

筑波大学 情報学群 情報メディア創成学類

卒業研究論文

行列表現によるカテゴリデータの概観を
組み込んだ検索インタフェースの開発

長嶋 岳志

指導教員 三末 和男, 志築 文太郎, 田中 二郎

2015年1月

概要

本研究は2値で表現可能なカテゴリデータを条件とした情報の検索支援を目的とする。従来の検索システムには、条件の選択を繰り返し、検索結果を絞り込む方式がある。しかしそのような検索方式には問題点がある。例えば検索条件を絞り込み過ぎると検索ヒット数が0になり、何度も繰り返し検索する必要がある。また、検索者の要望と検索結果のトレードオフを考慮した検索が難しい。そこで、我々は問題点の解決のために、カテゴリデータの概観を検索インタフェースに組み込むことを考えた。カテゴリデータの概観を可視化するには名義尺度の尺度水準を持つ多変量データを扱う必要がある。Parallel Setsという可視化手法により扱えるが、線が重なりあうため特徴の読み取りが困難であった。そのため我々はParallel Setsと行列表現を直接連結させ、カテゴリデータの概観を組み込んだ検索インタフェースを開発した。そして条件を絞る際に、検索件数が0にならないようにした。また検索者の要望と検索結果のトレードオフを考慮した検索結果を求めることを可能にした。

目次

第1章 序論	1
1.1 ファセット検索	1
1.2 ファセット検索の問題点	1
1.3 可視化する理由	2
1.4 本検索システムが可能にすること	2
1.5 2値データと非2値カテゴリデータを含むデータセットを可視化する場合の問題点	3
1.6 目的とアプローチ	5
1.7 貢献	5
第2章 関連研究	6
2.1 視覚的にファセット検索できるインターフェース	6
2.2 多次元データを対象とした可視化手法	6
2.2.1 多次元データにおける変量間の関係性を分析するための手法	6
Parallel Coordinates Plot	6
Parallel Sets	7
2.2.2 2値表現可能なデータの可視化手法	7
Onset	7
2.2.3 2値表現で表されるデータと表されないデータ間の分析を行えるツールに関して	7
第3章 情報の検索のための要件の整理	8
3.1 検索手順の整理	8
3.1.1 2値データの絞り込み	8
3.1.2 非2値データの絞り込み	9
3.2 検索支援に必要な要件	9
3.2.1 非2値カテゴリデータに付随する2値データの可視化	9
3.2.2 条件の組み合わせ方の変更がもたらす2値データと非2値データの変量間の関係性の変化の可視化	10
第4章 インタフェースの設計	11
4.1 対象データ	11
4.1.1 データ構造	11
4.2 非2値データの可視化	11
4.3 2値データの可視化	12

4.3.1	行列表現の設計に関する要件整理	12
4.3.2	行列表現の設計 1	13
4.3.3	行列表現の設計 2	14
4.3.4	行列表現の設計 3	14
	設計の方針	14
	設計の方法	16
	結果	16
	未選択の条件	16
	ハイライト	16
	色	17
第 5 章	実装	18
5.1	MainView	18
5.1.1	選択の幅を広げるための条件の Recommend	19
5.2	ResultView	19
第 6 章	ユースケース	20
6.1	アルバイトの求人検索	20
6.1.1	検索者のモデル設定	20
6.1.2	利用シーン	20
6.2	宿泊先データを用いた例	28
6.2.1	検索者のモデル設定	28
6.2.2	使用データの説明	28
6.2.3	宿泊所検索の利用シーン	28
第 7 章	結論	33
	謝辞	34
	参考文献	35

図目次

1.1	2 値データを PCP を用いて可視化した例	3
1.2	2 値データを Parallel Sets を用いて可視化した例	4
4.1	データ構造	12
4.2	Parallel Sets の部分	13
4.3	設計 1	15
4.4	行列表現の部分	17
5.1	ツールの概観	18
6.1	アルバイト検索：初期画面	21
6.2	アルバイト検索：未経験者歓迎を抜いたとき	22
6.3	アルバイト検索：未経験者歓迎に加えて駅近・駅中の条件を抜いたとき	23
6.4	アルバイト検索：4 時間以内と未経験者歓迎，駅近・駅中を抜いたとき	24
6.5	アルバイト検索：未経験者歓迎と外国語を追加	25
6.6	アルバイト検索：週 2 と未経験者歓迎，駅近・駅中を抜いたとき	26
6.7	アルバイト検索：外国語と未経験者歓迎を追加	27
6.8	宿検索：初期画面	29
6.9	宿検索：高評価を抜いた図	30
6.10	宿検索：高評価を抜かないときの図	31
6.11	コンビニと温泉を抜いた図	32

第1章 序論

世の中には様々な検索方式が存在する。本研究はその中でも、あらかじめ整形されたデータから特定のレコードを絞り込む検索方式に焦点を当てる。Web 上には場所や時間、その他詳細な条件を入力として情報を検索するサービスが数多く存在する。サービスの例として商品や住宅、仕事等を扱う検索サイトがあげられる。本論文ではアルバイト求人の検索を説明の例に取り上げることとする。アルバイトの求人情報を探す際に、働く地域や時給、その他複数の 2 値で表現可能な条件から検索キーワードを絞ることによりアルバイト先の情報を見つけることが可能である。ここで述べる 2 値で表現可能な条件とは、アルバイト先が駅に近いかな否かや、学生を歓迎するかな否かといった、YES または NO の 2 値で表現できるものとする。検索者は条件を絞り込み過ぎてしまったり、様々な条件を組み合わせで情報を探す。そのため、何度も条件を追加したり、削除することを繰り返しながら、自分の条件に満たす情報を探す。この検索方法は一般的にファセット検索 [1] と呼ばれ、幅広く使用されている。

1.1 ファセット検索

ファセット検索とは、あらかじめ用意された項目を選択することにより、検索結果の絞り込みを可能にした検索方式である。サイト運営者が検索者にとって有益と思われる条件をあらかじめ用意すれば、検索者は条件をテキスト入力せずに、タブやラベル等を選択するだけで検索結果を絞り込むことが可能である。ファセット検索に用いられるデータセット内には各レコードにおいて、2 値で表現できるデータ (2 値データ) と働く地域や給料のように 2 値で表現できない値 (非 2 値データ) が用意されている。尚、前述のレコードとは複数のフィールドより構成される 1 件分のデータのことを指す。本研究では 15 変数までの 2 値データを含むデータセットを対象に、ファセット検索をする際に生じる問題点を考慮した検索インタフェースを開発する。

1.2 ファセット検索の問題点

従来のファセット検索は、検索者が条件を表す項目を選択し、広い範囲から狭い範囲へと条件を絞って情報を探索することが一般的である。項目の内容としてはデータに付加される情報、例えば価格帯や種類が含まれる。また検索の際に複数のタグにチェックを付け、検索結果を絞り込む方法が広く利用されている。たくさんのサイトで使用されているファセット検索だが問題点として以下の 3 点があげられる。1 つ目は条件の組み合わせごとで、検索結果数を知ることが出来なかったり、検索結果の比較ができない点である。条件毎の検索ヒット数に関して知ることができたり、比較できれば、情報を検索する際に条件を絞り込み過ぎることはなくなり、検索結果が 0 件になることを防ぐことが可能になる。さらに検索ヒット数が 0 になる条件の組み合わせか否かについては、実際に検索してみなければ知る

ことができない。そのため、検索ヒット数0の検索を何度も行ってしまう可能性がある。他にもデータ全体を概観できないと、条件の組み合わせを探す際に何度も検索する手間があるのに加え、見落としてしまう条件の組み合わせが出てくるとも想定される。2つ目は、情報を検索する際に視点を変え検索対象を広げることが難しい点である。これは、一般的なファセット検索では情報を絞り込む方向へナビゲートすることが起因しており、検索者が条件の組み合わせ方を狭めてしまうことが考えられる。3つ目は、通常ファセット検索では、変量間の関係性を認知できないため、検索条件の組み合わせを変更することが、検索結果にどのような変化をもたらすか知ることができない。そのため、どの検索条件を選択すれば良いかヒントを得られず、検索者は自身の要望と検索結果のトレードオフを考慮して検索することが難しい。尚、本論文における変量とはデータの特徴を表す属性を意味している。

1.3 可視化する理由

検索者が必要とする情報を抽出するための方法として、検索履歴をもとにデータマイニングを行う方針も考えられる。しかしデータマイニングの技術を向上できたとしても検索結果としてレコードのみが出力されるため、カテゴリデータ全体を概観することや変量間の関係性を知ることはできない。データ全体を概観できれば、検索によって得られる情報の偏りや検索ヒット数を事前に知ることができる。また、検索結果の概観とインタラクティブな操作により、トレードオフを考慮しながら自分に必要な情報を得ることが可能となる。そのためファセット検索の問題点を解決するには、視覚的なアプローチをとらなければ難しいと考えられる。よって検索においてデータの可視化が効果的といえる。

1.4 本検索システムが可能にすること

1つ目は、カテゴリデータを概観し、2値データの要素と非2値データの要素間の関係性を読み取れるようにすることである。このことが可能になると条件を絞り込む前に、おおまかな検索ヒット数が分かるため、ヒット数が0になる検索を行うことはなくなる。2つ目は、条件ごとでレコードの分布の偏りを読み取れるようにすることである。偏りを読み取ることにより条件の特徴について知ることができる。3つ目は検索条件の選択を支援することである。検索者の意向に合うように、行列表現のレイアウトを最適化する。つまり、検索者が同じ優先度の条件が複数ある中で検索を絞り込む際に、どの条件を選ぶと検索にヒットする可能性が高くなるかを提示できるようにする。4つ目は条件を追加・削除すると、どのように非2値の変量との関係性が変わるか読み取れるようにすることである。例えば検索者がアルバイトの求人情報を探すときに、条件を変えることでどのように時給が上下するかを知ることができたとする。そうすれば、検索者自身が求める時給と条件の組み合わせ方の間で、検索結果に対してどのような影響を与えるかについて考えながらアルバイトを探せるようになる。

1.5 2 値データと非 2 値カテゴリデータを含むデータセットを可視化する際の問題点

多変量データにおける変量間の関係性を可視化する表現として Parallel Coordinate Plot (PCP) [4] や Parallel Sets[2] がある。PCP は、各変量の座標軸を水平に配置し、座標軸上に全てのレコードをプロットするとともに、同じレコード間を線で結ぶことにより描画される。そして、線の密集具合を見ることにより多変量データの分析を行うことを可能にする可視化手法である。図 1.1 がその例である。データはアルバイトの求人情報に関するものである。非 2 値データとしてアルバイト先の職種、時給データを使用している。またアルバイト先の労働条件に関する項目を 10 個用いた。図 1.1 の示す PCP は本来、順序尺度や間隔尺度、比例尺度の尺度水準を持つデータに対して用いる手法だが、PCP により、変量間の関係性を見ることができかどうかを確認するためにアルバイトの求人データを PCP に強引に当てはめ描画した。その結果、図 1.1 のように 2 値をプロットする際に点が集中してしまうため線が重なり合ってしまうことが分かる。そのため 2 値データと非 2 値データ間の関係性を知ることはできない。図 1.1 をもとに説明すると職種または時給と各条件との関係性を知ることは不可能である。また、線が一箇所に密集しているため条件の組み合わせを変更したときにどれ程に時給が上がるのか知ることができない。よって PCP では条件を追加したり削除するときに非 2 値データとの関係性がどのように変わるか知ることができない。

次に Parallel Sets における問題点について述べる。Parallel Sets[2] は PCP と Mosaic Displays [3] を複合した可視化手法である。カテゴリデータ、つまり名義尺度の尺度水準を持つデータの変量間の関係性を可視化する際に用いられる。図 1.2 のように長方形の面積によって度数を表し、次元ごと階層式にしきつめることにより多変量データの分析を可能としている。図 1.2 を見ると Parallel Sets において、カテゴリデータの変量数が増加すると、変量間の関係性を視覚的に表現することが困難になることが分かる。これは変量数が増加することにより線の密集度が増加し、視覚的混雑度も増加してしまうためである。

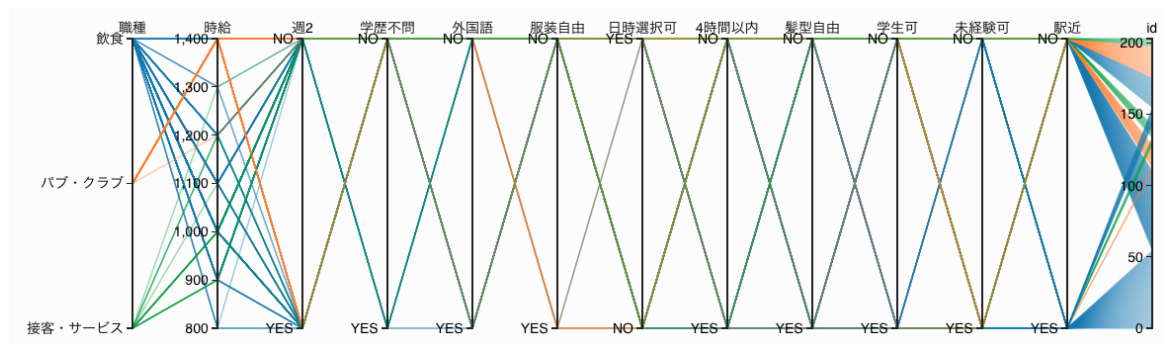


図 1.1: 2 値データを PCP を用いて可視化した例

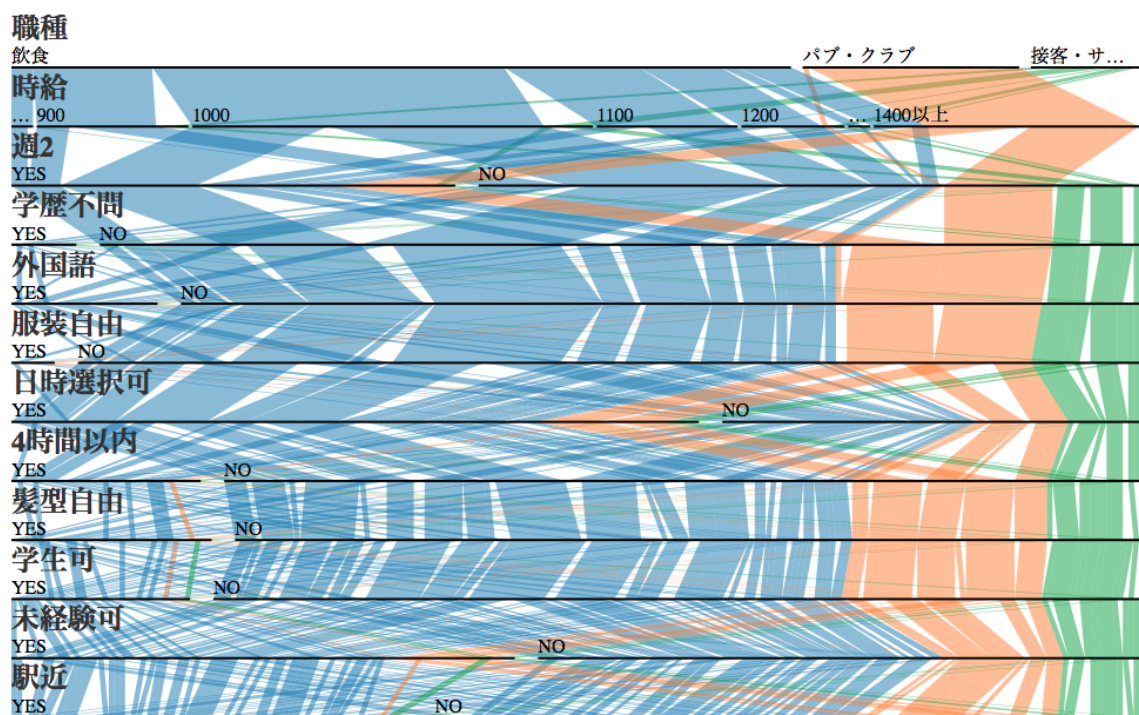


図 1.2: 2 値データを Parallel Sets を用いて可視化した例

1.6 目的とアプローチ

本研究では、2 値表現が可能なカテゴリデータを対象とした情報の検索支援を目的とする。検索者が多くの 2 値データを含むカテゴリデータ全体を概観することで検索にヒットする条件を発見できるようにする。また、検索条件を追加または削除しながら、検索者の要望と検索結果におけるトレードオフを考慮した検索を可能にする。アプローチとしては、Parallel Sets と行列表現を直接連結し、似た検索条件を近くに配置することでデータの概観を可能にする。また、検索条件全てに該当するレコードはハイライト表示することで検索者が求めている情報を視覚的に提示できるようにする。

1.7 貢献

貢献は以下の 2 点である

- Parallel Sets と行列表現を連結し、2 値データと非 2 値データを含むカテゴリデータを概観できる表現手法を開発した点。
- 上記の可視化手法を用いて検索を支援するインタフェースを開発した点。検索結果が 0 件にならないように検索を支援した。また検索者の要望と検索結果におけるトレードオフを考慮した検索を可能にした。

第2章 関連研究

本研究の関連研究について述べる。関連研究の分類の仕方として、大きく二つに分ける。一つ目にファセット検索を行うインターフェースの研究、二つ目に多次元データを扱う可視化手法の研究に関して述べる。本研究では、2 値データと非 2 値データの関係性を知るための可視化手法を開発することでファセット検索を俯瞰して行えるようにすることを目的としている。そのため、変量間の関係性を分析するための手法や 2 値データを分析するための手法、ないしは複数の可視化手法を統合する研究について関連研究として取り上げる。

2.1 視覚的にファセット検索できるインターフェース

Hearst はファセット検索におけるデザインのガイドラインを作成し、それをもとにインターフェースを開発した。[6]Hearst はカテゴリごとの階層構造を持つファセットメタデータの特徴を活かし、ドリルダウンしながらより詳細な情報へと探索できるようにした。また Clarkson らは Tree map を用いて検索結果を俯瞰できるようにした。[5] この研究により、検索するためのキーワードを入力し、もともと木構造でタグ付けられたドキュメントを Tree map 上に描画した。検索ヒットしたドキュメントが属するカテゴリを見てどのサイトを閲覧するか決めることが可能になった。一方 Hansaki らは、検索条件の組み合わせ方を決定する支援をするために、検索者がデータフロー図を組み合わせながら検索結果を絞り込む検索インターフェース Find Flow を開発した。[9] 検索者は、検索条件の組み合わせ方を考えるときに、新しい検索結果を元にいろいろな条件で検索を繰り返し、試行錯誤しながら必要な情報を手に入れる。Hansaki らは、このような検索行為の支援をするために、複数の検索タスク間を行き来したり、検索タスク間で検索クエリを共有したりするといった情報探索過程における操作を可能にした。

2.2 多次元データを対象とした可視化手法

2.2.1 多次元データにおける変量間の関係性を分析するための手法

Parallel Coordinates Plot

Parallel Coordinates Plot(PCP)[4] は、次元を表す垂直な座標軸を 平行に並べることで多次元データを表現する手法である。垂直な座標軸上にデータと対応する点を配置し、各軸上の同じレコードである点同士を線で繋ぎ、1 レコードを 1 本の線として表現する。このような表現により分析者は各次元のデータ分布の一覧や、線の傾きから次元間の関係性を把握できる。しかし、2 値で表されるデータをパラレルコーディネートで表現すると座標軸上の 2 箇所に点が集中してしまう。そのため、2 値表現

可能なデータが隣接して並んだとき、線が一部に密集してしまう。それが原因で視覚的混雑度が増してしまうため各レコード間の関係性を知ることができなくなる

Parallel Sets

Parallel Sets[2] は、PCP と Mosaic Display [3] を組み合わせることで多変量データを表現する手法である。PCP の垂直な座標軸上にプロットされた点をカテゴリごとに分類し矩形として表現する。さらに、矩形間を幅を持った線としてつなぐことで複数の属性間の関係性を読み取ることが可能にしている。Parallel Sets を用いて 2 値を表現することも可能でだが、次元数が増加すると組み合わせは指数関数上で増加してしまう。その結果次元数が増える程、線同士が多く重なり合い視覚的混雑度が増すため変量間の関係性を読み取ることが困難になる。

2.2.2 2 値表現可能なデータの可視化手法

Onset

Kim らは、マトリックス中に 2 値データの有無を表すセルを配置し、セット同士を重ねることで共通点と相違点を可視化できるようにした [7]。またセット間を幅のある線で結び、線の太さで類似度の大きさを表現することも可能である。この可視化手法は 2 値で表現される変数の数が多いときに有効だが、2 値データを含む変量と非 2 値データを含む変量間の関係性を分析することはできない。なぜならば、マトリックスを用いて非 2 値データを表現できないためである。

2.2.3 2 値表現で表されるデータと表されないデータ間の分析を行えるツールに関して

Domino[8] はデータに対して様々な可視化手法を包括的に適用させることで、データを柔軟に配置や結合、抽出することを可能とした。この研究によりデータ構造の特性に柔軟に対応したデータの可視化が可能になった。このツールを使用すれば Parallel Sets と行列表現を結合させ非 2 値データと 2 値データの関係性を見ることが可能である。しかし、マトリックスの仕様を変更することはできないため、より多くの条件に当てはまるレコードを直感的に識別できるように設計することはできない。また、条件をインタラクティブに抜いたり、追加することはできない。

第3章 情報の検索のための要件の整理

本章では情報を検索する際に支援すべきことについて整理する。まずアルバイトデータを例に出しながら、求める情報を見つけるまでの検索手順を整理する。次に検索手順を踏まえ、検索の支援に必要な要件を列挙する。

3.1 検索手順の整理

一般的にファセット検索を用いて情報の検索を行う際、最初に検索条件の作成を行う。検索条件の作成方法には大別して二つの方法があげられる。一つ目は、条件を検索に含めるか否かを選択する方法である。例えば、従来の検索方式ではチェックボックスにチェックマークを入れて選択の有無を表現している。(2 値データの絞り込み) 二つ目は固有名詞や数値の範囲を表すカテゴリを選択する方法である。(非 2 値データの絞り込み) この二つの方法について以下で例を用いながら説明する。検索条件を作成する際は、まず検索者自身が強く必要とする条件を選択する。このとき検索件数が多い場合は優先順位の低い条件を選択していき、徐々に検索結果を絞り込んでいく。検索結果が 0 件になってしまった場合は条件を緩めるために選択済みの条件を削除する。この操作を繰り返し、検索者が十分と思う条件が見つかった後は、検索結果に列挙された情報の詳細を確認する。

3.1.1 2 値データの絞り込み

2 値データとは、YES や NO などの 2 値で表現可能なデータのことを示すとする。また条件とは 2 値データで表現されるカテゴリデータの変量を意味している。2 値データの例としてアルバイトのデータを用いて説明する。アルバイトのデータセットには、例えばアルバイト先が交通費を支給するか否かを YES か NO の 2 値で表現する変量がある。あるアルバイト先が交通費を支給する場合は表 3.1 のように YES と記録されている。検索者は交通費支給の条件にチェックマークを入れて検索にかけたとき交通費支給が YES となっているレコードのみを抽出することになる。

表 3.1: アルバイトデータ

ID	職種	時給	日時選択可
1	飲食	1,000 円台	YES

3.1.2 非 2 値データの絞り込み

非 2 値データとは場所や時間、金額等のように 2 値で表現できないデータを表す。例えば、検索条件として場所をセットするときは、地名やエリア名のような固有名詞を選択し検索条件に反映させる必要がある。アルバイトデータを例に出すと場所とは勤務先の地域名を指す。検索者は自分が働くことが可能な地域を選択し、その中に含まれる地域からアルバイトの求人情報を絞り込むことを行う。また、検索条件として時間や金額のような幅を持つ変数をセットするときはカテゴリ化された時間の範囲を選択したり、スライダーを使用して時間の範囲を調整し検索条件に反映させる方法がある。例えば今回ユースケースで使用するアルバイトデータにはそれぞれのレコードに対して週何日間働くことが可能かというメタデータが付与されている。またアルバイトデータにおいて金額を表すデータとしては時給があげられる。金額の範囲をカテゴリ化して、900 円以下、900 円から 1000 円、1100 円から 1200 円、1200 円から 1300 円、1300 円から 1400 円のように料金を範囲ごとに分けて表現することが可能である。検索者は上のようにカテゴリ化された時間や時給データを検索条件として、検索を行うと考えられる。

3.2 検索支援に必要な要件

上記の検索過程を踏まえて検索支援について必要な要件を整理する。

3.2.1 非 2 値カテゴリデータに付随する 2 値データの可視化

1 つ目の要件は非 2 値カテゴリデータに対応する 2 値データが分かるようにすることである。検索における問題点を述べた後にこの要件が必要な理由を説明する。従来の検索方法では、優先度が高い順で条件を選択し、検索結果を絞り込む形をとる。しかし優先度が高い条件から順に条件を選択していくと、条件の絞り込み過ぎから検索ヒット数が 0 になってしまうことがある。しかも条件の数が多いと、条件の組み合わせの数は指数関数状で増加するため網羅的に検索をすることができないと考えられる。また、選択した条件ごとに対して何度も検索をしていると、条件の組み合わせの見落としが起きる可能性も高まり、検索条件を選び損ねてしまうことがある。これらの問題点を解消する方法として、検索結果を概観することが必要である。そして、どの条件の組み合わせであれば検索がヒットするか分かるようにしなければならない。そのため優先度の高い条件を選んだ後に、非 2 値カテゴリデータから成る要素に対してどのように 2 値データに対応するか知ることは重要である。アルバイトデータを例に出すと優先度が高い条件として、1 日の内で 4 時間以内の労働が可能か否かや、学生が歓迎であるか否か、労働時間を選択できるか否かを選んだ後に、検索者は次にどの条件を選択しようか迷うことが考えられる。そんなときに、データ全体を概観し、目をつけている非 2 値データの組み合わせに対してどの条件が他に該当するか知ることができれば、2 値で表される条件を一つ一つ追加したり削除して条件の組み合わせをチェックする手間が省ける。以上の観点からも非 2 値カテゴリデータに対してどのような 2 値条件がどれ程対応しているか知ることは重要である。

3.2.2 条件の組み合わせ方の変更がもたらす2値データと非2値データの変量間の関係性の変化の可視化

2つ目の要件は条件を追加・削除すると、どのように非2値の変量との関係性が変わるか読み取れるようにすることである。例えば非2値データにアルバイトの時給があり、2値データに複数の条件が存在するときを想定する。時給と、ある条件の優先度が同じとき、検索者はどちらを優先にしようか迷うことが考えられる。そのため、検索者は条件と時給のどちらを重視するか考えながら時給の変位を見ることで時給と条件のトレードオフを考慮したアルバイト選びを可能にする。つまり条件を選択するかしないかでどれほど非2値データに対して2値データが含まれるか分かるようにする必要があると言える。

第4章 インタフェースの設計

本研究の検索インタフェースに対して適応可能なデータ構造と、非2値データ部分の可視化設計、2値データ部分の可視化設計、それに加えツール全体の設計について記述する。

4.1 対象データ

本節では、アルバイトのデータを例に取り上げ、対象とするデータの説明をする。対象のデータについて説明をした後は、2値データと非2値データを一つにまとめたデータ構造について述べる。

本インタフェースでは表のようなデータセットを扱っている。データセットは非2値データを含む変量と2値データを含む変量をそれぞれ持つものを対象としている。例えば表4.1は非2値データに職種と時給を持つ。また2値データとして日時選択可や駅近・駅中、学生歓迎、服装自由、未経験歓迎を持つ。

表 4.1: アルバイトデータ

ID	非2値データ		2値データ				
ID	職種	時給	日時選択可	駅近・駅中	学生歓迎	服装自由	未経験歓迎
1	飲食	1,000 円台	YES	YES	NO	YES	NO
2	接客	1,200 円台	NO	YES	NO	NO	NO

4.1.1 データ構造

非2値データと2値データの両方を含むデータセットのデータ構造は図4.1の様に木構造にて表現する。上層に非2値データを、下層に2値データを配置する。尚、レコードが無い場合はノードから分岐しないようにする。

4.2 非2値データの可視化

本インタフェースでは非2値データに関しては Parallel Sets を使用し2値データには行列表現を使用する。非2値データの値を読み取るための可視化手法は従来通り Parallel Sets や PCP を用いればよい。しかし1.4節で挙げた問題点にあるように2値データを扱うと線が重なり合うため、グラフから情報を読み取ることが困難になる。よって非2値データと2値データの表現手法を分けて可視化設計を行

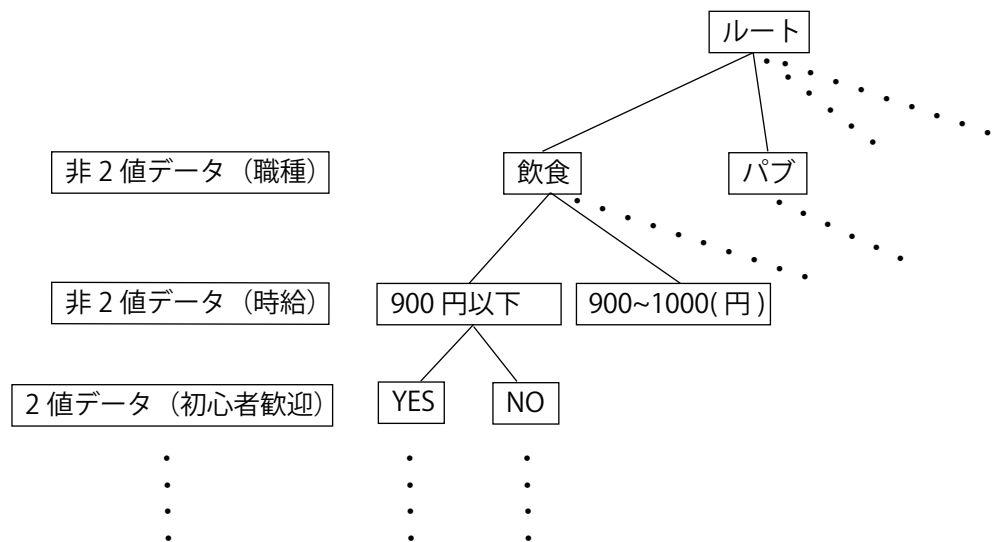


図 4.1: データ構造

う．非 2 値データを可視化する手段として結果的に Parallel Sets を使用した．Parallel Sets はカテゴリ間の結びつきを一本の線で表現できるためである．行列表現と接続する際に線同士が重なったとしても、どのカテゴリから線が延びているかすぐに知ることができる．一方 PCP を使用しなかった理由は、PCP を適用するとレコード毎で線が独立してしまい、線が交差したときに見難いからである．ただし、Parallel Sets を使用したとしても、非 2 値データの 변수数が多くなると組み合わせ方が増え、行列表現と接続箇所において視覚的混雑度が増加してしまう．そのため非 2 値データの対応 변수数は 2 から 3 変量とする．

4.3 2 値データの可視化

この節では、行列表現の設計に必要な要件を整理した後に、2 値データと非 2 値データを直接繋ぐ可視化手法の設計に関して、現在の設計に至った経緯を織り交ぜながら説明する．最初に 2 値データの配列パターンから行と列の類似度を求めて行列表現を設計したケースについて述べる (行列表現の設計 1)．次に行のみの類似度を考慮した配列方法について説明し (行列表現の設計 2)，最後に Parallel Sets と行列表現を直接連結させた際の設計に関して紹介する (行列表現の設計 3)．

4.3.1 行列表現の設計に関する要件整理

行列表現の設計に関して以下に述べる 3 点をおさえた手法を考える必要がある．1 点目は、Parallel Sets と接続する箇所において、できるだけ線が重なり合わないようにすることである．Parallel Sets からたくさんの線が行列表現に対して伸びればその分 2 値データと非 2 値データの関係性を知るのは難しくなる．3 章で述べたようにトレードオフを考慮した検索ができるようにするためにも Parallel Sets

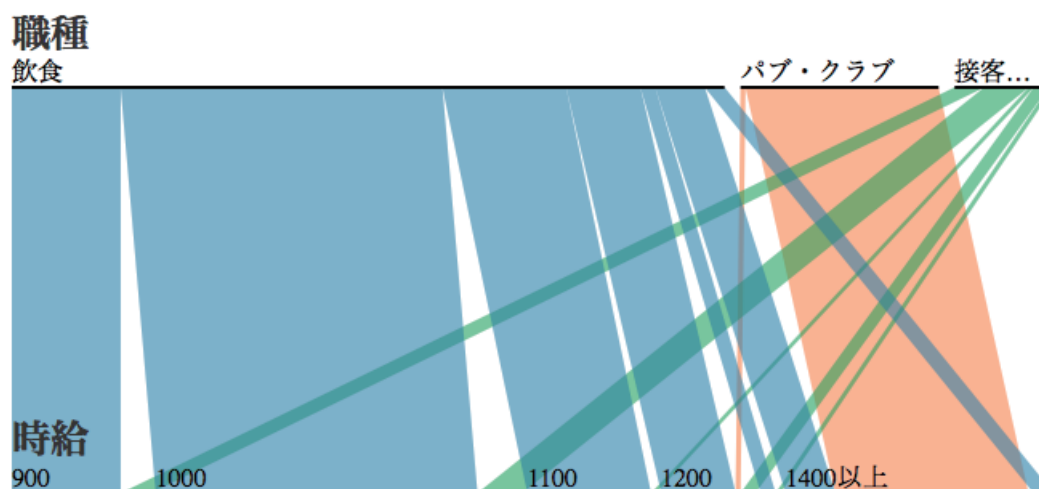


図 4.2: Parallel Sets の部分

から伸びる線は可能な限り少なくする必要がある。2点目は、同じ条件の組み合わせをできるだけ近くに配置することである。3.2.1節で非2値カテゴリデータに対してどのような2値データが対応するか知ることが重要であると述べた。この要件を満たすために、似た条件の組み合わせ方を持つレコードが近くに集まるように行列表現の可視化設計をする。そして非2値データに対応する2値データを見やすくする。3点目は、条件の選択を変更したときに、全ての条件に該当するレコードの数がどのように変化したか分かるようにすることである。3.2.2節で条件の組み合わせ方を変えたときの、非2値データと2値データの関係性の変化を知ることができるようにすることが重要であると述べた。このことから分かるように非2値カテゴリデータの集合に対して全条件に該当するレコードの数がどのように変化するか視覚的に認知できるように工夫する必要がある。

4.3.2 行列表現の設計1

設計1では、行と列に対して評価式を用いて最適な配置方法を算出する。この設計方法では、行と列において条件の組み合わせが近いもの同士を近くに配置することを重視する。次に設計手順について説明する。まず1行ずつ適当にレコードを並べる。尚、列は各2値条件を表している。次にランダムに列を配置し、評価式から値を計算する。得られる値が閾値より小さい値が出るまで同じ操作を繰り返し行う。列において閾値より小さな値が得られたら、列の配置を固定したまま行に関しても同じ操作を行う。行に関しても閾値を下回れば行と列の整列を終了する。

評価式の求め方について以下に記す。各列に YES, NO などの2値で表現できる条件をランダムにセットし、各行にもカテゴリデータのレコードを一行ずつランダムにセットする。まず各列をベクトルと見立てると以下ようになる。

$$\vec{v} = (v_1, v_2, \dots, v_m)$$

$$v_i = 0 \text{ または } 1$$

次に全てのベクトルから 2 つのベクトルを抽出し内積をとったものにたいして arccosine で類似度の計算をする．さらに表上における二つのベクトルの距離を求め類似度を算出したものに掛ける．この操作を全てのベクトルの組に行い総和を求める．評価式は以下のようである． r は

$$\vec{v}$$

の表上における位置を表し，関数 d で r 間の距離を計算している．

$$\sum_{j=0}^{c-1} \sum_{i=j+1}^c d(r_i, r_j) * \cos^{-1}(\vec{v}_i \cdot \vec{v}_j)$$

以上の操作によって得られた行と列の位置関係を図示すると図 4.3 のようになる．この図を見ると列に関してはレコードの数が似ている列が近くに配置されているように見えるが，明らかに条件の組み合わせが似たレコード同士が近くに配置されている訳ではないことが分かる．そのため上記における 2 点目の要件を満たさないため，別のレイアウト方法を考えた．

4.3.3 行列表現の設計 2

設計 2 については，多くのレコードを含む条件程，Parallel Sets との接続箇所から近くに配置した．また各レコードにおける条件の組み合わせ方を数値化し数値が高いもの程上に配置する．例えばあるレコードが [1,0,1,0,1,1,0,1,0] という条件の組み合わせ方をしているとすると．そのときこのレコードを 1010111010 と 2 進数で表し数値が大きいレコード表の上に配置する．

そのように設計することで左上から右下に向かって似た条件の組み合わせごとに分布させることが可能になる．しかしこの配置方法は Parallel Sets との連結が難しい．レコードは非 2 値データの条件の組み合わせが同じもの同士でグルーピングすることは全く考えていないため Parallel Sets と行列表現を連結する線が様々なところで交差すると考えられる．そのため上記における 1 点目の要件を満たしていないと見なした．

4.3.4 行列表現の設計 3

設計の方針

設計 3 は設計 1・2 と異なり Parallel Sets の末端部分の線が分岐する箇所と行列表現を直接接続する．そのように設計すると Parallel Sets と行列表現の間に線を用いる必要がなくなる．次に行列表現内における設計の方針について説明する．親ノードのうち，条件が該当するか否かで線を分岐させることを繰り返す．そのようにすることで Parallel Sets と行列表現を直接連結することができる．また非 2 値データの組み合わせが同じであれば，同じ条件の組み合わせは隣接して配置することができる．さらに選択条件を全て満たす場合においては必ず一番右側にレコードが集中するためハイライト表示すれば，非 2 値データに対する位置関係が分かりやすくなる利点もある．



図 4.3: 設計 1

設計の方法

最初に、2 値データの行の配置方法について述べ、次に列の配置方法について説明する。行の配置は条件に当てはまるレコード数が多い条件程、上に配置するように設計した。なぜならば、つながりを持ったノードを多くすることで条件に該当するか否かを見分けやすいように分離するためである。

次に列の配置方法は、図 4.4 のように親ノードに対して条件に該当するか否かを判定し、該当するレコードは、親ノードの右側にまとめる。尚、条件に当てはまるレコードには非 2 値で使用されている色と同じ色を用いて色付けする。この操作を 2 値を持つ変数全てに対して繰り返し行う。

結果

結果的に上に示した 3 つの行列表現の中で設計 3 が要件を一番満たしていると判断した。非 2 値データと 2 値データが直接連結されており、Parallel Sets と行列表現を線で繋ぐ必要がなく視覚的混雑度が増さないためである。また、該当条件が多い条件を Parallel Sets の近くに配置することで、2 値の分布を可能な限り値ごとに分けて配置できている。その上、非 2 値カテゴリデータの組み合わせ方が同じものに対しては、2 値データの条件の組み合わせが同じレコードは隣接して配置される。また、似た条件の組み合わせも近くに配置される傾向があるため、この設計方法を採用することにした。尚、2 値データの対応次元数は 15 以下とする。15 次元までとした理由は、検索者が 1280×800 程度の画面領域で検索することを考えたときに行列表現で 15 行分まで表示することができるためである。

未選択の条件

選択された条件を上部に、選択されていない条件を下部に 2 分して配置する。選択されていない条件に関しては、検索者が未選択であることが視覚的に分かるようにグレーで着色した。初期画面ではどちらの領域も、条件に当てはまるレコード数が多い条件程、上に配置するように設計した。なぜならば、つながりを持ったノードを多くすることで着色部分と非着色部分をできるだけ見分けやすいように分離するためである。また非 2 値データを含む変数の範囲をを絞って、条件に当てはまるレコード数が多い順でソートすることも可能にした。アルバイトデータを例に出すと検索者が求める時給以上の価格の中で、当てはまるレコード数が多い条件を上配置することが可能である。この機能によって選択する非 2 値データの変数に合わせて見やすさを最適化することが可能になった。

ハイライト

選択条件を全て含むレコードはハイライト表示することで選択条件に全て当てはまるレコードをすぐに見つけられるように設計した。色は元の表上の明度の変化を大きくしている。検索者は条件を変える度にハイライトされる箇所や範囲が変化する様子を見て条件を追加または削除したときの非 2 値データの 2 値データの関係の変化を読み取ることが可能である。

色

Kovalerchuk らの研究では 2 値データの分布を比較しやすくするために、白と黒の 2 色を使用し、HanselChain のアルゴリズムを適用させ 2 値の境界をはっきりさせるためのレイアウトを提案した。[10] 本研究においても Parallel Sets で使用された色と白を用いて行列表現で 2 値データを描画する。カテゴリごとに色分けする際にカラースペース上で離れている色を使用した。一般的にカテゴリを区別するために赤、緑、青、黄色を使用することが効果的である。[12]

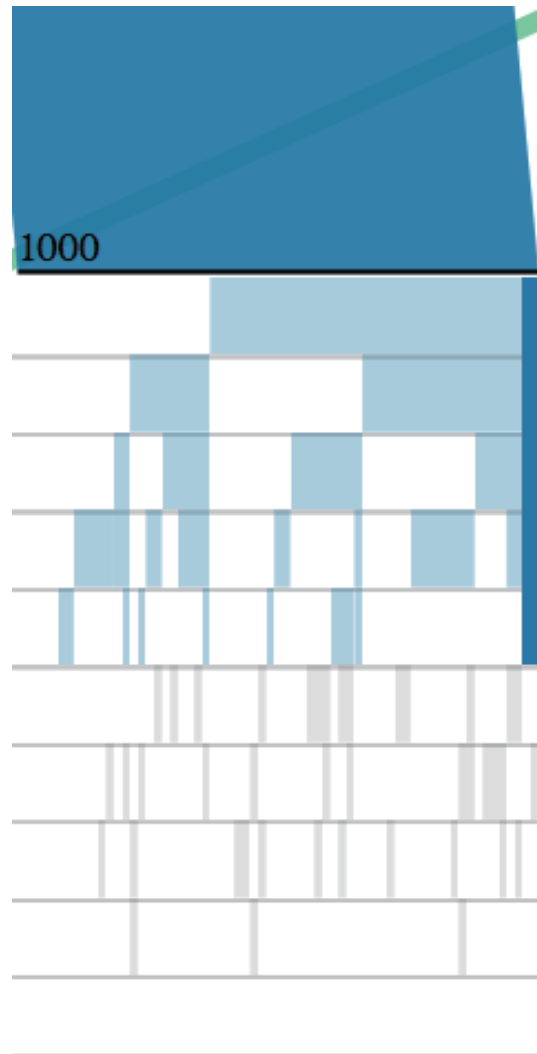


図 4.4: 行列表現の部分

第5章 実装

web サービスとして、ファセット検索が多く用いられているため、本ツールは様々な web サービスに適応できるように JavaScript を使用した。また図 5.1 の左側のビューを MainView、右側のビューを ResultView と呼ぶ。本章では2つのビューに関して説明する。

5.1 MainView

Parallel Sets 部分を MainView の上方 (1)、行列表現部分を下方 (2) に配置した。Parallel Sets 部分の実装に関しては D3.js で配布されているオープンソースを使用した。行列表現部分に関しては Parallel Sets を拡張した。また2値条件の選択をするためのタブを行列表現の行として配置した。検索者が条件の選択を行う際、図 5.1 の (3) のタブをクリックし、選択または選択解除を行う。未選択のタブがクリックされるとボタンの明度が下がるように実装した。そうすることにより検索者にとってタブの選択状態が分かるようになった。

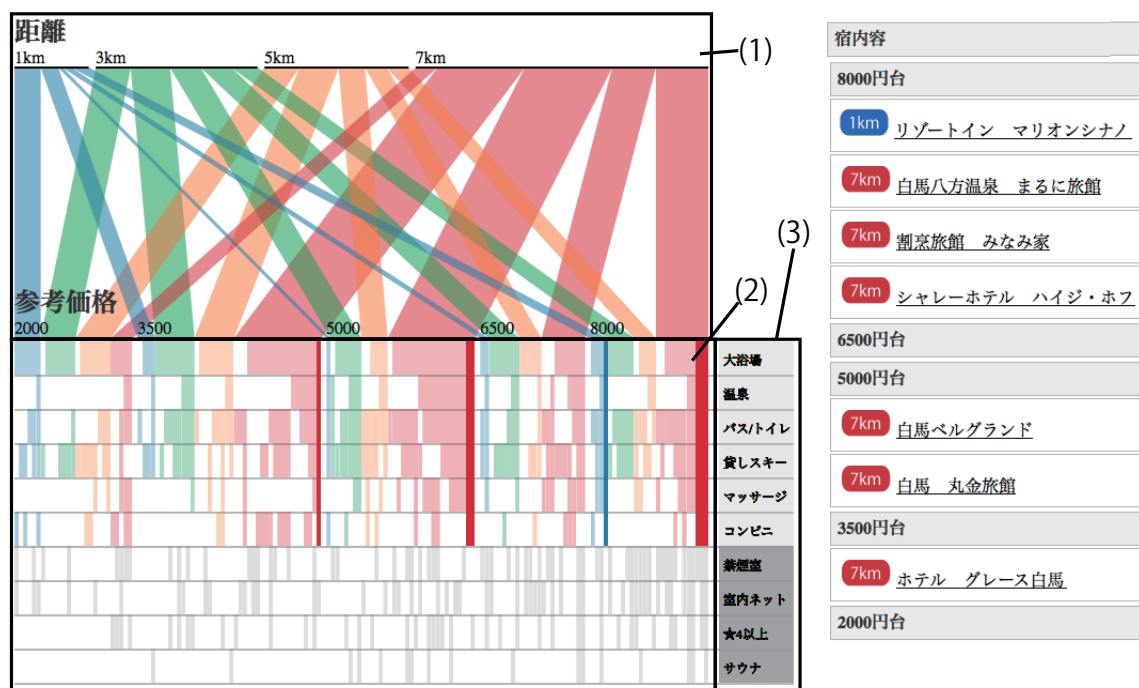


図 5.1: ツールの概観

5.1.1 選択の幅を広げるための条件のレコメンド

検索者は、条件に対する優先度が同じときに検索にヒットするレコード数が多めの条件を選択すると考えられる。先に多めの条件で絞った方が、多くのレコードが検索にヒットする可能性が高い。例えば優先度が同じ条件が3つあったとする。条件Aは30件、条件Bは5件、条件Cは3件が検索にヒットするときに条件を絞り込むならば、選択する条件はAが良い。なぜならば、条件AはA、B、Cの中で一番たくさんレコードを持っているため、条件Aを選んだ後も他の条件を選択できる可能性が高くなるからである。レコード数にのみ着目すれば、まずAを選択し次にB、Cを選択するという順序が一番多くの条件を選択できる可能性が増す。また、カテゴリデータを絞った条件のレコメンドも重要である。例えばアルバイト全件に当てはまる条件の数と時給が1400円以上に対する条件の数は異なる可能性があるためである。例えばレコード全体であれば条件Aのレコード数が30件だが1400円以上と絞り込むことによって6件まで下がるかもしれない。そしてそのときにBが全体にたいして20件、1400円以上に対して10件であれば、1400円以上のときにBの方がAより検索ヒット数が多くなる。そのため検索者が1400円以上の時給が欲しいときにレコード数が少ないAよりも多いBを優先して選択した方が多くの条件を絞り込む可能性が高くなる。本インタフェースではこのことも考慮しタブの配置を最適化できるように実装を行った。

5.2 ResultView

検索結果をリスト形式で表示した。条件の選択後に **ResultView** で、条件全てを含むレコードが列挙される。また非2値データのカテゴリごと分けてテーブル上に検索結果を表示する。検索結果の情報をクリックすると詳細のURLへ移動することが可能である。

第6章 ユースケース

ユースケースは本検索インタフェースを用いてアルバイトの求人検索と旅行先における宿泊所検索を行う。どちらのユースケースにおいても、まず検索者モデルを設定しつつ、検索者の性質を反映させた検索システムの使用過程について記載する。

6.1 アルバイトの求人検索

6.1.1 検索者のモデル設定

検索者 A は新宿の自宅から目黒にある大学に通っている。アルバイト先の場所は、自宅から学校の間のエリアで探しているため検索エリアを新宿と目黒、渋谷に絞る。また人と関わる仕事をしたいため職種は、飲食とパブ・クラブ、接客の3つのカテゴリを選択した。尚、3つの職種の中でも特に客とコミュニケーションを取ることができそうなパブを志望している。

優先度が高い職場の条件として以下の5つをあげている。

1. アルバイトの時間や曜日を選択できること (日時選択可)
2. 週 1, 2 日から働くことができること (週 2)
3. 未経験者でも歓迎していること (未経験者歓迎)
4. 職場が駅から近いまたは、駅中にあること (駅近・駅中)
5. 1 日 4 時間以内の勤務が可能であること。 (4 時間以内)

尚日時選択可と週 2, 4 時間以内は特に優先度が高めである。

6.1.2 利用シーン

今回の検索では非 2 値データとして場所と職種、時給データを用いる。場所は新宿と目黒、渋谷で絞る。職種は飲食とパブ・クラブ、接客・サービスを条件として扱う。時給は 900 円以下, 900 円台, 1000 円台, 1100 円台, 1200 円台, 1300 円台, 1400 円以上に分割して表示する。また 2 値データとして上記の 5 つの条件に加え以下の 5 つも使用する。

1. 髪型・カラー自由 (髪型自由)
2. 大学生歓迎 (学生歓迎)

3. 服装自由
4. 外国語と触れ合える機会がある（外国語）
5. 学歴不問

尚、最初に検索を実行する2値データの条件は検索者が希望する1から5の2値条件とする。
以上の条件を用いて検索を実行すると図6.1が表示される。

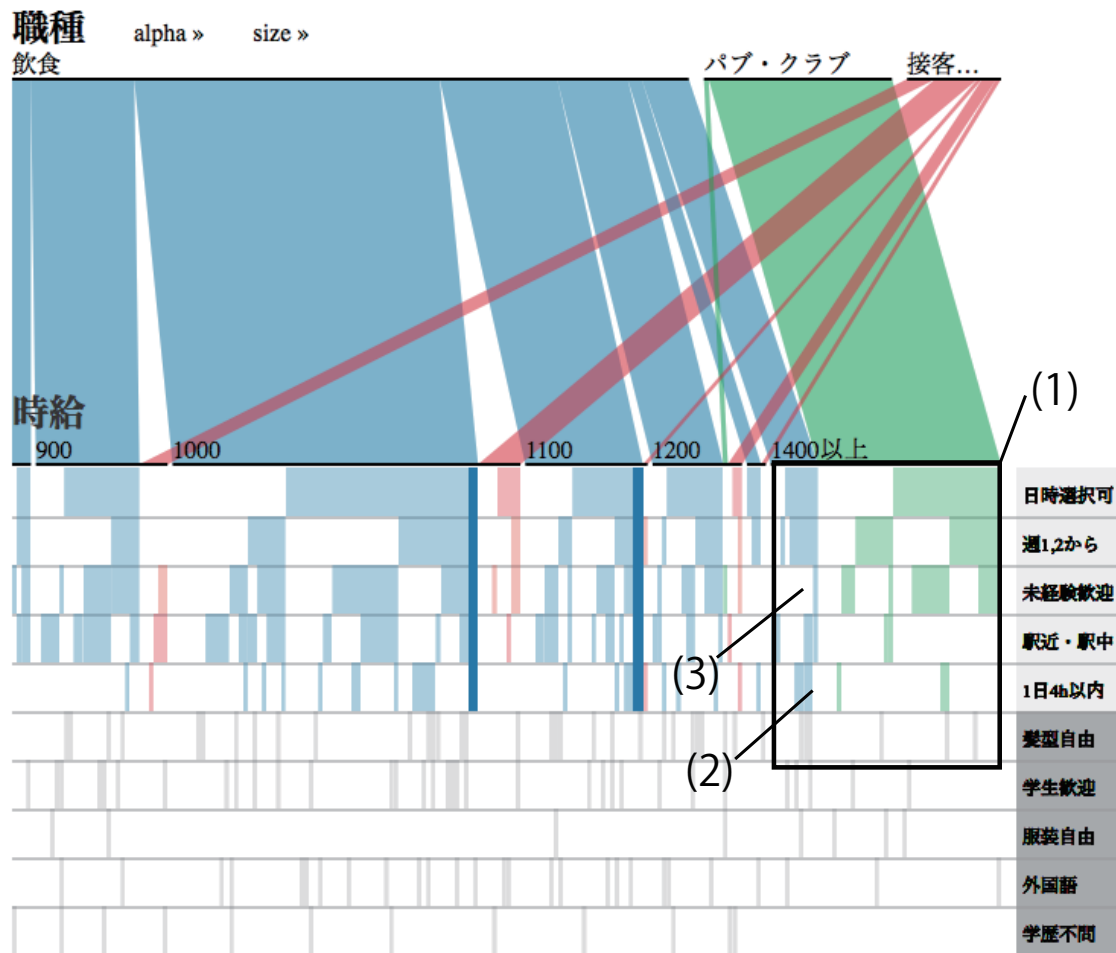


図 6.1: アルバイト検索：初期画面

検索者 A は図 6.1 の (1) を見て時給 1400 円以上のアルバイトがヒットしないことに気づく。全ての条件に該当するレコードについてはハイライトされているため検索者は概観してすぐに 1400 円以上のアルバイト先がヒットしていないと分かる。また、(2) と (3) を見て、1400 円のアルバイトをヒットさせるためには 4 時間以内か未経験者歓迎を削除することが必要であることが分かる。以上より非 2 値データの要素と 2 値データの要素間の関係性を読み取ることができたとと言える。次に、1400 円以上の

アルバイト先を見つけるために条件を変更する。4時間以内の条件が未経験者歓迎より優先度が高いため未経験者歓迎を抜くことにする。すると図 6.2 の (4) のようにハイライトされる箇所が広がり 1400 円のアルバイトもヒットすることが分かる。検索者は、未経験者歓迎の条件を抜くとどの程度、時給が下がる傾向にあるか把握することができた。つまり、条件を抜くことで 2 値データと非 2 値データの関係性の变化を知ることができたと言える。前述した操作で、1400 円の飲食のアルバイトはヒット

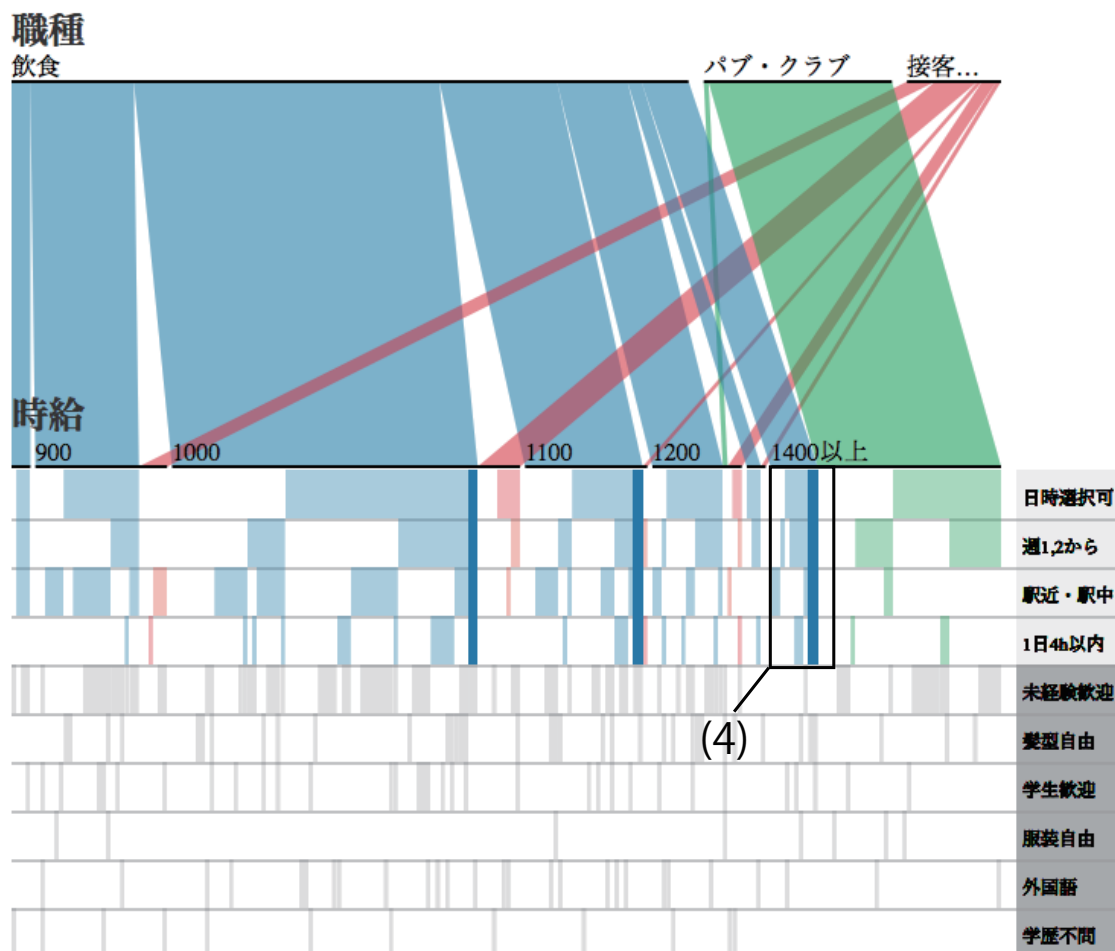


図 6.2: アルバイト検索：未経験者歓迎を抜いたとき

したが希望のパブ・クラブに関してはまだヒットしないため駅近の条件も抜いた。図 6.3 の (5) を見ると週 2 または 4 時間以内の条件を抜く必要があることが分かる。

4 時間以内の条件を抜くと図 6.4 の (6) のようになり 1400 円以上のパブ・クラブの求人を見つけることができた。また、図 6.4 の (6) を見ると日時選択可と 4 時間以内以外に未経験歓迎と外国語がヒットすることが分かる。

未経験歓迎と外国語を追加すると図 6.5 のようになった。また 4 時間以内の代わりに週 2 を抜くと図 6.6 が表示される。1400 円のパブ・クラブにおいて日時選択可と週 2 が含まれる描画箇所から下に向

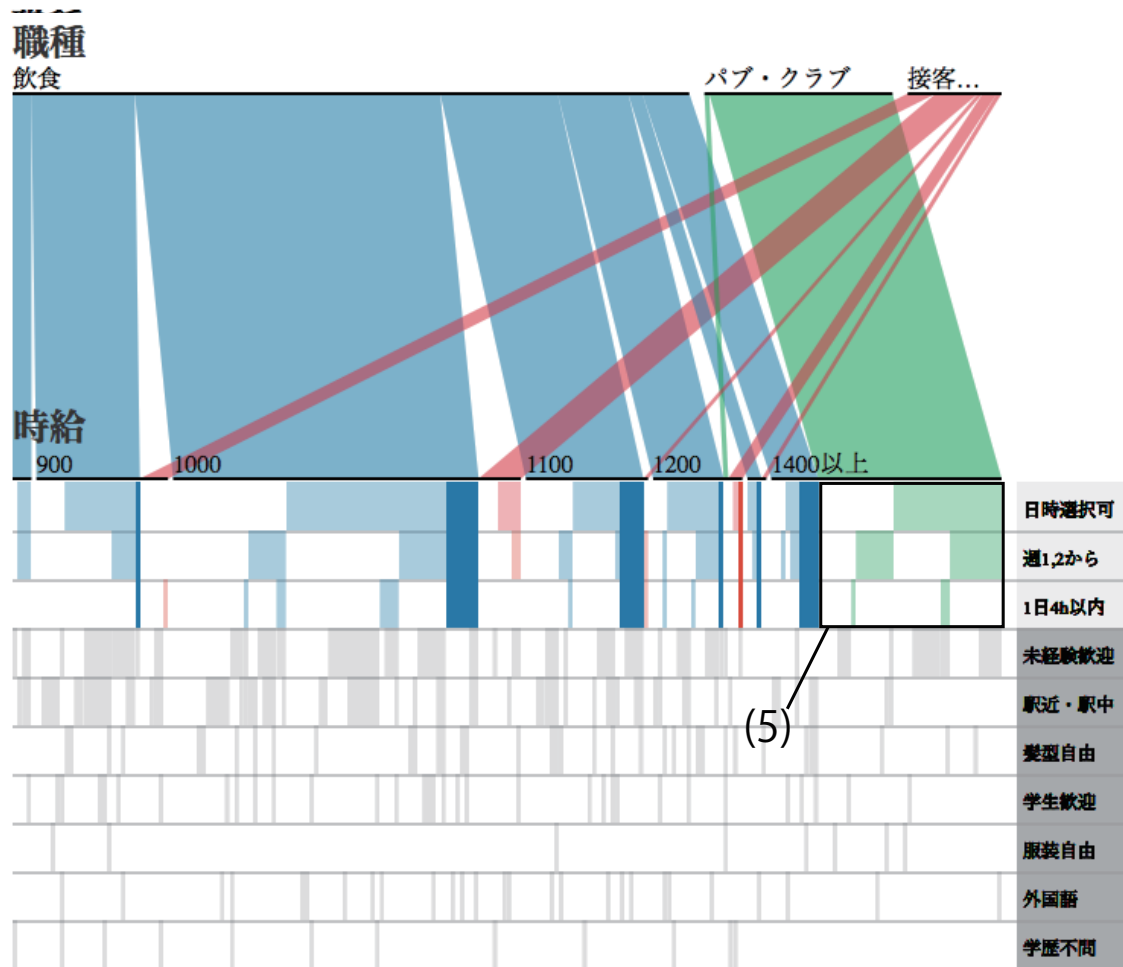


図 6.3: アルバイト検索：未経験者歓迎に加えて駅近・駅中の条件を抜いたとき

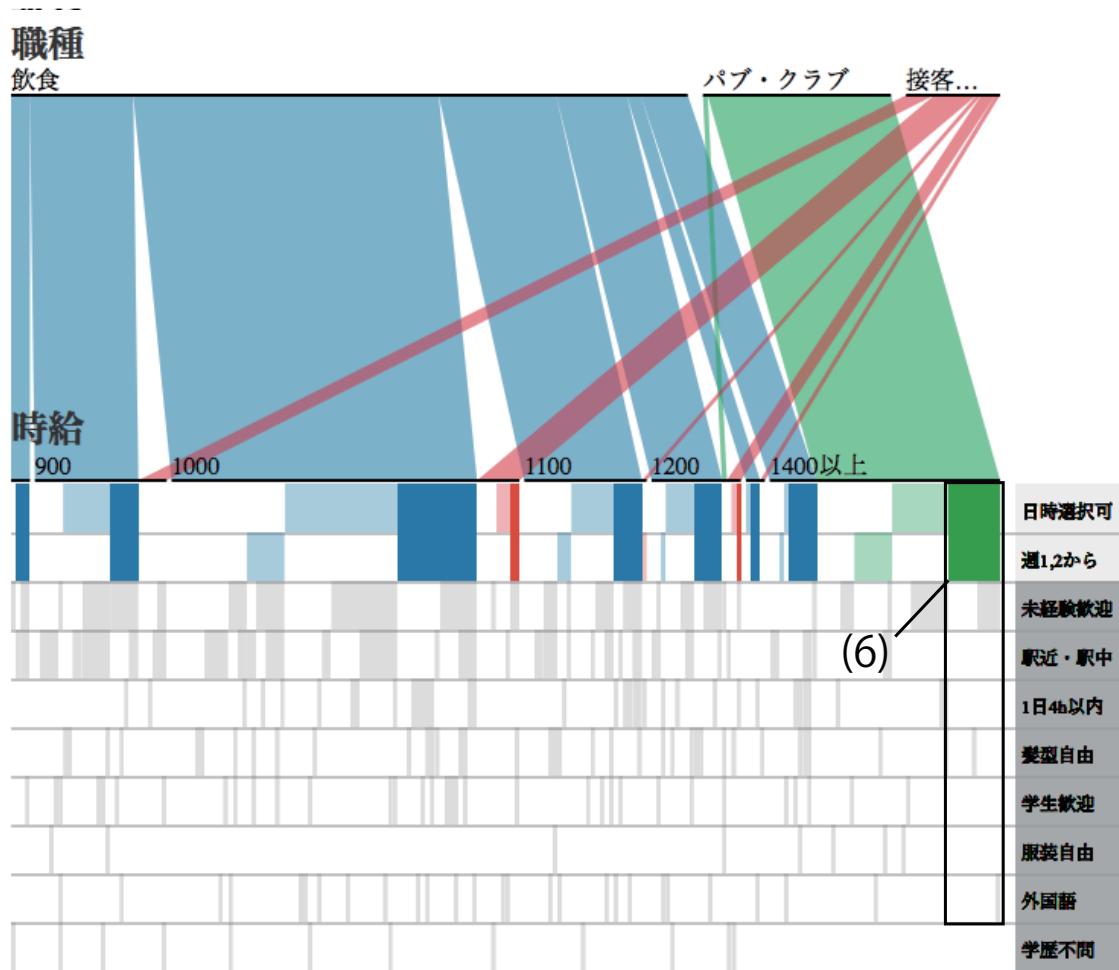


図 6.4: アルバイト検索：4 時間以内と未経験者歓迎，駅近・駅中を抜いたとき

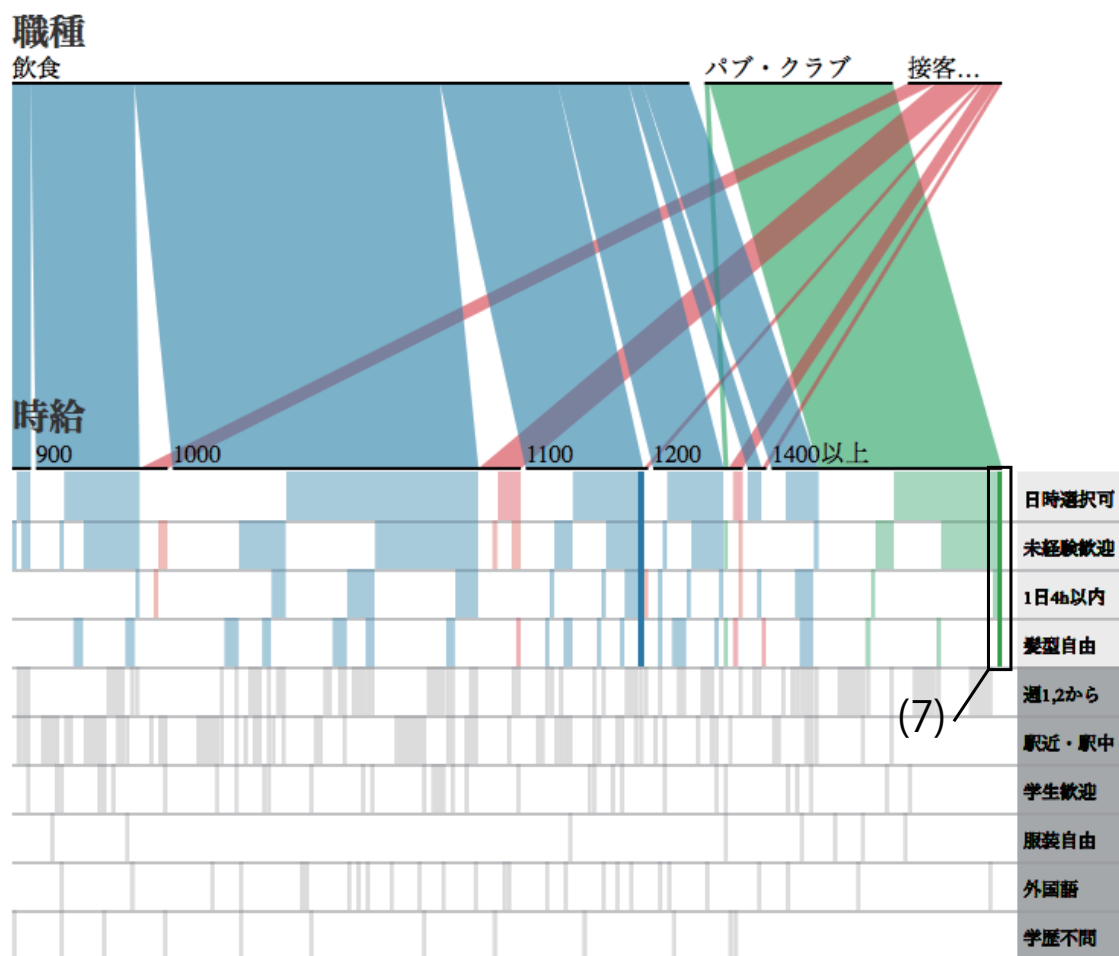


図 6.5: アルバイト検索：未経験歓迎と外国語を追加

かって見ていく (8)，すると他の条件として服装自由と未経験者歓迎の追加が可能なが分かった，

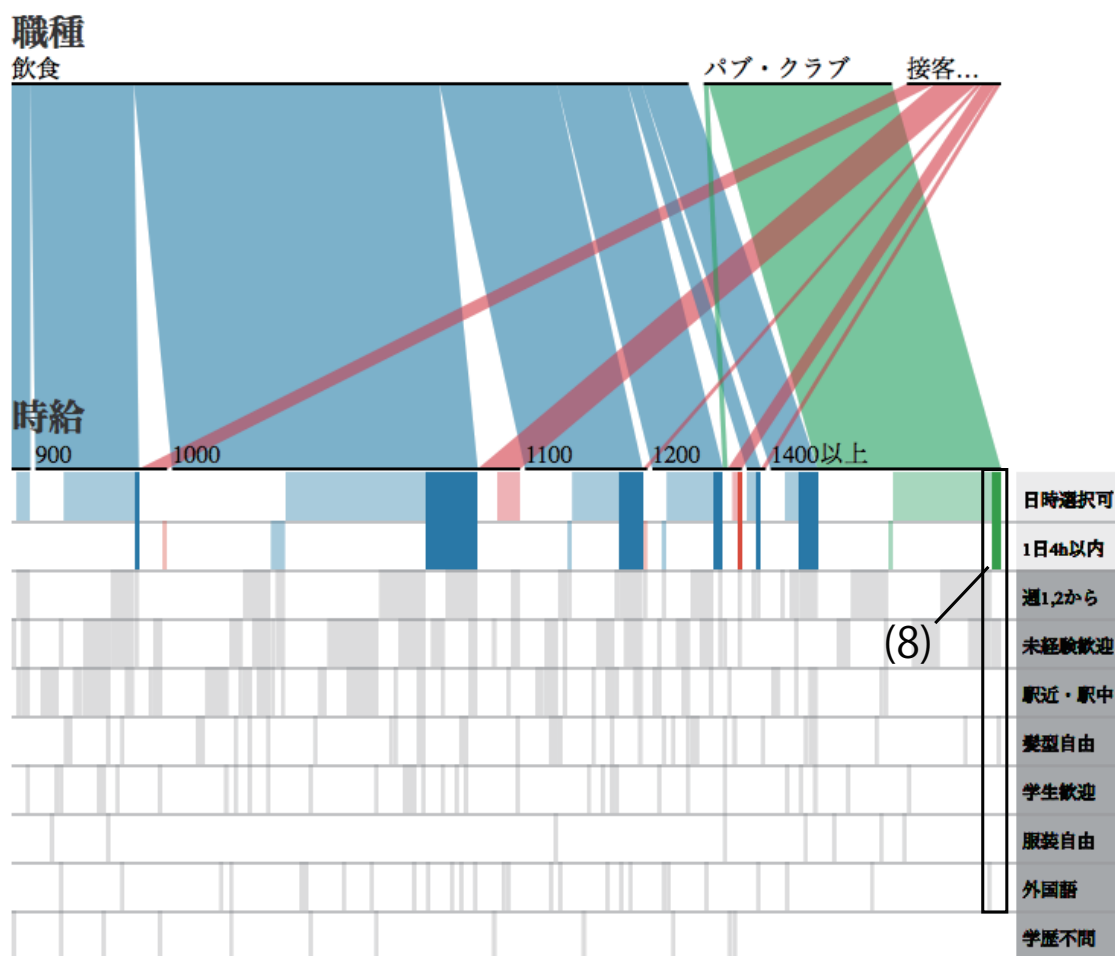


図 6.6: アルバイト検索：週 2 と未経験者歓迎，駅近・駅中を抜いたとき

尚，未経験者歓迎は最初に希望していた条件である．実際に条件を変更すると図 6.7 の (9) ようになる．

以上の操作により検索者 A はトレードオフの関係にある 4 つの検索条件を知ることができた．

1. 時給 (1100 円) と職種 (飲食) を妥協する代わりに希望の条件 5 つを全て含む．(初期の条件から服装自由も追加することが可能)
2. 未経験者歓迎とパブ・クラブをあきらめる代わりに，未経験者以外の条件を抜かずに 1400 円の飲食が見つかった．(初期条件から服装自由と学生歓迎も追加することが可能)
3. 4 時間以内をあきらめる代わりに 1400 円以上のパブが見つかった．(初期の条件から外国語も追加することが可能)

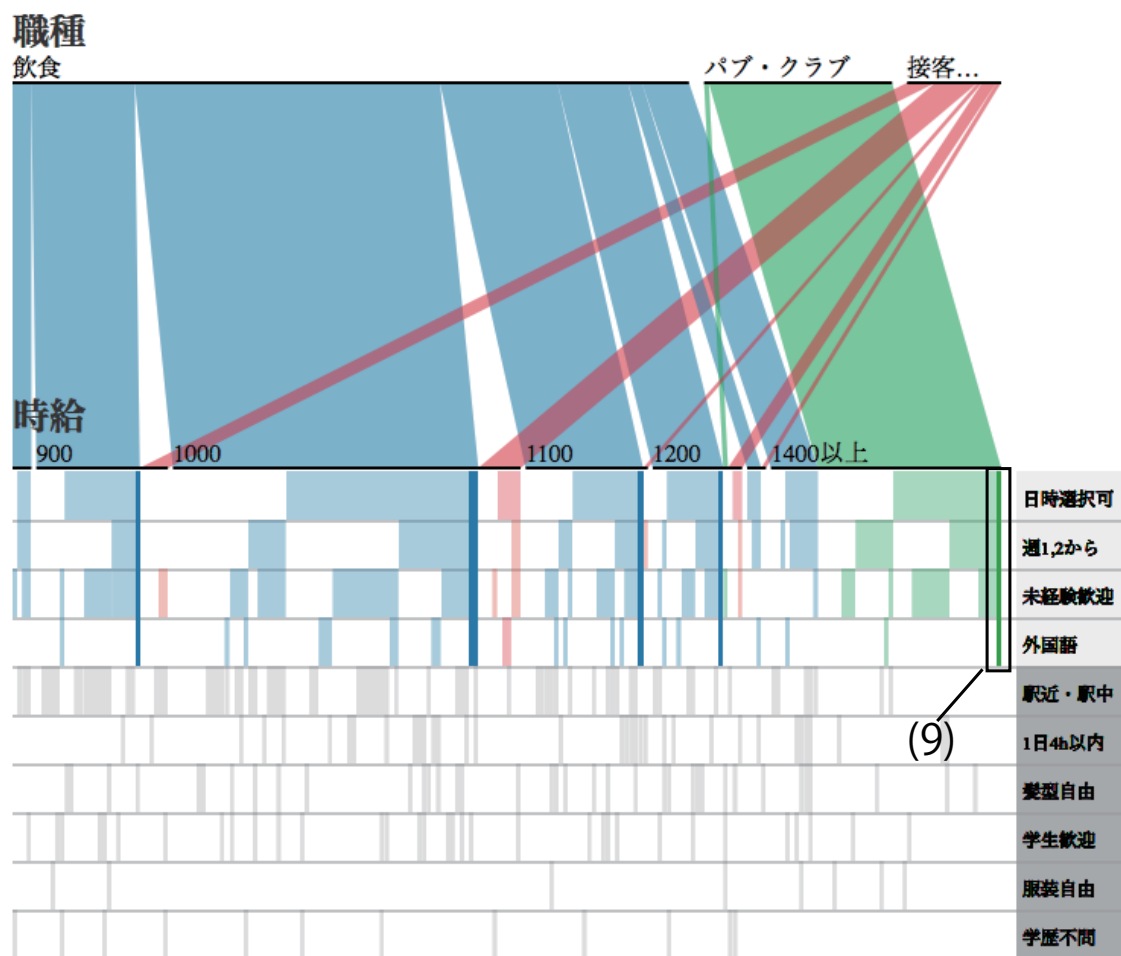


図 6.7: アルバイト検索：外国語と未経験者歓迎を追加

4. 週2をあきらめる代わりに1400円以上のパブが見つかった。(初期の条件から服装自由も追加することが可能)

6.2 宿泊先データを用いた例

6.2.1 検索者のモデル設定

検索者Bは、2015年3月15日に2泊3日、4人で白馬のスキー場に行くことを考えている。部屋は2部屋予約するつもりである。宿の場所はスキー場からできるだけ近くで、料金は5000円程度を希望している。

6.2.2 使用データの説明

非2値データとしてスキー場までの距離とホテルの宿泊費(参考料金)を用いる。

距離は行こうと思っているスキー場から宿泊所までの距離を表している。また料金は2000円から3499円、3500円から4999円、5000円から6499円、6500円から7999円、8000円以上にカテゴリ化した。2値表現可能な条件は下記が用意されている。

1. 館内に内湯・大浴場がある宿泊施設(大浴場)
2. 客室内にバス・トイレが付いている宿泊プランのある宿泊施設(バス・トイレ)
3. 館内・敷地内に宿泊者向けに貸しスキーがある宿泊施設(貸しスキー)
4. 部屋でインターネットが利用できる宿泊プランのある宿泊施設(ネット)
5. 温泉がある宿泊施設(温泉)
6. 禁煙ルームのある宿泊施設(禁煙室)
7. マッサージサービスがある、もしくは、手配可能な宿泊施設(マッサージ)
8. 最寄のコンビニまで徒歩5分以内の宿泊施設(コンビニ)
9. サウナ設備がある宿泊施設(サウナ)
10. レイティングが4以上の施設(高評価)

6.2.3 宿泊所検索の利用シーン

検索者Bは、2値条件に関して大浴場、バス・トイレ、貸しスキー、ネット、温泉、コンビニ、高評価を希望している。そこで大浴場、バス・トイレ、貸しスキー、ネット、温泉、コンビニ、高評価で検索にかけると以下の図のようになる。図6.8を概観すると(1)が赤くハイライト表示されている。しかし料金が8000円以上かつ距離が5kmから7kmしかヒットしない。宿泊所の料金が低いほうから

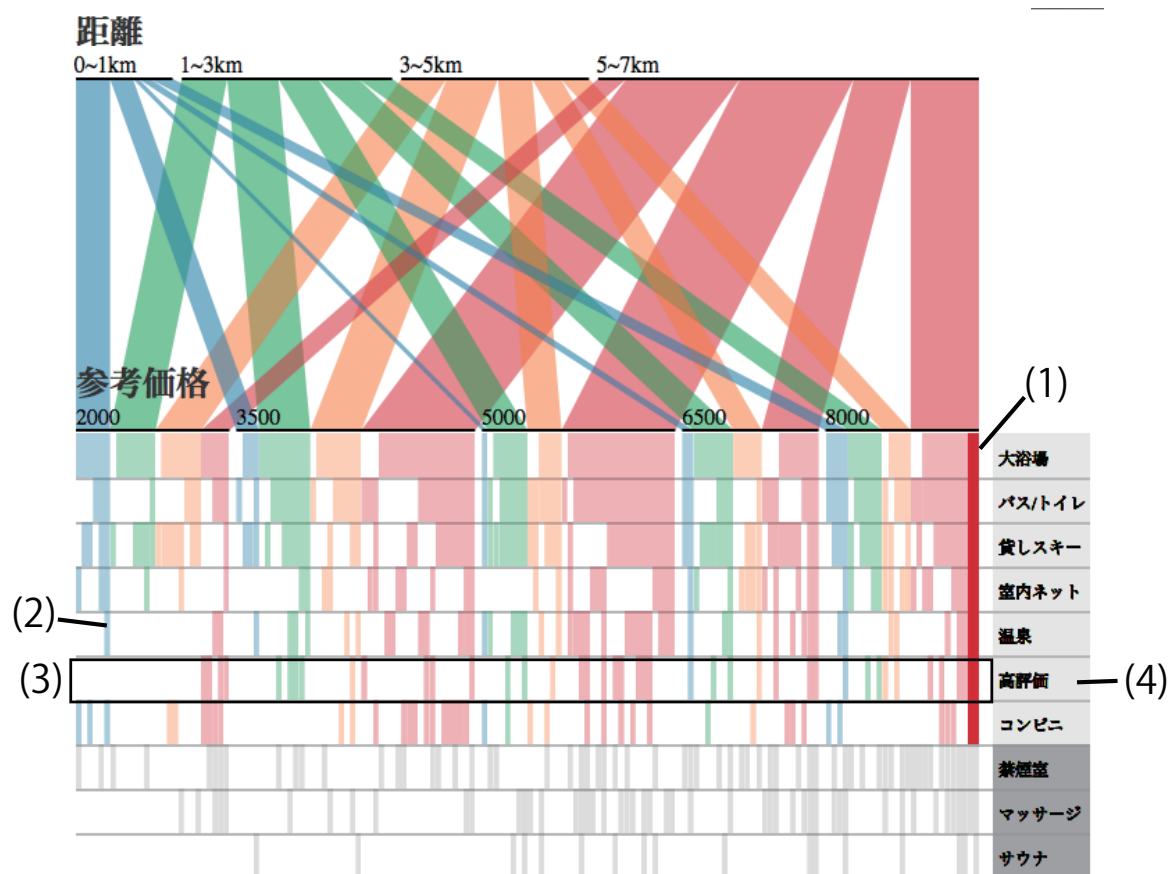


図 6.8: 宿検索：初期画面

見ていくと、1km 以内の 3500 円以内の記録 (2) において、高評価を抜けば 2000 円で 1km 圏内に泊まることが可能であることが分かる。検索者 B は本インタフェースを概観することで非 2 値データと 2 値データの関係性を知ることができたと言える。また (3) の範囲で高評価の条件を見るとレ

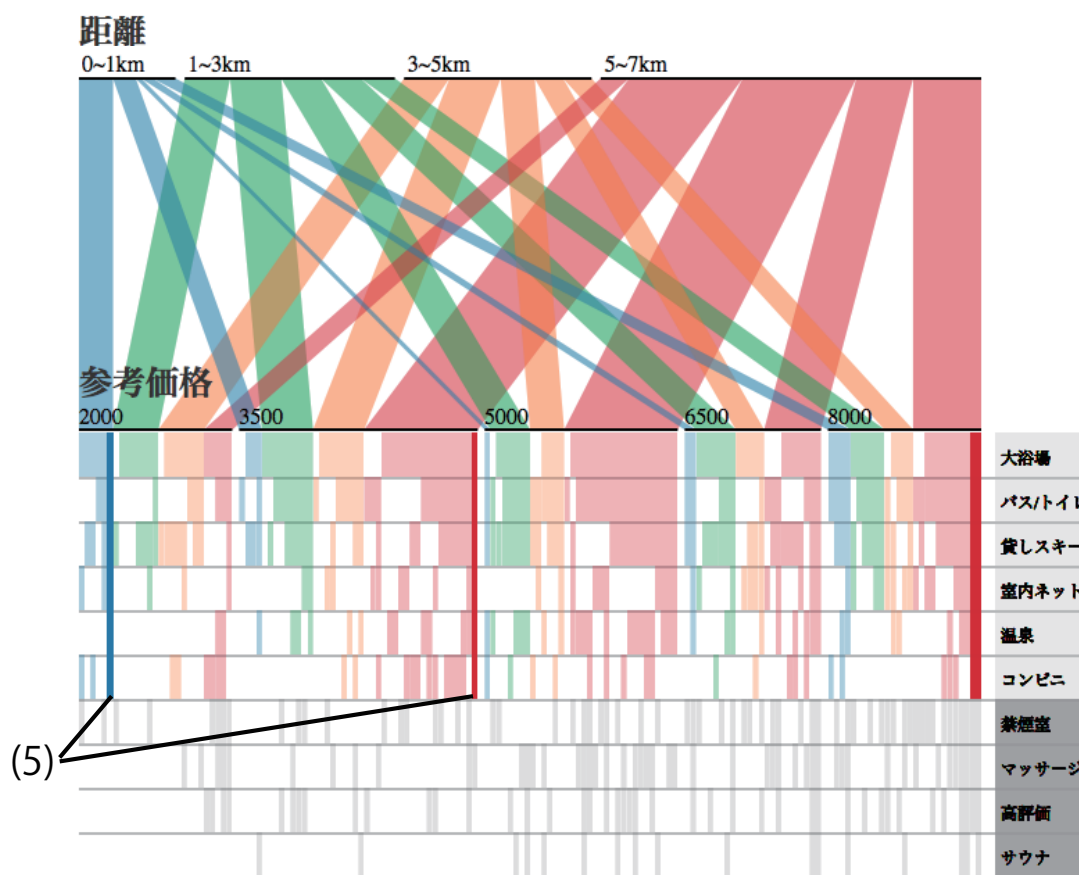


図 6.9: 宿検索：高評価を抜いた図

コードが参考価格が高い方に偏っているため高評価の縛りを抜けば価格が安めの宿泊先もヒットしやすくなることが分かる。本インタフェースを使用すれば、条件を表す行を見て、レコードの分布の仕方を知ることによってデータの傾向を把握することも可能である。(4) をクリックし高評価を抜くと図 6.9 のようになり (5) の 1km 以内かつ 2000 円から 3499 円と 5km から 7km かつ 3500 円から 4999 円の宿泊先が増えた。料金が低めのところにおいてもハイライト表示されるようになり、条件を抜いたときの非 2 値データと 2 値データの関係性の変化も知ることができた。

しかし口コミがいい場所に泊まりたい欲求があるため、高評価を抜かない方法で 5000 円以内で 1km から 3km 以内にあるレコードを調べる。検索者は高評価を抜かないと決めていて、さらに該当レコード数が少ないため、図 6.10 の高評価の行に着目する。すると 3500 円以内かつ 3km 以内で該当するレコードが存在しないことが分かる。3500 円から 5000 円であり 3km から 5km の宿泊先を見ると (6)、コンビニと温泉の条件かネットの条件を抜けばヒットすることが分かる。検索者 B にとってネットの

方が温泉より重要なのでコンビニと温泉を抜く (7) と図 6.11 のようになる

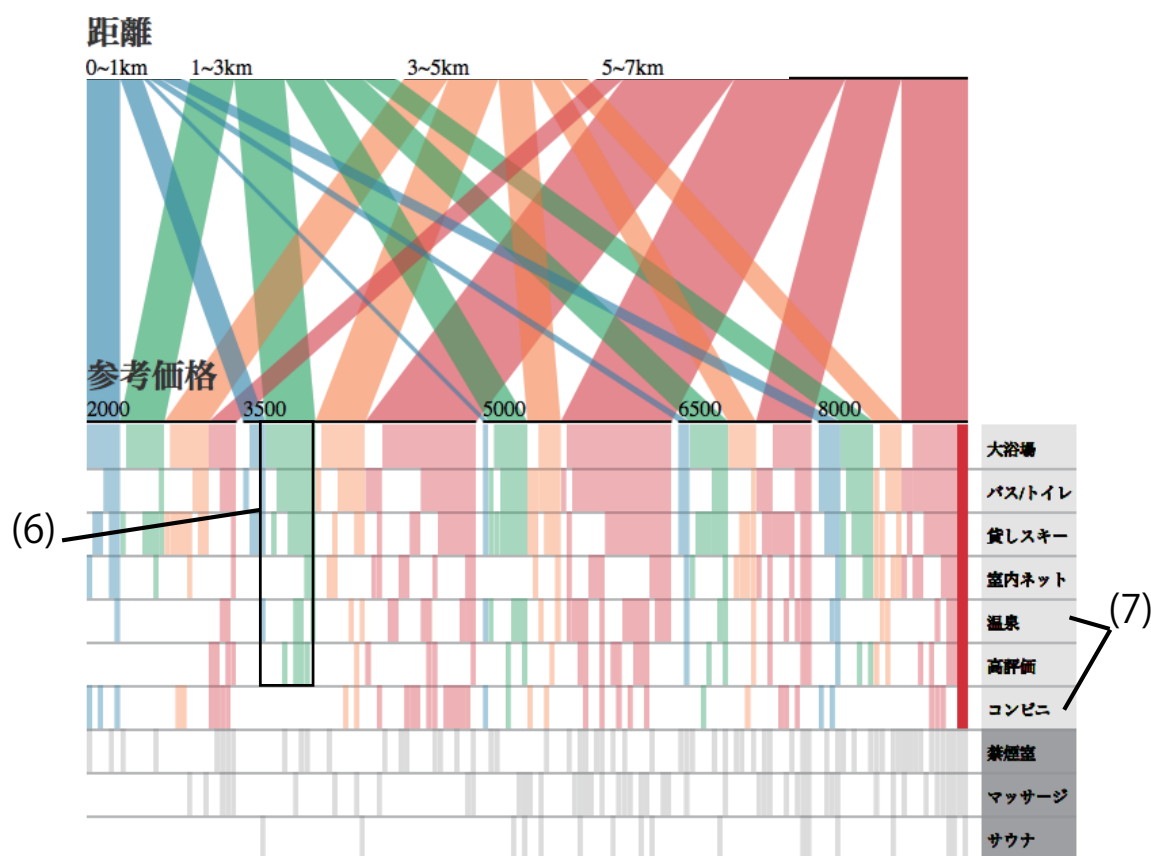


図 6.10: 宿検索：高評価を抜かないときの図

このときに料金が高い方へ視点をずらすと 6500 円から 8000 円かつ 1km 以内の記録も当てはまることが分かる (8). また 6500 円の記録は禁煙室と温泉, マッサージも該当することが分かる
以上の操作により検索者 B はトレードオフの関係にある 3 つの検索条件を知ることができた.

1. 高評価をあきらめる代わりに 1km 以内であり 2000 円から 3500 円を選択可能な条件
2. コンビニと温泉をあきらめる代わりに 1km から 3km 以内であり 3500 円から 5000 円と高評価を選択可能な条件
3. 料金 (6500 円から 8000 円になってしまう) とコンビニをあきらめる代わりに, マッサージ, 禁煙室を追加できる上に距離が 1km 以内の条件

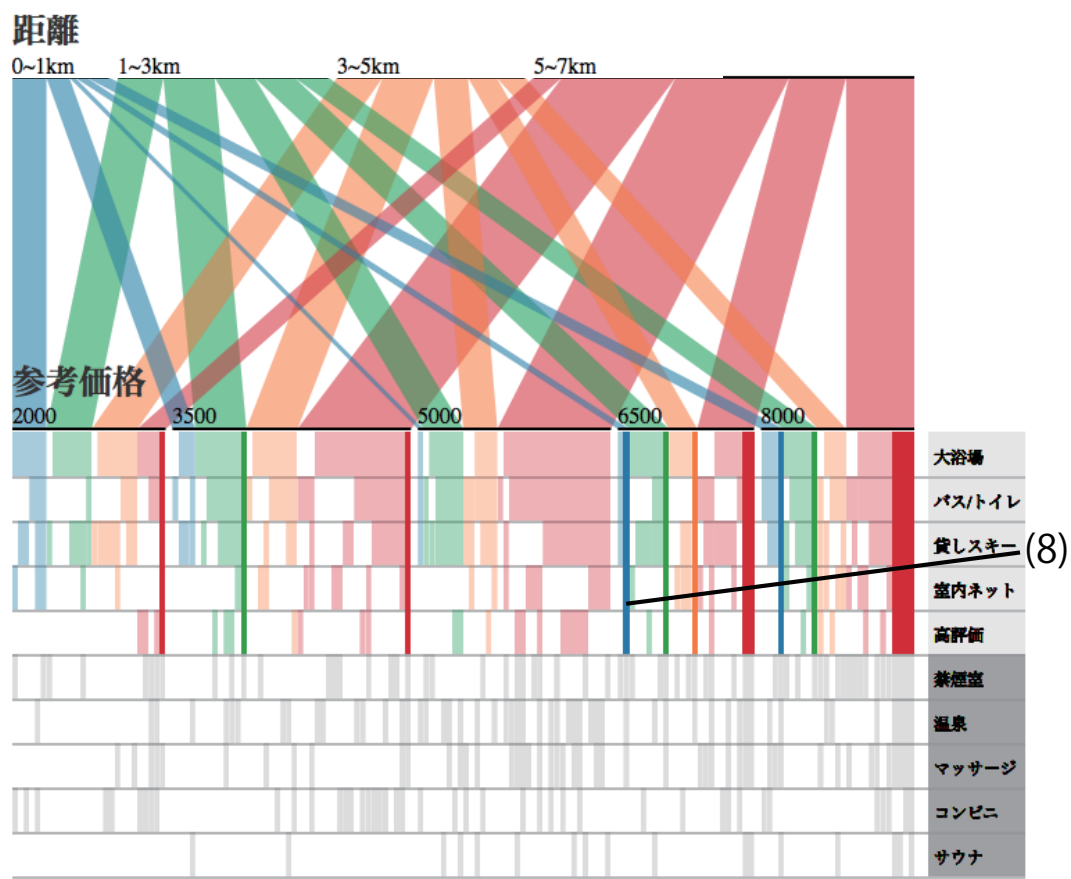


図 6.11: コンビニと温泉を抜いた図

第7章 結論

本研究は2値で表現可能なカテゴリデータを条件とした情報の検索支援を可能にした。既存の検索システムには、条件の選択を繰り返し、検索結果を絞り込む方式がある。しかし、この検索方式には幾つかの問題点がある。例えば、検索条件を絞り込み過ぎると検索ヒット数が0になる可能性があり、検索者は何度も繰り返し検索しなければならない点があげられる。他にも検索者の要望と検索結果のトレードオフを考慮して情報を見つけることが難しい点があげられる。そこで、我々はそのような問題点を解決するために、カテゴリデータの概観を検索インタフェースに組み込み、検索の支援をするインタフェースを開発することを考えた。通常カテゴリデータの概観を可視化するには名義尺度の尺度水準を持つ多変量データを扱う必要がある。Parallel Sets という可視化手法を用いれば、そのようなデータを扱うことが可能である。しかし Parallel Sets では表示の際に線が重なりあってしまうため特徴の読み取りが困難であった。そのため我々は Parallel Sets と行列表現を直接連結させることで、カテゴリデータの概観を組み込んだ検索インタフェースを開発した。このインタフェースにより、検索者は条件を絞る際に、カテゴリデータを概観できるため検索件数が0にならないような検索が可能となった。また、検索条件に全て該当するレコードに対してハイライト表示を施した。そのように設計することで検索者が条件を絞る際に、ハイライト表示の箇所の変化や条件に対するレコードの分布を見ることが可能となった。結果、検索者の要望と検索結果のトレードオフを考慮した検索結果を求めることができるようになった。

謝辞

本研究を進めるにあたり、三末和男准教授をはじめ田中二郎教授、志築文太郎准教授、高橋伸准教授にはたくさんの助言をいただきました。ありがとうございます。特に三末先生は、研究が行き詰まったときに親身になって相談に乗って下さったこと深く感謝しております。また研究のみならず論文執筆においても熱心に添削して下さいのおかげで、論文の質を高めることができました。インタラクティブプログラミング研究室の皆様、特に NAIS チームの方々には心からお礼申し上げます。私が研究の進め方が分からず右往左往しているときに、NAIS チームの方々から、ためになる助言や指摘を多くいただきました。また論文執筆の際も丁寧なコメントをいただき感謝の気持ちでいっぱいです。最後に、大学生活を送る中でお世話になった方や友人、生活を支えて下さった家族に感謝を申し上げます。

参考文献

- [1] Marti A. Hearst, Next Generation Web Search: Setting Our Sites, *IEEE DATA ENGINEERING BULLETIN*, 23(3), pp. 38-48, 2000.
- [2] F. Bendix, R. Kosara, and H. Hauser, Parallel sets: Visual analysis of categorical data, in *Proceedings IEEE Information Visualization*. IEEE CS Press, pp. 133-140, 2005.
- [3] M. Friendly, Mosaic displays for multi-way contingency tables, *American Statistical Association*, 89(425), pp. 190-200, 1994.
- [4] A. Inselberg and B. Dimsdale, The plane with parallel coordinates, *The Visual Computer*, 1(4), pp. 69-91, 1985.
- [5] E. Clarkson, K. Desai, and J. D. Foley, ResultMaps: Visualization for Search Interfaces, *IEEE Transactions. Visualization and Computer Graphics*, 15(6), pp. 1057-1064, 2009.
- [6] J. English, M. Hearst, R. Sinha, K. Swearingen, and P. Yee, Flexible Search and Navigation using Faceted Metadata, 2002.
- [7] B. Kim, B. Lee, and J. Seo, OnSet: A Visualization Technique for Large scale Binary Set Data, *IEEE Transactions on Visualization and Computer Graphics*, 20(12), pp. 1993-2002, 2014.
- [8] S. Gratzl, N. Gehlenborg, A. Lex, H. Pfister, and M. Streit, Domino: Extracting, Comparing, and Manipulating Subsets across Multiple Tabular Datasets, *IEEE Transactions on Visualization and Computer Graphics (InfoVis '14)*, 20(12), pp. 2023-2032, 2014.
- [9] T. Hansaki, B. Shizuki, K. Misue, and J. Tanaka: FindFlow: Visual interface for information search based on intermediate results, in *Proceedings of the 2006 Asia-Pacific Symposium on Information Visualisation-Volume 60*, pp. 147-152, 2006.
- [10] B. Kovalerchuk and F. Delizy, Visual Data Mining Using Monotone Boolean Functions, *Visual and Spatial Analysis. Advances in Data Mining, Reasoning, and Problem Solving*, 4, pp. 387-406, 2004.
- [11] B. Kim, B. Lee, and J. Seo, Visualizing Concordance of Sets, *Interacting with Computers*, 19(5), pp. 630-643, 2007.
- [12] C. Ware, Second Edition *Information visualization: Perception for Design*, Elsevier, pp. 123-127, 2004.