高いタグほど，重要度が大きくなる。この性質を利用して，各専門家に対して，距離と重要度の和が大きい順にタグをソートして，その順番で階層構造に追加していくことで，全体の階層構造を作成する。

あらかじめ三つの閾値θ_d，θ_I，θ_Cを決めておく。二つのタグ間の距離$d(t_i, t_j)$が閾値θ_dよりも小さく，重要度の差$|I(t_i) - I(t_j)|$が閾値θ_Iより小さく，かつ，頻度共起$Coo(t_i, t_j)$が閾値θ_Cよりも小さかった場合に，その二つのタグを類義語として，横方向にリンクする。これは，特徴量が近いならば，同一のトピックの文書で多く出現し，それゆえに通常は同一の文書で多く現れるはずであるという仮説を基にしている。また，類義語であれば，同一文書中で現れたとしても，どちらかがメインとして使用され，他

方は，括弧書き等でだけ使用され，多くは出現しないと仮定している。

一方で，上記の類似性の条件を満たさないタグ間については，抽象レベルが異なっているとして，縦方向のリンクで繋げる。この際に，木構造が適切な深さと幅を持つようにする。具体的には，木構造に対して，その重さを

$$cost(T) = \sum_{i<j} d(t_i,\ t_j) \quad \text{(5)}$$

と定義して，タグを追加した新たな木構造をT'として，

$$\mathrm{f}\left(T'\right) = cost\left(T'\right) - cost(T) + depth(T')/\log\left|T'\right| \quad \text{(6)}$$

を最小化するような位置に，タグを追加する。(6) 式の第三項の分子は木の深さ，分母は木の中のタグの総数の対数である。これは，木が幅を持たずに一直線に深くなりすぎたり（図 4 左），逆に，幅ばかりが大きくなり階層構造を持たなくなったりするようなこと（図 4 右）を防ぐ。このようにして，専門家の名前をルートのタグとして，順次タグを加えていくことで，図 2 のような階層構造を作成する。階層構造中のタグの数があらかじめ決めた一定数を超えたところで，その専門家に対する階層構造の作成を終える。

特に新入社員などで，関連する文書数が十分に多くない者がいる場合がある。そのような専門家に対しては，3. 2 の方法では，タグの数が十分集まらないため，検索語と関連があるにもかかわらず，検索されない可能性がある。そのため，今回は，それらの専門家に関するタグと距離の近いタグを，その専門家と関連の無い文書からも拾ってきて，その専門家のタグとして，階層構造に追加する。

4. 提案法の検索方法

図 5 のように作成した階層構造に対して，検索の際には，検索語とある専門家との距離を，専門家と与えられた検索語全てを通る最短パスの距離で定義する。その際に，重複するパスが存在する場合には，そのパス分の距離の加算は一回のみとする。図 5 では，検索パスの例を太字で示している。パスが存在しない時には，距離を∞とする。

例えば，図 5 において，「言語処理」で検索した際，最短パスは「A さん」－「検索」－「言語処理」となり，「A さん」との距離は，0.2 + 0.3 = 0.5 となる。また，「言語処理　クローラ」で検索した際には，検索パスは図に示すようになり，0.2 + 0.3 + 0.22 + 0.29 = 1.01 となる。

5. 結果と考察

図 1 に示す従来型のタグ付け方法と，図 5 に示した，提案するタグ付け方法の比較を行った。従来法における検索方法としては，専門家の検索クエリに対する評価値として，関連度の和を利用する方法を用いた。評価する対象としては，239 人の専門家を対象とした。3. 1 の Web 上のデータとしては，Google Scholar[3]，CiNii[4]に対して，専門家名を検索クエリとして入力し，検索結果の PDF または HTML ファイルをダウンロードして利用した。PDF ファイルはそのままではプログラムから扱いづらいため，テキスト抽出を行ってテキストファイルとした。3. 2 の形態素解析では，技術用語約 10,000 語を形態素解析用の辞書として利用して，

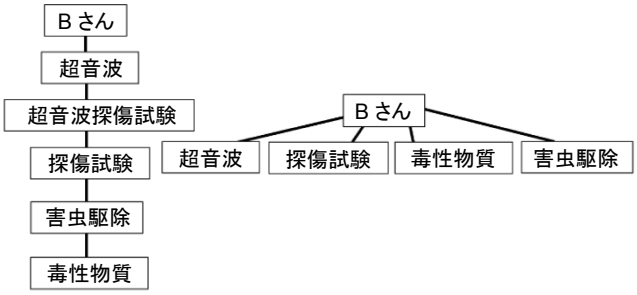

図 4.　不適切な階層構造

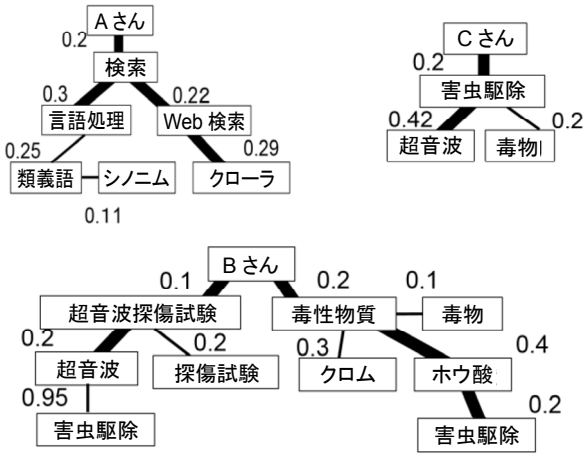

図 5.　階層的なタグ付けからの検索方法
太い線は距離計算時の検索パスの例を表している。

表 1.　検索結果例（「画像処理」で検索）

検索順位	従来法	提案法	正答
1	A さん(1)	E さん(3)	E さん(3)
2	B さん(2)	F さん(3)	F さん(3)
3	C さん(0)	B さん(2)	B さん(2)
4	D さん(0)	G さん(0)	I さん(2)
5	E さん(3)	H さん(1)	J さん(2)
6	F さん(3)	J さん(2)	A さん(1)
7	K さん(0)	I さん(2)	H さん(1)
8	J さん(2)	O さん(0)	Z さん(1)
9	M さん(0)	A さん(1)	C さん(0)
10	Z さん(0)	Z さん(0)	D さん(0)
…	…	…	…

名前の後ろの括弧書きの数値は，検索語との関連度を 4 段階で表している。

表 2.　NDCG 法による検索結果の比較（20 件の平均）

	従来法	提案法	正答
NDCG@5	0.612	0.814	1.0

形態素解析を行った。また，それだけでは，一部のタグについて，タグの距離等を求める際に十分なデータが得られなかったので，3. 2，3. 3，3. 4 において，現代日本語書き言葉均衡コーパス(5)を用いて，コーパス中の文章のタグとトピックから得られたデータを，タグ間の距離計算中の（3）式を計算するために利用した。

比較方法としては，あらかじめ，20 の検索語に対して，それらに対応する各専門家の関連性を 0 から 3 の 4 段階（3 が高い関連性を，0 は無関係を表す。）で定めておき，正答のランキングとした。各手法の検索結果に対して，ランキングの正当性を示す指標である NDCG@5(6)を用いて評価した。表 1 は，検索語として「画像処理」を選んだ場合の検索結果の例を表しており，提案法の方が従来法に比べ，関連度の高い専門家が上位にランキングされている。このことを数値的に表したのが，表 2 である。表 2 は，20 件の検索語に対する NDCG での比較結果を示している。NDCG は，最大値 1 で，値が大きな方が性能がよいとされており，提案法が従来法に比べ優れていた。

このように精度が改善した理由として，提案法では，一人の専門家が複数の専門分野に対応している場合にも，検索の精度を落とさないことが挙げられる。例えば，従来法のタグ付けの結果として，図 1 が与えられた際に「超音波害虫駆除」で検索した時，B さんの評価値は 4.0 + 1.0 = 5.0，C さんの評価値は 2.8 + 1.2 = 4.0 となり，B さんの評価の方が大きくなる。実際には，図 5 のように，B さんは超音波探傷試験と毒性物質の専門家であり，必ずしも超音波を用いた害虫駆除に詳しくないにもかかわらず，あたかも超音波で害虫駆除を行っている専門家のように評価される。一方，提案法を用いて図 5 の階層構造が得られた時には，検索語「超音波　害虫駆除」と「B さん」を含む最短のパスの距離は，0.1 + 0.2 + 0.2 + 0.4 + 0.2 = 1.1，検索語と「C さん」を含む最短パスの距離は 0.2 + 0.42 = 0.62 となり，C さんの方をより専門家として評価するようになる。「超音波探傷試験」に関連する「超音波」と「害虫駆除」に関連する「害虫駆除」を区別して捉えることで，従来法での問題点が改善される。この例のように，タグ間の関連を階層で捉えることにより，一人の専門家が複数の専門分野を持つ場合でも，より正確に評価可能となり，NDCG にもその結果が現れたと考えられる。

提案法の精度は従来法よりも改善されているが，表 1 の G さんのように，無関係な専門家もリストアップされている。これは，3. 5 において，その専門家と関連の無い文書のタグを取り込んでいるためと思われる。特に新入社員は，文書数が 2, 3 の者もおり，関連の無いタグを多く追加するためノイズが生じる。

6. 今後の課題とまとめ

提案法の課題として，検索のための処理に非常に時間がかかることが挙げられる。特に時間を要するのは，検索を行えるようデータを整理するための 3. 1 から 3. 5 の処理である。提案法では，約 300 人の専門家を対象とした場合，全体の計算時間は 1 台の PC で数日程度かかる。更に，提案手法は，既存のデータに対してオフラインで計算を行うことを前提としているため，途中で専門家の対応する分野が増えた場合や新たな技術用語が誕生した場合などに対応しようとすると，最初から計算し直す必要がある。現在の科学技術の発展速度は目覚ましく，新たな技術用語は次々に生まれているため，時間の経過と共にデータが変化した場合にも，比較的短時間で対応できるような方法を考案することが課題となる。

また，検索を行う際にタグ間の最短距離を求めているが，現状ではその部分の計算負荷が大きく，専門家一人に対して，1CPU コアでの計算で，数 100 ミリ秒程の時間がかかっている。実際に提案手法を利用する際には，検索クエリに対する検索結果をキャッシュする方法や，大規模なクラスタを組む方法を用いて，最短距離を求めるアルゴリズムを改良し，ユーザビリティを損なわないように注意する必要がある。

新たな分野での製品開発を行う企業においては，その分野の専門家とコンタクトを取ることは重要である。また，成長企業では，人数が増加するにつれて，誰がどの分野の専門家かを把握することは困難となる。今後，このような開発型中小企業に向けて，本研究成果の活用を働きかけていく。

（平成 26 年 7 月 7 日受付，平成 26 年 8 月 20 日再受付）

文　献

(1) Pavel Serdyukov, Mike Taylor, Vishwa Vinay, Matthew Richardson, and Ryen W. White : “Automatic People Tagging for Expertise Profiling in the Enterprise”, Proceedings of the 33rd European Conference on Advances in Information Retrieval, Dublin, Ireland, pp.399-410 (18-21 April, 2011)
(2) Craig Macdonald and Iadh Ounis : “Expertise Drift and Query Expansion in Expert Search”, Proceedings of the 16th ACM Conference on Information and Knowledge Management, Lisbon, Portugal, pp.341-350 (06-10November, 2007)
(3) Google Scholar, http://scholar.google.co.jp
(4) CiNii, http://ci.nii.ac.jp
(5) 現代日本語書き言葉均衡コーパス, http://www.ninjal.ac.jp/corpus_center/bccwj
(6) Kalervo Jarvelin and Jaana Kekalainen : “Cumulated Gain-Based Evaluation of IR Techniques”, ACM Transactions on Information Systems, 20(4), pp.422-446 (2002)