

国立国語研究所学術情報リポジトリ

Enhancement for Supporting Language Analysis in Full-Text Search System "Himawari"

メタデータ	言語: jpn 出版者: 公開日: 2018-03-20 キーワード (Ja): キーワード (En): 作成者: 山口, 昌也 メールアドレス: 所属:
URL	https://doi.org/10.15084/00001477

全文検索システム『ひまわり』における言語分析支援機能の拡張

山口昌也 (国立国語研究所音声言語領域) †

Enhancement for Supporting Language Analysis in Full-Text Search System “Himawari”

Masaya YAMAGUCHI (Spoken Language Division, NINJAL)

要旨

本稿では、筆者が開発している全文検索システム『ひまわり』の言語分析支援機能の拡張について述べる。元来、『ひまわり』は言語資料の検索と閲覧を目的に設計されたコンコーダンスであり、検索結果を分析するための機能を十分に備えていなかった。しかし、検索対象の資料の規模が大きくなると、大量の検索結果を単に表示するのではなく、集約して分析する必要性が生じる。また、検索結果の統計的な分析には、資料に含まれる文字数といった、基本的な情報を計測できなければならない。そこで、(1) 検索結果の集約機能、(2) 統計的分析のための基礎データの収集機能を『ひまわり』に実装した。拡張された機能を用いることにより、例えば『名大会話コーパス』の各会話中の発話数、文字数、単語数、特定の単語の出現数といった情報を収集できるようになる。

1 はじめに

全文検索システム『ひまわり』(山口昌也・田中牧郎, 2005)¹は、言語研究用に設計された全文検索システムであり、XMLでタグ付けされたテキストを全文検索することができる。もともと、2005年に公開された『太陽コーパス』を検索するためのシステムとして開発された。本論文の主題である、分析支援機能としては、検索結果を閲覧するためのコンコーダンスとしての機能、および、収録された資料を指定した形式で表示する機能を持っている(図1)。これらの機能では、利用者が検索結果や言語資料を「目で見える」ことの支援に焦点が当てられている。そのため、検索結果の集約や統計的分析については、R(統計分析用プログラミング言語)²やMicrosoft Excelなどの外部プログラムに検索結果をエクスポートして処理するという前提である。

当初の設計から10年以上を経て、『ひまわり』にはさまざまな改良が加えられているが、分析を支援する機能については、基本的に当初のままである。その一方で、言語資料は大規模化し、個人のレベルでも入手できるようになった。『ひまわり』でも、青空文庫、国会会議録、名大会話コーパス、Wikipediaなどを手軽に検索できるようになっている³。

言語資料が大規模化すると、検索結果は増大するため、目で見きれなかったり、そもそも、メモリ不足などにより、検索結果を表示しきれないということも起こりうる。したがって、当初の設計のように検索結果を単純に表示するだけでなく、集約して表示する必要性が生じる。また、統計的な分析を行うためには、各種の統計量を計算するための基礎的なデータ(例:総文字数、総発話数)を利用者が自由に計測できなければならないが、現状の『ひまわり』では計測自体ができなかったり、できたとしても長時間の処理が必要になっていた。

そこで、本稿では、『ひまわり』の分析支援機能の拡張として、検索結果の集約方法、および、統計的な分析の支援方法の設計を行い、実装した結果を示す。

[†]<http://www2.ninjal.ac.jp/masaya>

¹<http://www2.ninjal.ac.jp/lrc/>の『ひまわり』ホームページから無料でダウンロードできる。

²<https://www.r-project.org/>

³『ひまわり』のホームページで配布している。簡単にインストールできるよう、パッケージ化されている。

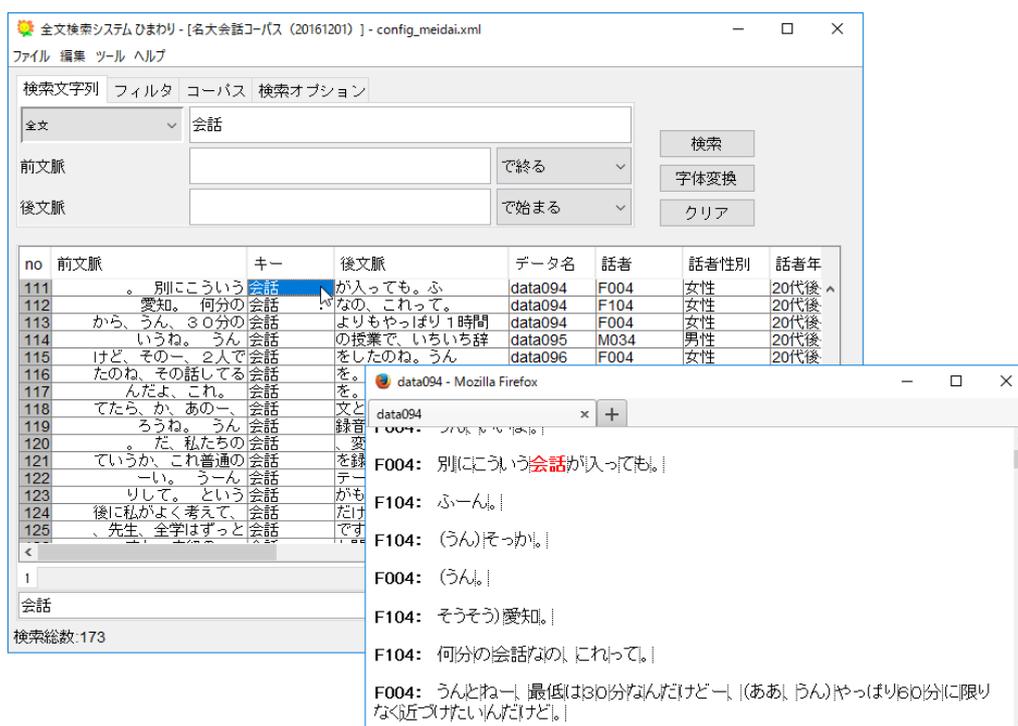


図 1: 『ひまわり』の実行例

2 既存システムと本システムの位置づけ

これまでに、言語資料の利用を支援するためのさまざまなシステムが開発されている。分析の支援という観点から見ると、支援のための機能は、大きく分けて、次の三つに分類されることが考えられる。なお、例として挙げたシステムは、当該分類だけに含まれるわけではなく、複数の分類に入りうる。

- (1) コンコーダンスとしての機能（「中納言」⁴(小木曾智信ほか, 2011), 「梵天」⁵ など)
- (2) 検索語のコロケーションを表示するなど、検索結果を集約して表示する機能 (AntConc(Anthony, 2016), NINJAL-LWP(今井新悟ほか, 2013) など)
- (3) 検索結果に対する統計的分析ツールとしての機能 (KHCoder⁶ (樋口耕一, 2014)) など

現状の『ひまわり』の分析支援機能は、(1)である。また、前節で述べたとおり、(2)(3)は現状の問題を解決するための解決方法となる。したがって、支援機能拡張の方向性として、『ひまわり』の特徴を活かしつつ、(2)(3)を実現することが好ましい。

他のシステムに対する『ひまわり』の特徴は、XMLによりアノテーションされた多様な形式の言語資料を検索し、アノテーションされた情報を検索結果として抽出できることである。また、『ひまわり』改良の過程では、言語資料の作成を支援する機能として、テキストのインポート機能が実装されている(山口昌也, 2013)。この機能を用いると、テキストに付与された独自形式のタグや、テキスト表記上の規則(例:「太郎:」のように、行頭の:で区切られた文字列は、後続する文字列の発話者を表す)をXMLに基づいたアノテーションをインポート時に行うことができる。

以上の特徴をまとめると、言語資料の作成者が必要であると考えてアノテーションした結果を分析に活かせるということである。したがって、新しい『ひまわり』を、アノテーション結果の「分析」

⁴<https://chunagon.ninjal.ac.jp/>

⁵<http://pj.ninjal.ac.jp/corpus.center/nwjc/bonten-overview.html>

⁶<http://khc.sourceforge.net/>

支援機能を持ったコンコーダンスとして位置づける。アノテーションの分析機能を、上記の(2)(3)と結びつけていく方法については、次節で述べる。

3 基本的な設計方針

ここでは、アノテーション結果の分析支援という面から、検索結果の集約機能と、統計的分析機能についての設計方針を示す。

まず、『ひまわり』の検索結果を集約することは、検索結果が検索文字列とそれに付随するアノテーションから構成されるため、アノテーション結果を集約することに他ならない。現状の『ひまわり』でも、検索結果から選択した任意の列から、出現頻度付きの一覧表を作成することによって、検索結果を集約することができる。例えば、図1の「話者性別」列を選択して、一覧作成機能を実行すれば、検索文字列に対して、性別ごとの出現頻度を求めることができる。ただし、検索結果を求めた後でないと、一覧が作成できないという問題がある。そのため、前節で述べたように、検索結果が大量だとメモリ不足などの問題が発生する。また、後述するように、列選択による単純な一覧作成では、求める結果を得られない場合もある。そこで、本稿では検索結果の集約方法について、次の機能拡張を行う。

- 検索時の一覧作成機能 (4.3 節)
- 一覧作成機能の改善 (4.4 節)

次に、統計的分析については、分析機能自体は『ひまわり』に持たせないかわりに、統計的分析のための基礎データをアノテーション結果から得られるようにする。この理由の一つは、すでに統計的分析用の有用なツールが存在することである。もう一つの理由は、分析ツールに入力するデータ自体(ここではアノテーション結果)を利用者が確認した上で、分析ツールを利用することが重要であるからである。本稿で扱う拡張は、次の2点である。これらは、アノテーション結果から統計的分析用の基礎データを収集する際の問題を解決する。

- アノテーション結果の集計機能 (4.1 節)
- 外部アノテーション機能の改善 (4.2 節)

前者は、現状の『ひまわり』では、検索文字列を指定して検索しないと、アノテーション結果を抽出できない、という問題を解決する。後者は、形態素解析結果など、大量のアノテーションを行う場合に用いられる「外部アノテーション」機能の改善である。この機能はデータサイズ増大に対応するための機能であるが、現状では十分な性能が得られていない。

4 拡張機能の実現

4.1 アノテーション結果の集計機能

前述のように、『ひまわり』用の言語資料にはさまざまな研究用の情報がアノテーションされている。ここでは、付与されているアノテーションを集計する方法について考えてみる。

例えば、『ひまわり』用の「名大会話コーパス」パッケージの場合(会話データ data016 の冒頭部分を引用)、次のように XML で記述されている⁷。

この例には、「始めまーす」「はーい」という二つの発話が含まれている。冒頭の meidai タグは、一つの会話全体に対してマークアップするタグである。u, s タグは、それぞれ発話、単語(短単位)を表す⁸。それぞれのタグは、属性を持つことが可能である。例えば、(3行目の)「始め」をマークアップしている s タグは、l (基本形), p (品詞), f (活用形), c (活用型) の属性を持っている。

⁷紙面の都合上、一部の属性・タグを省略している。また、見やすいように、適宜改行を入れている。

⁸名大会話コーパスの規模は 142 万語程度なので、外部アノテーションではなく、XML でアノテーションしている。

```

<meidai name="data016" speakers="F004,F028" duration="56 分" ns="14023">
<u s="F004" i="0" sex="女性" age="20 代後半">
<s l="始める" p="動詞-非自立可能" f="連用形-一般" c="下一段-マ行">始め</s>
<s l="ます" p="助動詞" f="終止形-一般" c="助動詞-マス">まーす</s>
<s l="。" p="補助記号-句点" f="" c="">。</s>
</u>
<u s="F028" i="0" sex="女性" age="20 代後半">
<s l="はい" e="ハイ" p="感動詞-一般" f="" c="">はい</s>
</u>
:
</meidai>

```

名大会話コーパスには複数の会話データが含まれているが、会話データ数を計測するには meidai タグを、単語数を計測するには s タグを列挙すればよい。しかし、現状の『ひまわり』では、仕様上、言語資料の作成者が列挙内容を事前に定義しておくか、すべてのタグの情報が表示されるような検索を行う⁹方法しか用意されておらず、いずれも、実現可能性、計算時間の点で現実的ではない。

そこで、一覧に表示するタグとその属性を利用者が対話的に指定できるようにした。図2は、u タグの s 属性を指定し、話者の一覧を表示した例である。一覧を作成する対象となるタグの指定を左図のダイアログで行う。ここでは、「第一階層タグ」で u タグを選択し¹⁰、その右のボタンを押すことにより、属性選択ダイアログ(図中央)が表示されるようになっている。この例では、タグ指定のダイアログで、「頻度」オプションをチェックしているため、単に話者の一覧を表示するだけでなく、話者(u/s 列)ごとに発話数(頻度列)が表示される。また、一覧の下部に表示される総数と異なりによって、総発話数、および、総発話者数を求めることができる。

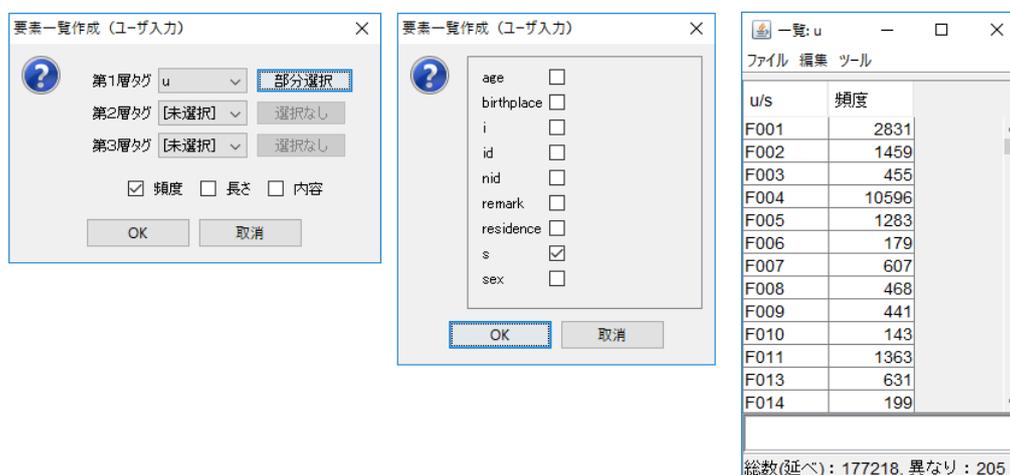


図 2: アノテーションの集計例

一覧に表示されるタグは、3階層まで指定可能である。一覧としては、最下層のタグの一覧が作成されるようになっているが、作成の過程で上位階層の属性を一覧に取り込むことができる。これらの情報は、一覧作成対象のタグに制約を加えたり、付随的な情報を集計するために用いられる。これにより、次のような一覧を作成することが可能である。

⁹例えば、すべての単語を列挙するには、正規表現で任意の文字列(^\.\$)にマッチするような条件を指定する。

¹⁰指定できるタグは、選択リストに表示される。

- 会話データごとに話者の発言数を求める
[第1階層タグ meidai (name 属性), 第2階層タグ u (s 属性), 「頻度」オプション]
- 会話データごとに総単語数を求める
[第1階層タグ meidai (name 属性), 第2階層タグ s (属性は無指定), 「頻度」オプション]

一覧表示の際のオプションには、「頻度」以外にも「長さ」「内容」がある。「長さ」は、最下層のタグでマークアップされているテキスト長¹¹を求める。例えば、これにより、会話データ (meidai タグ)、発話 (u タグ) に含まれる文字数の一覧を作成することができる。「内容」はタグでマークアップされているテキストを表示する。このオプションは、発話内容をすべて列挙する場合などに利用できるだろう。

4.2 外部アノテーション機能の改善

4.2.1 検索性能の改善

『ひまわり』では、テキストに対するアノテーションを記述する方法として、2種類の方法が用意されている。一つは、4.1節で示したように、XMLで記述する方法である。もう一つは、外部のリレーショナル・データベースに記述する方法である。後者の場合、アノテーションする情報は、テキストの位置情報と関連付けられて、リレーショナル・データベースに格納される。

後者の方法を導入した背景には、コーパスファイルの巨大化への対応がある。特に、形態素解析システムによって得られた結果は、統計的な分析を行う上で基本的な情報であるが、書誌情報や発話情報のアノテーションなどと比較して、アノテーションの量が多くなる。また、『ひまわり』で扱える形式のXML文書では、すべての形態素に対して、付随するすべての情報を記述しなければならず、冗長な記述とならざるを得ない。そのため、XMLで直接記述すると、『ひまわり』で扱えるコーパスファイルの上限を越えてしまう場合が出てくる。

以上の背景のもと導入した方法だが、(検索はできるものの) データベース用のファイルサイズが巨大になるという問題があった。例えば、『ひまわり』用に公開している『青空文庫』パッケージの場合、データベースのサイズが約6.2GBにもなる。

そこで、外部アノテーション検索用に用いていたリレーショナル・データベースを独自のデータベースで再実装した。検索処理は、(1) マークアップしている範囲の検索、(2) マークアップされているタグの属性検索に分けられる。(1)には山口昌也・田中牧郎(2005)と同様に2分検索を使用し、(2)はアノテーション集合(形態素解析結果の場合、辞書に相当)をメモリ上で線形検索する。(2)のタグの属性検索は、正規表現にも対応する。

表1に、新旧データベースのファイルサイズを示す。青空文庫は『青空文庫』パッケージ(20160401版)、国会会議録は『国会会議録』パッケージ(20140327_rev20170201版)を用いた。また、検索速度の参考値として、検索キーとして「あの」(基本形)を検索した時の実測値(CPU: Xeon E5-1620 3.7GHz 4core, OS: Ubuntu 16.04, Memory: 24GB)を示す。検索総数は青空文庫で78561件、国会会議録で41949件である。使用した『ひまわり』は ver.1.5.5(旧), ver.1.6.a20170120(新)である。

この結果のとおり、言語資料のサイズは青空文庫で約26%に削減された。検索速度も約2.5倍に向上している。新しいデータベースはサブコーパスごとに分割することも可能になっているため、配布の際の問題¹²も軽減されると考えられる。

4.2.2 外部アノテーション内容の閲覧機能

外部アノテーションの内容は、XML文書中に直接記述されないため、実際にどのようなアノテーションがなされているのかを確認しづらい。特に、形態素解析システムなどのツールによるアノテ

¹¹マークアップされているタグ、改行文字を除外した上で計測される。

¹²一般公開する場合は、ファイルを圧縮しているが、巨大すぎて使用環境によっては展開できないなどの問題が発生する。

表 1: 言語資料のサイズと検索速度 (参考値)

言語資料	旧 (サイズ)	新 (サイズ)	旧 (検索)	新 (検索)	総語数
青空文庫	6.2GB	1.6GB	8.9sec	3.6sec	1.0 億語
国会会議録	—	4.1GB	—	2.8sec	2.7 億語

ション結果は、誤解析も含まれるため、一定の範囲で外部アノテーション結果を確認する手段を用意すべきである。

従来の『ひまわり』では、検索結果をダブルクリックすると、当該の会話データや作品全体を Web ブラウザで表示できるようになっている。今回の拡張では、外部アノテーション結果でも会話や作品全体を一覧できるようにした。

図 3 は、『ひまわり』に付属する青空文庫サンプルから芥川龍之介の「蜘蛛の糸」全体の形態素解析結果を表示した結果である。この一覧は、検索文字列「釈迦」を検索し、その検索結果の一つをダブルクリックすることにより表示される。この図のとおり、選択した検索結果(「釈迦」)の出現位置にジャンプする。前後の形態素は上下に位置し、“_TEXT”列がテキストでの出現形である。この一覧を使えば、作品ごとに語彙や品詞の分布を容易に求めることができる。また、『ひまわり』のソート機能を使って、ランダムにソートすれば、ランダムサンプリングが実施できる。

SER.NO.	_TEXT	品詞	品詞細...	品詞細...	品詞細...	活用型	活用形	基本形	読み	発音
UUUU1734	へ	助詞	格助詞	一般				へ	へ	エ
00001735	落ち	動詞	自立			一段	連用形	落ちる	オチ	オチ
00001736	て	助詞	接続助詞					て	テ	テ
00001737	しまっ	動詞	非自立			五段・ワ...	連用タ接...	しまう	シマッ	シマッ
00001738	た	助動詞				特殊・タ	基本形	た	タ	タ
00001739	の	名詞	非自立	一般				の	ノ	ノ
00001740	が	助詞	格助詞	一般				が	ガ	ガ
00001741	,	記号	読点					,	,	,
00001742	御	接頭詞	名詞接続					御	ゴ	ゴ
00001743	釈迦	名詞	一般					釈迦	シャカ	シャカ
00001744	様	名詞	接尾	人名				様	サマ	サマ
00001745	の	助詞	連体化					の	ノ	ノ
00001746	御	接頭詞	名詞接続					御	ゴ	ゴ
00001747	目	名詞	一般					目	メ	メ
00001748	から	助詞	格助詞	一般				から	カラ	カラ
00001749	見る	動詞	自立			一段	基本形	見る	ミル	ミル
00001750	と	助詞	接続助詞					と	ト	ト
00001751		記号	読点							

総数(延べ): 2083

図 3: 外部アノテーション結果の表示例 (芥川龍之介:「蜘蛛の糸」)

4.3 検索時の一覧作成機能

検索時の一覧作成機能は、利用者が実行した検索の結果をそのまますべて表示するのではなく、集計した結果を表示するものである。この機能は、メモリ不足の問題が発生するような、大量の検索結果が得られる場合に用いることを想定している。従来版の『ひまわり』でも (検索結果をすべて表示

するのではなく) 検案件数のみを表示することはできたが、この機能では、アノテーションの集計と同様に各種の付与属性を含めて一覧表示できるようにした。

図4右は、出現形「が」の検索結果を集計機能を使って一覧表示した結果である。左図は、検索条件と一覧表示のための設定である。一覧表示したい属性は、左図のようにマウスで選択しておく。

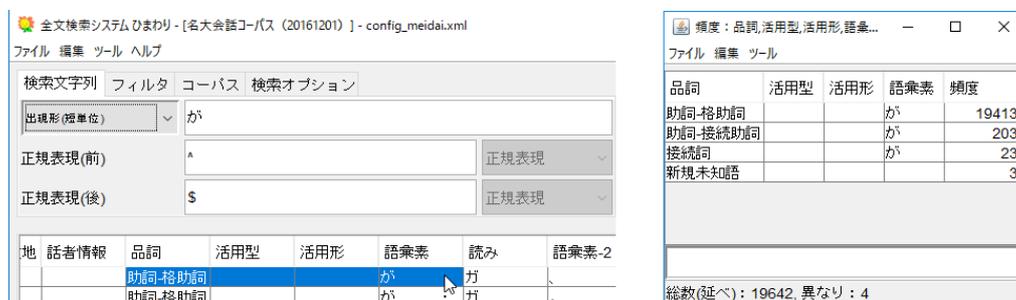


図 4: 検索時の一覧作成例

4.4 一覧作成機能の改善

4.4.1 結果の整形

検索結果の中には、そのままでは、一覧作成の際にうまく利用できないデータが含まれる場合がある。例えば、『ひまわり』用の『国会会議録』パッケージを使って、特定の検索語の経年変化を調べることを考える。このパッケージでは、検索結果として、検索文字列を含む会議の開催日が得られる。この結果に対して、年ごとに集計を行う場合、得られるのは開催日なので、年を抽出する必要がある。

このような問題を解決するために、検索結果に対する置換機能を追加した。置換は、図5のように、列ごとに行う。置換の条件式には、正規表現を使うことができる。図5では、月日の部分(例:-03-30)は不要なので、正規表現-.*にマッチする文字列を空文字列で置換するように条件を指定する。この置換結果に対して、次節で述べる結果の再集計を行うと、年ごとの集計ができる。



図 5: 置換機能の実行例

4.4.2 再集計

前節の置換結果のように、検索結果を再度集計しなければならない場合を考慮して、集計結果を再度集計する機能を実現する。再集計するには、検索時の一覧作成機能と同様に、再集計したい列を選択して、一覧を作成する。

図6左は、図5の「開催日」と「発言者」列を選択して、一覧を作成した結果である。このように、発言者ごとに検索文字列の出現頻度を経年変化を調べることができる。さらに、この結果から検索文字列の出現頻度の経年変化を求めるには、左図の「開催日」列を選択して、一覧を作成する(図6中)。この場合、左図の「頻度」の値を年ごとに合計する。それに対して、各年ごとの発言者の異な

り数を求めたい場合は、「頻度」を合計せずに「開催日」の頻度を求めればよい(右図)。元の一覧の頻度欄を考慮する・しないは、再集計するときを選択できる。

発言者	開催日	頻度
海部俊樹	1969	1
海部俊樹	1977	6
海部俊樹	1980	2
海部俊樹	1985	6
海部俊樹	1986	19
海部俊樹	1989	78
海部俊樹	1990	133
海部俊樹	1991	413
海部俊樹	1995	7
海部八郎	1979	13
海野三朗	1948	6
海野三朗	1953	1
海野三朗	1956	3

開催日	頻度
1947	488
1948	621
1949	793
1950	1011
1951	801
1952	1127
1953	1043
1954	771
1955	720
1956	549
1957	439
1958	605
1959	651

開催日	頻度
1947	160
1948	169
1949	176
1950	176
1951	159
1952	211
1953	227
1954	180
1955	158
1956	136
1957	108
1958	130
1959	136

図 6: 再集計の実行例 (左: 発言者ごとの経年変化, 中: 検索文字列, 右: 発言者数の異なり)

5 おわりに

本稿では、全文検索システム『ひまわり』の言語分析支援機能の拡張として、(1) 検索結果の集約を行う機能、(2) アノテーション結果から統計的分析の基礎データを収集するための機能を設計・実装した。

謝 辞

本研究は、国立国語研究所共同研究プロジェクト「大規模日常会話コーパスに基づく話し言葉の多角的研究」、および、科研費基盤研究(B)『「昭和話し言葉コーパス」の構築による話し言葉の経年変化に関する実証的研究』の一環で行われたものである。

文 献

- 山口昌也・田中牧郎(2005). 「構造化された言語資料に対する全文検索システムの設計と実現」 自然言語処理, 12:4, pp. 55-77.
- 小木曾智信・中村壮範・鈴木泰山・八木豊・山崎誠・前川喜久雄(2011). 「コーパス検索システム「中納言」デモンストレーション」 日本語コーパス完成記念講演会予稿集.
- Laurence Anthony (2016). *AntConc ver.3.4.4*. <http://www.laurenceanthony.net/>.
- 今井新悟・赤瀬川史朗・プラシャント・パルデシ(2013). 「筑波ウェブコーパス検索ツール NLT の開発」 第3回コーパス日本語学ワークショップ予稿集, pp. 199-206.
- 樋口耕一(2014). 『社会調査のための計量テキスト分析——内容分析の継承と発展を目指して』 ナカニシヤ出版.
- 山口昌也(2013). 「個人用コーパスの作成とアノテーションを支援する環境の実現」 第3回コーパス日本語学ワークショップ予稿集, pp. 369-372.