

多言語ブログにおける文化間ギャップ発見システム

ー海外の人の考え、知りたくありませんか？ー

1. 背景

近年、多くの人が海外に出かけ、また多くの外国人が日本を訪れるようになった。また、それに伴って、異文化交流の機会も増えてきた。そのような際に、自分の知っている知識や考えと相手が知っている知識や考えのギャップに驚く人も多いのではないだろうか。

日本と海外で捉え方が違うトピックの例として「臓器移植」がある。日本では臓器受け渡しを行っている団体は、日本臓器移植ネットワークのみである。さらに、「脳死＝死」という認識が日本ではまだ弱いために、脳死判定に異論を唱えている人も多い。アメリカと比較しても移植件数は著しく低く、臓器移植が浸透していないことが分かる。海外では臓器移植は盛んに行われているが、臓器不足が問題となっている。また、脳死判定に関する議論はほとんど無い。

このような問題に対して多くの人はどう思っているのだろうか。多くの人がこのような問題に対してどう思っているのかという情報を得るために、新聞社やテレビ局などのマスコミは共同通信の記事や他国記事の映像を購入したり、特派員を派遣して現地取材を行ったりする。また、シンクタンクが海外調査を行ったり、ネット上の情報を人海戦術で調査したりするかも知れない。しかし、このような情報のある部分は、現地に足を運ばずともウェブ上の情報を調べるだけで知ることが出来る。なぜなら、インターネットの普及や、WEB2.0 的コンテンツの発生によって、多くの人々がウェブ上に意見を書き込むようになったからである。そのため、日本にいながら、外国の多くの人が書いた意見や情報を読むことが出来る。

2. 目的

本プロジェクトでは、ユーザの入力するトピックに対して、文化間のギャップを発見する手がかりとなるような、ブログ中のキーワードや文章を提示するシステムを開発することを目指す。ユーザが使いやすいシステムにするためには、トピックの選択が容易にできること、また、文化間ギャップのあるキーワードをすぐに探し当てることができることの 2 点があげられる。そこで、本プロジェクトでは、トピックを分類し、文化間での異なりがどの程度あるのかを発見しやすい検索システムの製作を目指した。

ユーザが入力したトピックに対して、ブログから得られた共起語をキーワードとして提示し、またその共起語に関連するブログ記事をランキングして提示するシステムを開発した。また、ユーザが文化間ギャップのあるトピックを探し出すために、トピックをカテゴリわけし、分野ごとにトピックを調べることを可能にした。さらに、調べたいトピックが決まっている場合にすぐにそのトピックの情報を得られるように、トピックの検索を行う機能を実装し、また、共起語から調べたいトピックをたどることのできる共起語検索機能も実装した。共起語の提示に関しては、共起語マップを作成し、各共起語が日本語と英語のどちらで多く語られているか、どの程度の頻度で語られているかを、ユーザが直感的に知ることのできるようにした。

3. 開発内容

本システムを開発するにあたって必要な各項目について、以下の節で説明する。

3.1 検索トピックの選定

本システムはWikipediaを情報源として利用する。そのため、システムの対象となるトピックは、「Wikipediaに日本語と英語のエントリがあるトピック」とした。さらに、エントリ名を検索トピックとしてブログサイト検索を行ったときに、検索ヒット数が1万～50万の範囲のものに、多くのブログ記事がありそうなトピックが集中していることがわかっている。そこで、本システムはWikipediaに日本語と英語のエントリがあり、かつ日本語、英語共にヒット数が1万から50万の範囲のWikipediaエントリのエントリ名を対象トピックとする。また、これらの条件に当てはまるトピックは約6000トピックほど存在する。しかし、システムの実装に必要なデータを6000トピック分収集するには時間がかかりすぎるため、これらの6000トピックから、人手で選定した訳200トピックを対象とした。

また、Wikipediaエントリは、それぞれ親となるカテゴリを持つ。選定した200トピックについて、Wikipediaのカテゴリを対応付け、さらに人手で調整を加えた。

3.2 共起語マップの作成

各共起語について、ユーザが直感的に、日本語に特徴的か英語に特徴的かを把握するため、共起語マップを作成した[図2]。

共起語マップの横軸は言語特徴量とし、共起語が日本語の記事と英語の記事のどちらに多く出現するかで日本語に特徴的か、英語に特徴的かが決定する。共起語の日本語で出ている割合と英語で出ている割合を計算し、その比を用いて、横軸の座標を決定した。

また、縦軸を話題度とする。共起語が各言語でどの程度語られているかを表す。そのため、共起語が各言語のブログ記事内で語られる頻度を求め、頻度が多いものを高い位置へ、低いものを低い位置へ配置した。

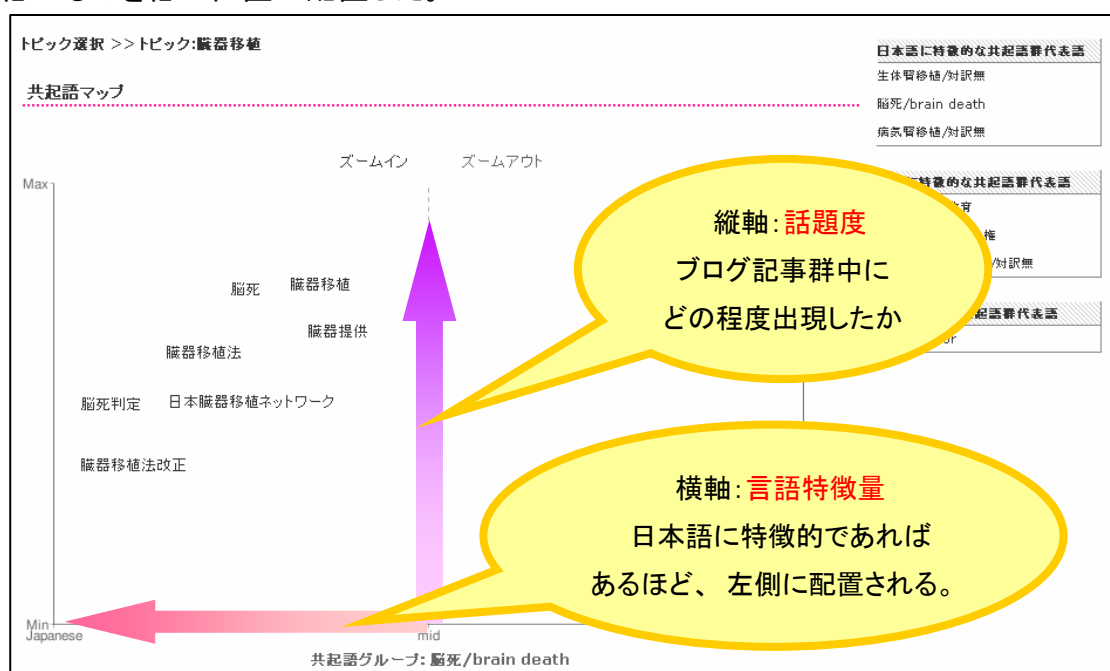


図 2. 共起語マップ

3.3 インタフェースの製作

これまでに述べた機能をわかりやすくユーザに提示するためのインタフェースを製作した。作成には PHP というプログラミング言語を用いた。

インタフェースはユーザがトピック選択を行うと共起語マップへ移動し、さらに共起語からブログ記事へと移動できるように作られている[図 4~6]。



図 4. トピック選択画面

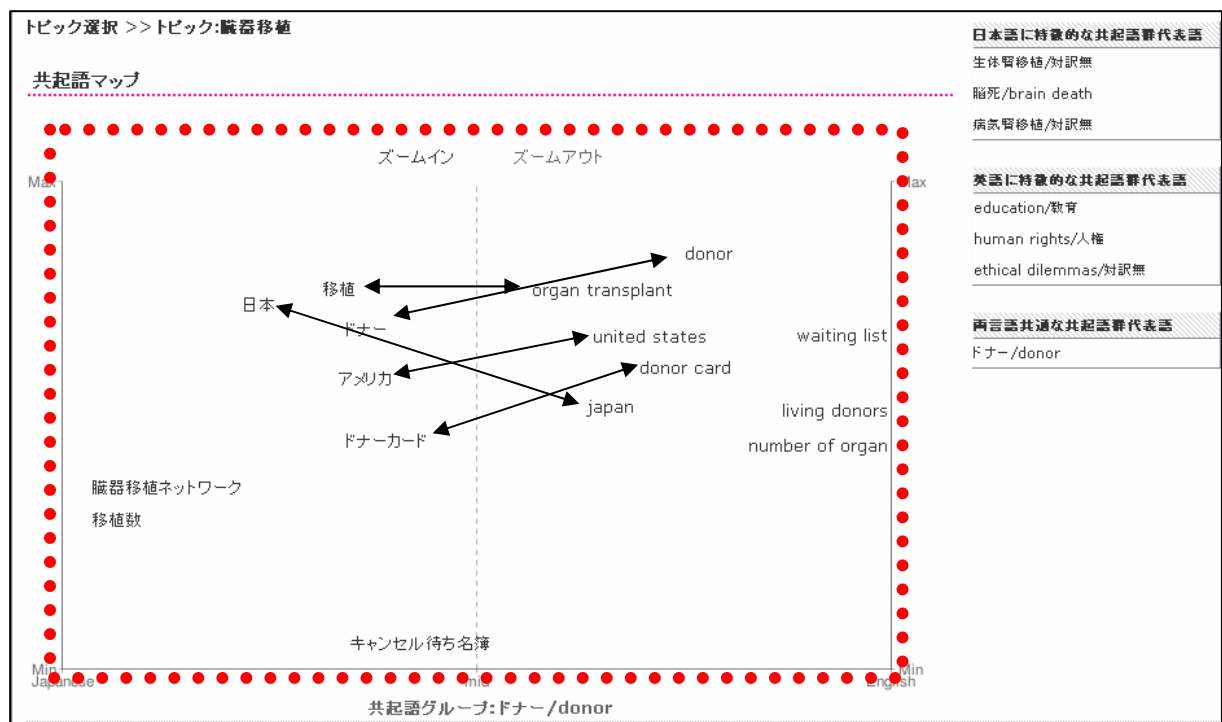


図 5. 共起語マップ画面(臓器移植)



図 6. ブログ記事ランキング画面

4. 従来の技術(または機能)との相違

従来の検索エンジンでは Kizasi.jp など日本語ブログを対象にトピックに関連するキーワードやブログ記事を提示するシステムが存在するが、本プロジェクトのように二言語を対象としたシステムは過去にあまり例がない。

また、ユーザは入力したトピックについて特徴的な共起語を眺めることで、大量の英語文書(もしくは日本語文書)を読まなくとも、文化間ギャップのある可能性の高い記事にたどり着くことのできるシステムを開発した。共起語をマップ化したことにより、より直感的に文化間のギャップ発見を支援できるようになった。

5. 期待される効果

今回作成したシステムは、マーケティングや社会系問題の意見調査に活用することが出来る。日本語ブログはもちろん、トピックについて詳細に述べている英語ブログも提示するため、日英間の比較が行いやすく、ブログの検索時間を短縮することができる。特に日本語ブログから得ることはできない英語ブログに特徴的な情報を提示することが出来るので、海外戦略向けのマーケティングに対応できるのはこのシステムの強みだといえる。また、社会系問題のトピックに関する英語独特の情報も提示することができる。例として、新型インフルエンザを挙げる。新型インフルエンザは日本で発症した病気ではないため、発生当初は病気の詳細情報が不足したり確定した情報を取得したりすることが困難であった。しかし、本プロジェクトのシステムを利用することで、医療専門家による英語ブログ特有の情報をユーザに提示することが可能である。また、今後システムで扱うトピックを増やすことによって、様々な分野のトピックの情報を提示することができるので、より多くの人にとって利便性の高いシステムにすることができる。

6. 普及(または活用)の見通し

今後の課題として、以下の3点が挙げられる。

- トピック数の増加

現在のデータは約 200 トピックと大変少ない。より網羅的に様々なトピックを扱うためにトピックを増やすことは急務である。3. 1 節でも述べたとおり、Wikipedia に対象トピック候補となるものが約 6000 個あることが確認されている。この 6000 個の候補の中からトピックを増やすことによって幅広い分野の情報を網羅することが出来るので、より多くの人のニーズに応えることができるシステムにすることができる。

- 最も情報量の高い共起語の組み合わせの提示・要約文の追加

現在の共起語マップでは、ユーザにとってあまり情報量が高くない共起語が提示されている。今後トピック数の増加と共に、多くのユーザにとって最も情報量の高い共起語の組み合わせを自動生成することで共起語マップの性能を上げていく。その結果、そのトピックにおいて重要な共起語をわかりやすく提示することができ、ユーザによる文化間ギャップ発見の支援強化につながる。また、ユーザがより文化間ギャップを発見しやすい共起語マップにするために、提示している共起語を含むブログの要約文を提示できる機能を今後追加していきたい。

- ユーザによるコメント機能の追加

より多くの人を使いやすいインタフェースを目指すために、ユーザからシステムに関する意見を出し合える環境を作りたい。具体的には、コメント欄やWiki等の機能を実装する予定である。これらを実装することによって、ユーザにとってどのようなシステムが使いやすいのかシステム管理者が把握することができるので、ユーザの意見を即座にシステムに反映することが出来る。また、コメント機能を追加することにより、それぞれのトピックにおいて、ユーザ間で議論を行うことが可能になる。インタフェース上に提示された情報を閲覧するだけでなく、ユーザからもトピックに関する意見や情報を発信することで、リアルタイムで多くの情報が共有できるシステムとなる。

7. 開発者名(所属)

川場 真理子(日本電信電話株式会社 NTT サイバースペース研究所)

中崎 寛之(筑波大学大学院システム情報工学研究科知能機能システム専攻)