DEIM Forum 2022 H21-1

ユーザーコンテキストを考慮したインサイト評価手法

野澤　拓磨† 　董　　于洋† 　榎本　昌文† 　小山田昌史†

† 日本電気株式会社 データサイエンス研究所　〒 211–8666 神奈川県川崎市中原区下沼部 1753

E-mail: †{nozawa-takuma,dongyuyang,masafumi-enomoto,oyamada}@nec.com

あらまし　近年，データベースに蓄積された大規模データから有用な知見を発見するインサイト自動発見の研究に注目が集まっている．インサイト自動発見では，過去にエキスパートによって可視化されたデータの特徴とその可視化形式を学習したインサイト評価モデルを用いることで，有用なインサイトを与えるチャートをユーザーに推薦する．しかしながら，既存のインサイト評価モデルは，ユーザーの関心事項や分析目的などのユーザーコンテキストに関係なく，可視化されるデータの特徴と可視化形式にのみに基づいた評価を行うため，本来はユーザーが求めていないチャートを推薦してしまう可能性がある．本研究では，ユーザーコンテキストと可視化されるデータの属性の共起関係に基づくキャリブレーションと，ユーザーコンテキストをクエリとみなした attention 機構の導入により，ユーザーが求めるインサイトを与えるチャートを推薦する手法を提案した．Plotly データセットを用いた実験の結果，既存手法と比べてユーザーの求めるチャートの推薦精度を大きく改善可能であることが確認できた．

キーワード　可視化，機械学習，Attention

1　はじめに

1.1　インサイト自動発見

データから発見される有益な知見はインサイトと呼ばれ，データベースに蓄積された大規模データからインサイトを発見することは，データ分析の重要なタスクの１つである．インサイトの発見のためには，活用可能なデータの豊富さが重要であるが，分析対象や可視化のバリエーションの増加によって，分析者の作業負荷を大きく増加させてしまう課題も同時に存在する [1]．このような背景から，データからインサイトを自動的に発見するインサイト自動発見技術に注目が集まっており，Business Intelligence (BI) ツールにも導入が進んでいる [2]．

1.2　既存技術の課題

インサイト自動発見の研究では，各チャートを構成するデータやその可視化形式に関するインサイト評価モデルを導入することで，有用な知見を与えると考えられるチャートに絞り込むアプローチが行われている．絞り込まれた有用なチャートを中心に観察することで，分析者は比較的少ない労力でインサイトを発見することが可能である．インサイトの評価モデルには様々なバリエーションが存在するが，過去にエキスパートによって可視化されたデータの特徴とその可視化形式を学習する Machine Learning (ML) ベースのモデルが近年の主流となっており，活発に研究が進められている [3–13]．しかしながら，既存の ML ベースモデルは，ユーザーの関心事項や分析目的などのユーザーコンテキストに関係なく，可視化されるデータの特徴と可視化形式にのみに基づいた評価を行うため，本来はユーザーが求めていないチャートを推薦してしまう可能性がある．例えば，あるユーザーが “売上推移” に関する分析を行うケースを想定する．この場合，ユーザーは何らかの顕著な特徴を備えた売上推移のチャートを期待していると考えられるが，既存の ML ベースモデルはユーザーが求めていない “商品価格” のチャートを推薦してしまう可能性がある．すなわち，インサイト評価モデルによって有用なチャートへの絞り込みを行った後でも，ユーザーは目的とは関係のないチャートを確認する必要があり，その分析作業は未だ煩わしさを伴う．また，ユーザーの関心事項や分析目的に関係するチャートの評価値が相対的に低かった場合には，絞り込みの対象からあぶれることにより，ユーザーが重要なインサイトを見逃してしまう可能性もある．このように，インサイトの評価においてユーザーの関心事項や分析目的などのユーザーコンテキストを考慮することは重要であると言える．

上記の問題に対処するシンプルなアプローチとして，ユーザーコンテキストと関連性のないデータをフィルタリングしてしまう方法が考えられる．ユーザーコンテキストに関係するデータのみを分析対象とすることで，関係のないチャートの推薦を防ぐことができる．しかしながら，このシンプルなフィルタリングには２つの問題が存在する．１つ目は，関連性の評価に関する問題である．例えば，ユーザーコンテキストとして売上推移が与えられている場合に，売上に関するデータは存在していても，推移と関係するデータは明示的に与えられていないことがあると考えられる．したがって，どのような基準に基づいてフィルタリングを実行するかについては一考の余地がある．２つ目は，ユーザーコンテキストに応じて重視するデータの特徴や適切な可視化形式が異なる問題である．例えば，売上推移に関する分析を行う場合には，時系列的な特徴を持ったデータが折れ線グラフ等の形式で可視化されることが期待されるが，商品価格に関する分析を行う場合にはその限りではない．シンプルなフィルタリングでは，このようにユーザーコンテキストごとに重要視するデータの特徴や可視化形式の違いを考慮することができない．

1.3 本研究の目的

上記の課題を解決するために，ユーザーコンテキストを考慮したインサイトの評価方法についての研究を行った．提案法は，ユーザーコンテキストと可視化されるデータの属性の共起関係に基づくキャリブレーションと，ユーザーコンテキストをクエリとみなした attention 機構を導入することにより，ユーザーの求めるインサイトを優先的に提示する ML ベースのインサイト評価モデルを実現する．

1.4 本研究の貢献

本研究による貢献は以下の 3 点である．

- ユーザーコンテキストを考慮したインサイト自動発見についての定式化を行った．
- ユーザーコンテキストを考慮したインサイト自動発見を実現する ML ベースのインサイト評価モデルを開発した．
- 実データを用いた実験により，上記のモデルが有用かつユーザーコンテキストに関連したチャートを発見可能であることを示した．

2 関 連 研 究

インサイト自動発見の先行研究は大きく分けてルールベース型と ML ベース型のモデルに大別される．ルールベースモデルは，経験知に基づき人が有用と考えるインサイトの種類やその評価関数を事前に定義しておくことで，目的のインサイトを与えるデータやチャートを選別する手法である [14–16]．古くから数多くの研究がなされており，これまでに大きく分けて 12 種類のインサイトに関する評価方法が提案されている [17]．ML ベースモデルは，過去にエキスパートによって可視化されたデータの特徴とその可視化形式を学習しておくことによって，機械学習を用いた推論に基づき有用なチャートを推薦する [3–8]．学習時には可視化コーパスが必要になるため，その整備に関する研究も並行して進められている [4, 9, 10]．ML ベースモデルは，インサイトの種類やその評価関数を事前に定義する必要がないというメリットがある．

近年では，コンテキストに応じたチャートを推薦するモデルの提案もなされている．例えば，VisGuide [11] は，分析の drill-down においてユーザーが興味持ったデータの特徴を分析し，ユーザーの好みと思われるチャートを推薦するモデルを提案している．VizRec [12] や PVisRec [13] も，過去の可視化履歴等を基に personalized されたインサイト評価モデルを導入している．これらのモデルは，ユーザーコンテキストに関係するチャートを推薦可能であると考えられるが，ユーザーの可視化履歴等の情報を必要とするため，コールドスタートな状況で用いることが難しい．一方で，提案法はユーザーの関心事項や分析目的に関する情報を自然言語で与えることで，可視化履歴等を利用することなく，有用かつユーザーの求めるチャートを提示することが可能である．Intent-Viz [18] は，本研究と近い問題設定に取り組んでいるが 2 つ相違点が存在する．1 つ目は，Intent-Viz が可視化形式と可視化属性の選択を別々に扱っているのに対して，本研究ではチャートのインサイトを end-to-end に評価している点である．2 つ目は，Intent-Viz が特定のデータテーブルにおけるチャートの推薦を想定しているのに対して，本研究は複数のデータテーブルから目的のチャートを発見する問題を扱っている点である．すなわち，本研究で提案するモデルは情報検索的な側面を持つと言える．

3 背景と問題設定

3.1 準　　備

Qian et al. [8] らによる先行研究における記述に倣い，本問題設定におけるデータセット，チャートの構成要素，ユーザーコンテキスト等の詳細について述べる．

3.1.1 可視化属性

本研究で扱うデータセット $\mathbf{X}_i$ は少なくとも 2 つ以上の属性を含む多次元データであり，チャートとして可視化される属性の組み合わせ候補が複数存在する．ある組み合わせを $\mathbf{X}_i^{(k)}$ とし，その集合を $\mathcal{X}_i = \{\mathbf{X}_i^{(1)}, ..., \mathbf{X}_i^{(k)}, ...\}$ と表記する．以降では，$\mathbf{X}_i^{(k)}$ を構成する属性を可視化属性と呼び，その属性名を z_{ij}，属性ベクトルを $\mathbf{x}_{ij}$ と表記する．

3.1.2 可視化形式

データを可視化する際には，可視化の形式すなわちデータをどのように表示するかについても考慮する必要がある．データセット $\mathbf{X}_i$ における，ある可視化形式を C_{ik} とし，その集合を $\mathcal{C}_i = \{C_{i1}, ..., C_{ik}, ...\}$ と表記する．可視化形式 C_{ik} には，チャートタイプ (bar, scatter, ...), チャートモード (line, marker, ...)，プロットのサイズ (1pt, 2pt, ...), x/y 軸のデータ型 (categorical, temporal, or quantitative) などが含まれる．

3.1.3 可視化コーパス

ML ベースのアプローチでは，N 個のデータセット $\{\mathbf{X}_i\}_{i=1}^N$ およびその可視化 $\{\mathbb{V}_i\}_{i=1}^N$ からなる可視化コーパス $\mathcal{D} = \{\mathbf{X}_i, \mathbb{V}_i\}_{i=1}^N$ を学習に用いる．可視化 $\mathbb{V}_i$ はチャートの構成要素となるデータや可視化形式を記述するものであり，可視化 $V_{ik} \in \mathbb{V}_i$ がチャートと 1 対 1 に対応する．可視化コーパスを構成するデータセットおよびその可視化は，任意のデータソースから収集可能である．例えば，Plotly Community Feed[1] や Tableau Public[2] といった可視化プラットフォーム上でコミュニティメンバーによって公開されているデータセット及びその可視化をクローリングすることで収集しても良い．コミュニティ上で公開されているチャートは，他のユーザーに何かしら有益な情報を示すものであることが想定されるため，クローリングによって収集された可視化 $\mathbb{V}_i^+$ は，ポジティブサンプルとして利用することができる．逆に，$\mathbf{X}_i$ において考えられるパターンにも関わらず公開されていない可視化 $\mathbb{V}_i^-$ は，有用ではなかったと考えられるため，ネガティブサンプルとして扱う．

データセット $\mathbf{X}_i$ について考えられる全可視化 $\mathbb{V}_i^{\text{all}}$ は，$\mathcal{X}_i$ と $\mathcal{C}_i$ の組み合わせによって与えられる．

1：https://chart-studio.plotly.com/feed/#

2：https://public.tableau.com/s/

$$\mathbb{V}_i^{\mathrm{all}} = \mathcal{X}_i \times \mathcal{C}_i. \tag{1}$$

前述のように，ネガティブサンプルは全可視化からポジティブサンプルを差し引くことで与えられる．

$$\mathbb{V}_i^- = \mathbb{V}_i^{\mathrm{all}} \backslash \mathbb{V}_i^+. \tag{2}$$

なお，ネガティブサンプルの数はポジティブサンプルよりも多く存在している場合があり，この不均衡さが学習時の課題になる可能性がある．そこで，ネガティブサンプルのうちのいくつかをランダムサンプリングすることによって，学習時のサンプルの不均衡さを解決する．ランダムサンプリングされたネガティブサンプルを $\hat{\mathbb{V}}_i^- \subseteq \mathbb{V}_i^-$ と表記すると，データセット $\mathbf{X}_i$ に関する可視化は $\mathbb{V}_i = \mathbb{V}_i^+ \cup \hat{\mathbb{V}}_i^-$ である．

3.1.4 ユーザーコンテキスト

ある可視化 $V_{ik} \in \mathbb{V}_i$ におけるユーザーコンテキストを $u_{ik} \in \mathcal{U}_i$ と表記する．ユーザーコンテキストとして捉えられる情報としては，ユーザーの関心事項や分析目的などの可視化意図，年齢や職業などのユーザー属性，時刻や場所などの可視化シーン等，様々な種類が考えられるが，簡単のため本研究では可視化意図にのみ注目する．ユーザーの可視化意図そのものを直接的に取得することはできないが，ユーザーの可視化意図はチャートタイトルとして表面化していると想定し，チャートタイトルをユーザーコンテキストとみなして用いることにする．したがって，上述のユーザーコンテキストの集合は，チャートタイトルの集合と同一であると考える．なお，提案法は，その他のユーザーコンテキスト（ユーザー属性，可視化シーンなど）についても，それぞれに関連する情報を用いることで考慮することが可能である．したがって，チャートタイトルをユーザーの可視化意図として利用することは，あくまでもユーザーコンテキストを対象にした研究の一例である．

3.2 インサイト評価モデル

ユーザーコンテキストを考慮したインサイト評価モデルを導入する．本問題設定における尤度は式 (3) のように与えられる．

$$L(\mathbf{w}) = \prod_{i=1}^{N} \prod_{V_{ik} \in \mathbb{V}_i, u_{ik} \in \mathcal{U}_i} p(Y_{ik} | \mathcal{M}(V_{ik}, u_{ik}; \mathbf{w})), \tag{3}$$

$\mathcal{M}$ は可視化 $V_{ik} \in \mathbb{V}_i = \mathbb{V}_i^+ \cup \hat{\mathbb{V}}_i^-$ およびユーザーコンテキスト $u_{ik} \in \mathcal{U}_i$ を入力としてその評価値 $\hat{Y}_{ik}$ を与える教師ありのレコメンドモデル，$\mathbf{w}$ はモデルパラメータである．

$$\hat{Y}_{ik} = \mathcal{M}(V_{ik}, u_{ik}; \mathbf{w}) \in [0, 1]. \tag{4}$$

Y_{ik} は ポジティブサンプルとネガティブサンプルを区別するラベルであり，式 (5) のように与えられる．

$$Y_{ik} = \begin{cases} 1 & (V_{ik} \in \mathbb{V}_i^+), \\ 0 & (V_{ik} \in \hat{\mathbb{V}}_i^-). \end{cases} \tag{5}$$

$p(Y_{ik} | \mathcal{M}(V_{ik}, u_{ik}; \mathbf{w}))$ は Bernoulli 分布で記述可能であるため，式 (3) は以下のように書き換えることができる．

$$L(\mathbf{w}) = \prod_{i=1}^{N} \Bigg(\prod_{V_{ik} \in \mathbb{V}_i^+, u_{ik} \in \mathcal{U}_i} \mathcal{M}(V_{ik}, u_{ik}; \mathbf{w}) \prod_{V_{ik} \in \hat{\mathbb{V}}_i^-, u_{ik} \in \mathcal{T}_i} (1 - \mathcal{M}(V_{ik}, u_{ik}; \mathbf{w})) \Bigg). \tag{6}$$

最後に，式 (6) の対数をとり，さらに符号を反転させることで，学習に用いる損失関数 $E(\mathbf{w})$ を定義する．

$$\begin{aligned} E(\mathbf{w}) &= -\sum_{i=1}^{N} \Bigg(\sum_{V_{ik} \in \mathbb{V}_i^+, u_{ik} \in \mathcal{U}_i} \log \mathcal{M}(V_{ik}, u_{ik}; \mathbf{w}) \\ &\qquad + \sum_{V_{ik} \in \hat{\mathbb{V}}_i^-, u_{ik} \in \mathcal{T}_i} \log (1 - \mathcal{M}(V_{ik}, u_{ik}; \mathbf{w})) \Bigg), \\ &= -\sum_{i=1}^{N} \sum_{V_{ik} \in \mathbb{V}_i, u_{ik} \in \mathcal{U}_i} \Big(Y_{ik} \log \mathcal{M}(V_{ik}, u_{ik}; \mathbf{w}) \\ &\qquad + (1 - Y_{ik}) \log (1 - \mathcal{M}(V_{ik}, u_{ik}; \mathbf{w})) \Big). \end{aligned} \tag{7}$$

式 (7) の損失関数を最小化するように学習されたレコメンドモデル $\mathcal{M}^*$ は，ポジティブサンプルと近い特徴を持った可視化に対して高いスコアを与えるインサイト評価モデルとして用いることができる．

3.3 ユーザーコンテキストを考慮したインサイトの評価

インサイト評価モデル $\mathcal{M}^*$ は，テストデータセット $\mathbf{X}_{\mathrm{test}} \notin \{\mathbf{X}_i\}_{i=1}^N$ におけるユーザーコンテキスト $u_t \in \mathcal{U}_{\mathrm{test}}$ と可視化 $V_t \in \mathbb{V}_{\mathrm{test}}$ を入力として，その評価値を与えることができる．したがって，式 (8) のように評価値の高い順番にテストデータの可視化を並び替えることが可能である．

$$\underset{V_t \in \mathbb{V}_{\mathrm{test}}}{\arg\mathrm{sort}}\, \mathcal{M}^*(V_t, u_t). \tag{8}$$

並び替えられた可視化のうち，より上位に位置する可視化はポジティブサンプルと同様の特徴を備えている有用な可視化である可能性が高い．したがって，上式の結果の top-k に絞り込んで提示することで，効率的なデータ探索を行うことができる．しかしながら，前述のように有用なチャートであったとしても，それがユーザーの求めるものである保証はない．そこで，本研究では式 (9) のように，新たにユーザーコンテキストと可視化の関連度 $\mathcal{R}^*$ を導入することで，有用かつユーザーコンテキストに関係する可視化が上位に来るように並び替える．

$$\underset{V_t \in \mathbb{V}_{\mathrm{test}}}{\arg\mathrm{sort}}\, \mathcal{R}^*(V_t, u_t) \mathcal{M}^*(V_t, u_t). \tag{9}$$

関連度 $\mathcal{R}^*$ は，ユーザーコンテキストと可視化の共起関係を学習することで得られる評価値であり，その計算方法の説明は 4.1.3 節で行う．

4 提 案 手 法

4.1 入力データのエンコーディング

各可視化は V_{ik} はその構成要素となる可視化属性の組み合わせ $\mathbf{X}_i^{(k)}$ と可視化形式 C_{ik} に分解可能であり，それぞれをベク

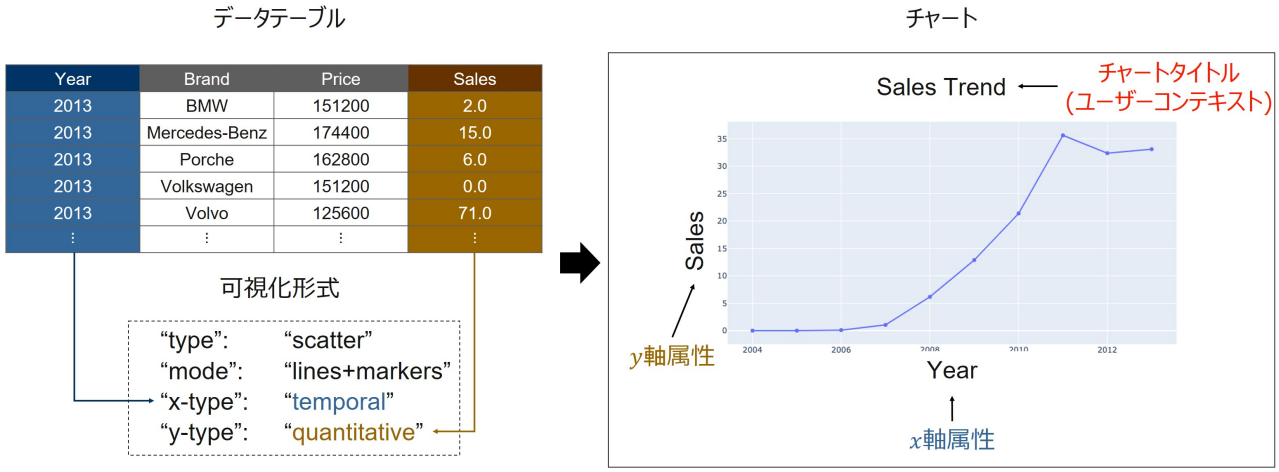

図 1　多次元データの可視化例．左上の図は複数の属性を含む多次元データを示しており，可視化属性は色付けされている．左下の図は可視化形式について示している．右図はこれらをチャートとして可視化したものを示している．

トル化したものがレコメンドモデル $\mathcal{M}$ への入力となる．また，ユーザーコンテキスト u_{ik} と可視化に利用される可視化属性名 $z_{ij} \in \mathbf{X}_i^{(k)}$ の関係性を考慮した評価が必要となるため，これらの関係性を表現するベクトルについても入力として与えられる．

4.1.1　可視化属性の特徴ベクトル化

可視化属性のベクトル表現は，関数 ψ を用いてエンコードすることで獲得する．ψ は，可視化属性ベクトル $\mathbf{x}_{ij} \in \mathbf{X}_i^{(k)}$ を入力として，p 次元の実ベクトルを返す．

$$\psi(\mathbf{X}) = \{\psi(\mathbf{x}) \in \mathbb{R}^p \mid \mathbf{x} \in \mathbf{X}\}. \tag{10}$$

特徴量の候補はいくつか考えられるが，本研究においては表 1 に列挙した統計量を並べたものを可視化属性の特徴ベクトルとして用いた．なお，本研究においては，x 軸及び y 軸に用いられる属性のみを考慮しており，$\mathbf{X}_i^{(k)}$ を構成する可視化属性は 2 つの場合のみを考えている．

4.1.2　可視化形式の埋め込み

f は可視化形式のベクトル表現を与える関数であり，ある可視化形式 C_{ik} を入力として，q 次元の実ベクトルを返す．

$$f(\mathcal{C}) = \{f(C) \in \mathbb{R}^q \mid C \in \mathcal{C}\}. \tag{11}$$

本研究では，f によるベクトル表現を C_{ik} ごとに計算された埋め込みベクトルをルックアップすることで獲得する．f によるベクトル表現の次元数 q は任意に設定可能だが，本研究では可視化属性の特徴ベクトルの次元と合わせて $q = p$ とした．

4.1.3　ユーザーコンテキストの埋め込み

前述のように，本研究ではチャートタイトルをユーザーコンテキストとみなし，$u_{ik} \in \mathcal{U}_i$ と可視化属性名 $z_{ij} \in \mathbf{X}_i^{(k)}$ の共起関係を考慮した埋め込みを行う．g はこれらのベクトル表現を与える関数であり，r 次元の実ベクトルを返す．

$$\begin{aligned} g(\mathcal{U}) &= \{g(u) \in \mathbb{R}^r \mid u \in \mathcal{U}\}, \\ g(\mathcal{Z}) &= \{g(z) \in \mathbb{R}^r \mid z \in \mathcal{Z}\}. \end{aligned} \tag{12}$$

チャートタイトルと可視化属性名の関連性を表現するベクトル

表 1　可視化属性の特徴量一覧．

Name	Equations
Num. values in attribute	$\lvert\mathbf{x}\rvert$
Num./frac. missing values	$\mathbf{s}$, $(\lvert\mathbf{x}\rvert - \mathbf{s})/\lvert\mathbf{x}\rvert$
Num. nonzeros, density	$\mathrm{nnz}(\mathbf{x})$, $\mathrm{nnz}(\mathbf{x})/\lvert\mathbf{x}\rvert$
Num. unique values	$\mathrm{card}(\mathbf{x})$
Q_1, Q_3	median of $\lvert\mathbf{x}\rvert/2$ smallest (largest) values
IQR	$Q_3 - Q_1$
Outlier LB $\alpha \in \{1.5, 3\}$	$\sum_i \mathbb{I}(x_i < Q_1 - \alpha\mathrm{IQR})$
Outlier UB $\alpha \in \{1.5, 3\}$	$\sum_i \mathbb{I}(x_i > Q_3 + \alpha\mathrm{IQR})$
Total outliers $\alpha \in \{1.5, 3\}$	$\sum_i \mathbb{I}(x_i < Q_1 - \alpha\mathrm{IQR}) + \sum_i \mathbb{I}(x_i > Q_3 + \alpha\mathrm{IQR})$
(α std) outliers $\alpha \in \{2, 3\}$	$\mu_{\mathbf{x}} \pm \alpha\sigma_{\mathbf{x}}$
Spearman (ρ, p-val)	$\mathrm{spearman}(\mathbf{x}, \pi(\mathbf{x}))$
Kendall (τ, p-val)	$\mathrm{kendall}(\mathbf{x}, \pi(\mathbf{x}))$
Pearson (r, p-val)	$\mathrm{pearson}(\mathbf{x}, \pi(\mathbf{x}))$
Min, max, range, median	$\min(\mathbf{x})$, $\max(\mathbf{x})$, $\max(\mathbf{x}) - \min(\mathbf{x})$, $\mathrm{med}(\mathbf{x})$
Geometric Mean	$\lvert\mathbf{x}\rvert^{-1}\Pi_i x_i$
Harmonic Mean	$\lvert\mathbf{x}\rvert / \sum_i \frac{1}{x_i}$
Mean, Stdev, Variance	$\mu_{\mathbf{x}}, \sigma_{\mathbf{x}}, \sigma_{\mathbf{x}}^2$
Skewness, Kurtosis	$\mathbb{E}(\mathbf{x} - \mu_{\mathbf{x}})^3/\sigma_{\mathbf{x}}^3$, $\mathbb{E}(\mathbf{x} - \mu_{\mathbf{x}})^4/\sigma_{\mathbf{x}}^4$
Moments, k-stat.	–
Quartile Dispersion Coeff.	$\frac{Q_3 - Q_1}{Q_3 + Q_1}$
Median Absolute Deviation	$\mathrm{med}(\lvert\mathbf{x} - \mathrm{med}(\mathbf{x})\rvert)$
Avg. Absolute Deviation	$\frac{1}{\lvert\mathbf{x}\rvert}\exp^T \lvert\mathbf{x} - \mu_{\mathbf{x}}\rvert$
Coeff. of Variation	$\sigma_{\mathbf{x}}/\mu_{\mathbf{x}}$
Efficiency ratio	$\sigma_{\mathbf{x}}^2/\mu_{\mathbf{x}}^2$
Variance-to-mean ratio	$\sigma_{\mathbf{x}}^2/\mu_{\mathbf{x}}$
Signal-to-noise ratio (SNR)	$\mu_{\mathbf{x}}^2/\sigma_{\mathbf{x}}^2$
Entropy, Norm. entropy	$H(\mathbf{x}) = -\sum_i x_i \log x_i$, $H(\mathbf{x})/\log_2 \lvert\mathbf{x}\rvert$
Gini coefficient	–
Quartile max gap	$\max(Q_{i+1} - Q_i)$
Histogram prob. dist.	$\mathbf{p}_h = \frac{\mathbf{h}}{\mathbf{h}^T\mathbf{e}}$

表現の方法はいくつか考えられるが，本研究においては，テストデータも含めた可視化コーパス全体を使って，各可視化におけるユーザーコンテキスト，x 軸属性名，y 軸属性名を 1 つの文としてみなしたデータセットを作成し，これを用いて Word2Vec [19] を学習することで，g によるベクトル表現を獲得する．Word2Vec にはいくつかのモデルが存在するが，本研究では Continuous Bag-of-Words + Negative Sampling のモデルを用いて $r = 100$ の分散表現を獲得した．得られた埋め込みベクトルを用いて計算されるユーザーコンテキストと可視化

属性名のコサイン類似度は，共通のコンテキストを持つ場合に高い値を示す．すなわち，可視化コーパス上で共起している場合には相対的に高い値をとり，ユーザーコンテキストと関係する可視化属性を見つけるのに有効な指標となるため，これを式 (9) における関連度 $\mathcal{R}^*$ として用いる．

g によるベクトル表現については，fastText [20] や BERT [21] などの先進的な言語モデルも利用可能である．しかしながら，これらの事前学習された言語モデルは，ユーザーコンテキストと可視化属性名が可視化コーパス上で共起していない場合でも，高いコサイン類似度を与えてしまう場合がある．これにより，後述の実験におけるモデルの評価において不都合を生じさせる可能性があるため，本研究においては，可視化コーパスにおけるチャートタイトルと可視化属性名の共起関係のみに注目し，上述のようなアプローチを取ることにした．なお，実用上の観点では，事前学習済の言語モデルを用いることで，それらのモデルが獲得した semantic な関係性に基づいて，ユーザーコンテキストと関連性の高い可視化を発見することが可能である．

4.1.4 モデルネットワーク

3.3 で導入したユーザーコンテキストを考慮したインサイトの評価を実現するため，図 2 に示すネットワークで構成されたニューラルネットワークを導入する．

図 2(a) は提案モデルの学習時について示している．モデルは，学習データセットにおける可視化属性ベクトル $\mathbf{x}_{ia}, \mathbf{x}_{ib} \in \mathbf{X}_i^{(k)}$，可視化形式 C_{ik}，ユーザーコンテキスト u_{ik} に関する以下 4 つのベクトルを入力とし，式 (7) の損失関数を最小化するようにニューラルネットワーク部分の学習が進められる．

- x 軸属性の特徴ベクトル $\psi(\mathbf{x}_{ia})$
- y 軸属性の特徴ベクトル $\psi(\mathbf{x}_{ib})$
- 可視化形式の埋め込みベクトル $f(C_{ik})$
- ユーザーコンテキストの埋め込みベクトル $g(u_{ik})$

$\psi(\mathbf{x}_{ia})$, $\psi(\mathbf{x}_{ib})$, $f(C_{ik})$, $g(u_{ik})$ が最初に入力される attention 層は，$g(u_{ik})$ をクエリ ($Q \in \mathbb{R}^r$), $\psi(\mathbf{x}_{ia})$, $\psi(\mathbf{x}_{ib})$, $f(C_{ik})$ の各特徴量のインデックスをキー ($K \in \mathbb{R}^r$), $\psi(\mathbf{x}_{ia})$, $\psi(\mathbf{x}_{ib})$, $f(C_{ik})$ の各特徴量をバリュー ($V \in \mathbb{R}^{p=q}$) とみなした source-target dot-product attention である．

$$\text{attention}(Q, K, V) = \text{softmax}\left(QK^T\right)V \qquad (13)$$

この attention 層はユーザーコンテキストごとに重視する特徴が異なる可能性を考慮して導入された機構であり，これが無くてもインサイト評価モデルとしては機能する．しかしながら，ユーザーが重要と考える特徴に注目することで，ユーザーコンテキストに関係するインサイトをより高く評価する効果を与えるため，有益な機構であると考えられる．図 3 は，可視化属性の特徴ベクトルについて導入された attention 層の例を示したものである．ユーザーコンテキストをクエリとして，各特徴量のキー及びバリューについての重みが学習されることにより，自己決定的な重み付けが可能となる．

図 2(b) は提案モデルの推論時について示している．モデルは，テストデータセットにおける可視化属性ベクトル及び可視化属性名 $\mathbf{x}_{ta}, \mathbf{x}_{tb}, z_{ta}, z_{tb} \in \mathbf{X}_{\text{test}}$，可視化形式 $C_t \in \mathcal{C}_{\text{test}}$，ユーザーコンテキスト $u_t \in \mathcal{U}_{\text{test}}$ に関する以下の 6 つのベクトルを入力とし，これらに関するインサイトの評価を与える．

- x 軸属性の特徴ベクトル $\psi(\mathbf{x}_{ta})$
- y 軸属性の特徴ベクトル $\psi(\mathbf{x}_{tb})$
- 可視化形式の埋め込みベクトル $f(C_t)$
- ユーザーコンテキストの埋め込みベクトル $g(u_t)$
- x 軸属性名の埋め込みベクトル $g(z_{ta})$
- y 軸属性名の埋め込みベクトル $g(z_{tb})$

図 2(b) に示すように，ニューラルネットワーク部分の出力は，ユーザーコンテキストと可視化属性名の埋め込みベクトルのコサイン類似度の最大値によってキャリブレーションされる．4.1.3 で述べたように，g が与えるベクトル表現のコサイン類似度の大きい場合には，u_t に関係する属性である可能性が高いと考えられる．ユーザーコンテキスト u_t は可視化コーパス中の任意のチャートタイトル中から自由に選択することが可能であり，図 2 のモデルの出力結果の top-k は，有用かつユーザーコンテキスト u_t に関係する可視化である可能性が高い．

5 実　　験

5.1 データセットの作成方法

本研究では，VizML [4] の公開している Plotly データセットを用いた．上記のデータセットは，Plotly Community Feed で公開されているデータ とその可視化を収集することで作成されたものである．VizML はいくつかのサイズのデータセットを提供しているが，今回は中規模のデータセット（$100K$ のデータ-可視化ペア）を採用し，以下の作業手順でモデルの学習及びテストに用いるデータセットを作成した．

（1） Plotly データセットのうち，チャートタイトルと属性名が存在しないデータを除外する．

（2） Plotly データセットのコーパスをデータソース単位で学習用とテスト用に分割する．

（3） 学習用およびテスト用のポジティブサンプルをそれぞれのデータソースに対応するコーパスから取得する．

（4） 学習用およびテスト用のネガティブサンプルをポジティブサンプルと同数作成する．

（5） テスト用のネガティブサンプルとポジティブサンプル纏め多数のレコードを含むテストデータセットを作成する．

上記のステップ 1 は，本研究で用いるモデルがチャートタイトルと属性名を必要としているために行われる処理である．ステップ 2 では，データソースの単位で学習用とテスト用に分割しているが，これは可視化コーパス中には同一のデータソースから作成された可視化が複数存在し，無作為な分割はデータリークを発生させる恐れを伴うため，データソースの単位で学習用とテスト用に分割することによって，これを回避している．ステップ 4 では，ポジティブサンプルの構成要素の一部を同一のデータソースの項目とランダムに入れ換えることで，ネガティブサンプルを生成している．ステップ 5 では，後述の実験で用いるテストデータが複数のユーザーコンテキストに関するデータを含むようにするための処理である．これによって，

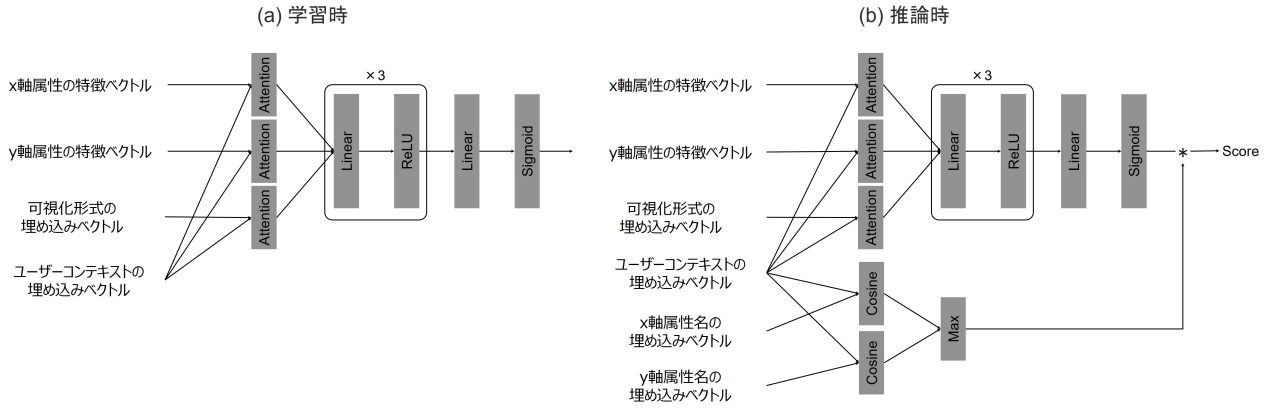

図 2　提案モデルのネットワークの概要図．左図はモデルの学習時，右図は推論時を示している．

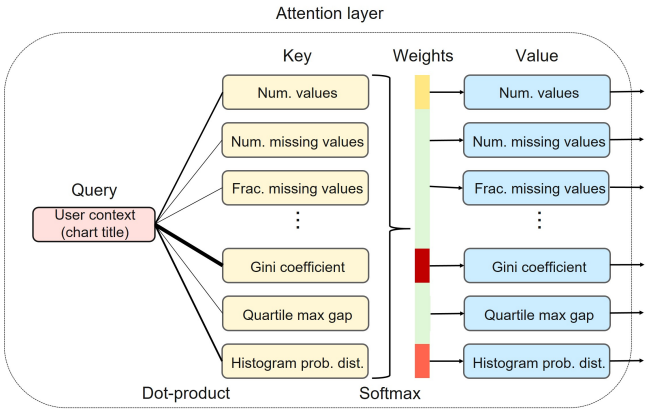

図 3　属性の特徴ベクトルについて導入された attention 層の概念図．テキストボックスは，記載の変数の線形変換によって与えられるベクトルを意味している．ユーザーコンテキストをクエリとして，各特徴量のキー及びバリューについての重みが学習される．

様々な種類のデータから目的のユーザーコンテキストに関係する可視化を発見する実験が可能になる．

5.2　実験条件

以下の実験結果では，K-分割交差検証によって複数のデータセットを作成し，各データセットにおいて学習及び評価した結果の平均値を表示している．また，推論時にはテスト用データセットからランダムに選んだチャートタイトルをユーザーコンテキストとして利用し，ポジティブサンプルであり，かつ，選択されたユーザーコンテキストと実際に共起している x 軸属性及び y 軸属性の両方が可視化される場合には正解 (1)，それ以外については全て不正解 (0) とみなす．ユーザーコンテキストについては，各交差検証において 100 個ランダムに選び，それらの評価指標の平均を各交差検証の評価値として用いる．

以下の実験では，提案法を含む 5 つのモデルについての結果を比較する．提案法は前述のように，図 2 のネットワークを用いて学習及び推論を行うモデルである．提案法 (w/o cos sim) は提案法のモデルからコサイン類似度によるキャリブレーションを除いたもの，提案法 (w/o attention) は提案法のモデルから attention 層を除いたものである．MLP は，Qian らのモデル [8] のうち，deep モデルの部分のみを用いたものであり，ユーザーコンテキストについては全く考慮していない．Random は，各可視化についてランダムに評価値を返すモデルである．

表 2　MRR 及び MAP を比較した結果．

Methods	MRR	MAP
Random	0.082386	0.028789
MLP	0.022496	0.059877
提案法 (w/o cos sim)	0.047931	0.105458
提案法 (w/o attention)	0.141830	0.207695
提案法	**0.187612**	**0.246399**

5.3　実験結果

モデルの提示するランキングの上位に正解となる可視化が含まれているかを確認するため，MRR (Mean Reciprocal Rank), MAP (Mean Average Precision), nDCG (Normalized Discounted Cumulative Gain) の 3 つの評価指標を計算した．これらは，あるクエリに対して出力されたランキングにおいて，正解となるデータがどれくらい良く拾えているかを測るレコメンド指標である．本研究においてはユーザーコンテキストがクエリに相当する．

表 2 は，Random, MLP, 提案法の MRR と MAP を比較した結果である．提案法は，MRR と MAP のどちらについても全てのモデルを上回る性能が発揮できていることが分かる．提案法の MRR はおよそ 0.19 であるから，平均して top-6 の中に 1 つは目的のインサイトを与える可視化が含まれていることが分かる．興味深いことに MLP については，MAP は Random を多少上回るが，MRR は Random よりもむしろ低い結果となる．すなわち，MLP はポジティブサンプルを高く評価することは可能だが，ユーザーコンテキストと関連する可視化を見つけることは困難であることが示されている．提案法 (w/o cos sim) の MRR 及び MAP は，MLP よりは高いが提案法よりはかなり低く，コサイン類似度によるキャリブレーションはユーザーコンテキストと関連のある可視化を見つけるのに非常に有効であることが分かる．また，提案法 (w/o attention) の MRR 及び MAP についても提案法には及んでいないことから，attention 層の導入も有効に機能していることが分かる．

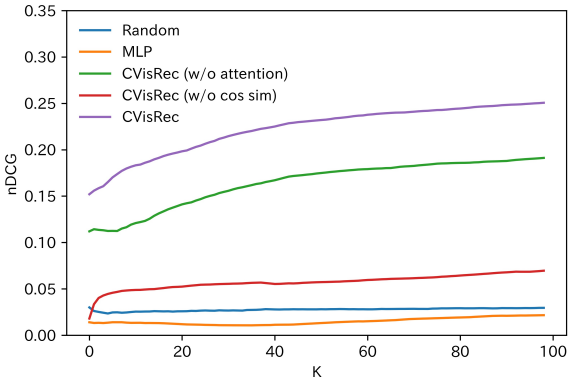

図 4　nDCG を比較した結果.

図 4 は，Random, MLP, 提案法の nDCG をいくつかの閾値 K で比較した結果である．提案法はいずれの K においてもその他の手法を上回っていることが分かる．MLP については，やはり Random とほぼ変わらず，K を変えても nDCG はあまり増加しない．MRR 及び MAP の場合と同様に，提案法 (w/o cos sim)，提案法 (w/o attention) は Random や MLP の結果よりも良いが，提案法には及ばないことから，コサイン類似度によるキャリブレーション及び attention 機構の導入は，どちらも有効に機能していることが分かる．

6　結　　論

本研究では，ユーザーコンテキストを考慮したインサイトの評価を可能とするモデルを開発した．また，Plotly データセットを用いた実験を行い，提案法の有効性を示した．提案法は，ユーザーコンテキストと可視化属性の共起関係に基づくキャリブレーションと，ユーザーコンテキストをクエリとみなした attention 機構の導入により，有用かつユーザーコンテキストに関係するチャートを効率的に発見することが可能である．

本研究においては，モデルの評価を行う都合から，可視化コーパスを用いて学習した埋め込みベクトルからユーザーコンテキストと属性の関連度を計算した．しかしながら，実用上はこの方式に限定する必要はなく，言語モデルが獲得したベクトル表現を利用することで，semantic な関係に基づいたチャートを発見することも可能である．

文　　献

[1] Doris Jung-Lin Lee. Insight machines: The past, present, and future of visualization recommendation, 2020.

[2] Po-Ming Law, Alex Endert, and John Stasko. What are data insights to professional visualization users? In *2020 IEEE Visualization Conference, IEEE VIS'20*, pp. 181–185. IEEE, 2020.

[3] Yuyu Luo, Xuedi Qin, Nan Tang, and Guoliang Li. Deepeye: Towards automatic data visualization. In *Proceedings of IEEE International Conference on Data Engineering, ICDE'18*, pp. 101–112, 2018.

[4] Kevin Hu, Michiel A Bakker, Stephen Li, Tim Kraska, and César Hidalgo. Vizml: A machine learning approach to visualization recommendation. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI'19*, pp. 1–12, 2019.

[5] Victor Dibia and Çağatay Demiralp. Data2vis: Automatic generation of data visualizations using sequence-to-sequence recurrent neural networks. *IEEE Computer Graphics and Applications, IEEE CG&A*, Vol. 39, No. 5, pp. 33–46, 2019.

[6] Mengyu Zhou, Tao Wang, Pengxin Ji, Shi Han, and Dongmei Zhang. Table2analysis: Modeling and recommendation of common analysis patterns for multi-dimensional data. In *Proceedings of the AAAI Conference on Artificial Intelligence, AAAI'20*, Vol. 34, pp. 320–328, 2020.

[7] Mengyu Zhou, Qingtao Li, Xinyi He, Yuejiang Li, Yibo Liu, Wei Ji, Shi Han, Yining Chen, Daxin Jiang, and Dongmei Zhang. Table2charts: Recommending charts by learning shared table representations. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, KDD'21*, pp. 2389–2399, 2021.

[8] Xin Qian, Ryan A Rossi, Fan Du, Sungchul Kim, Eunyee Koh, Sana Malik, Tak Yeon Lee, and Joel Chan. Learning to recommend visualizations from data. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, KDD'21*, pp. 1359–1369, 2021.

[9] Kevin Hu, Snehalkumar'Neil'S Gaikwad, Madelon Hulsebos, Michiel A Bakker, Emanuel Zgraggen, César Hidalgo, Tim Kraska, Guoliang Li, Arvind Satyanarayan, and Çağatay Demiralp. Viznet: Towards a large-scale visualization learning and benchmarking repository. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI'19*, pp. 1–12, 2019.

[10] Yuyu Luo, Nan Tang, Guoliang Li, Chengliang Chai, Wenbo Li, and Xuedi Qin. Synthesizing natural language to visualization (nl2vis) benchmarks from nl2sql benchmarks. In *Proceedings of the 2021 International Conference on Management of Data, ICDM'21*, pp. 1235–1247, 2021.

[11] Yu-Rong Cao, Jia-Yu Pan, and Wen-Chieh Lin. User-oriented generation of contextual visualization sequences. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems, CHI'20*, pp. 1–8, 2020.

[12] Belgin Mutlu, Eduardo Veas, and Christoph Trattner. Vizrec: Recommending personalized visualizations. *ACM Transactions on Interactive Intelligent Systems, TiiS*, Vol. 6, No. 4, pp. 1–39, 2016.

[13] Xin Qian, Ryan A Rossi, Fan Du, Sungchul Kim, Eunyee Koh, Sana Malik, Tak Yeon Lee, and Nesreen K Ahmed. Personalized visualization recommendation. *arXiv preprint arXiv:2102.06343*, 2021.

[14] Manasi Vartak, Sajjadur Rahman, Samuel Madden, Aditya Parameswaran, and Neoklis Polyzotis. Seedb: Efficient data-driven visualization recommendations to support visual analytics. In *Proceedings of the VLDB Endowment, VLDB'15*, Vol. 8, p. 2182, 2015.

[15] Çağatay Demiralp, Peter J Haas, Srinivasan Parthasarathy, and Tejaswini Pedapati. Foresight: Recommending visual insights. In *Proceedings of the VLDB Endowment, VLDB'17*, Vol. 10, pp. 1937–1940, 2017.

[16] Rui Ding, Shi Han, Yong Xu, Haidong Zhang, and Dongmei Zhang. Quickinsights: Quick and automatic discovery of insights from multi-dimensional data. In *Proceedings of the 2019 International Conference on Management of Data, SIGMOD'19*, pp. 317–332, 2019.

[17] Po-Ming Law, Alex Endert, and John Stasko. Characterizing automated data insights. In *2020 IEEE Visualization Conference, IEEE VIS'20*, pp. 171–175, 2020.

[18] Atsuki Maruta and Makoto P Kato. Intent-aware visualization recommendation for tabular data. In *International Conference on Web Information Systems Engineering*, pp. 252–266. Springer, 2021.

[19] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations 2013, ICLR 2013*, 2013.

[20] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics, TACL*, Vol. 5, pp. 135–146, 2017.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *North American chapter of the Association for Computational Linguistics, NAACL'19*, 2019.