

マルチユーザーチャットからの個人の知識や交流傾向を考慮した ユーザ埋め込み表現学習

吉田 司[†] 大塚 淳史[†] 野本 済央[†] 深山 篤[†] 中村 高雄[†]

[†] 日本電信電話株式会社 NTT デジタルツインコンピューティング研究センタ

〒 108-0023 東京都港区芝浦 3 丁目 4-1 グランパークタワー 33F

E-mail:

†{tsukasa.yoshida.zu,atsushi.otsuka.vs,narichika.nomoto.ds,atsushi.fukayama.ap,takao.nakamura.vp}@hco.ntt.co.jp

あらまし 本研究ではオンライン上のチャット履歴から個人の特性を抽出し、ユーザのベクトル表現を構成することを目的とする。特に個人の知識や興味の特徴と他人との交流傾向をベクトルに埋め込むことを目標とする。従来研究では、ユーザの SNS などの投稿テキストやアクティビティから個人の知識や興味などといった特性をベクトル化する手法や、他人とのフォロー関係のグラフ構造から交流傾向を抽出する手法などが別々に議論されてきた。しかしながらこれらは会社や学校におけるチーミングやペアマッチングなどへの利用シーンではどちらも重要な要素であり、双方の情報を合わせて適切にベクトルに埋め込むことが求められる。そこで本研究ではチャット履歴からチャットパッチを構成し、パッチの一部がマスクされたデータを復元するニューラルネットワークの中間層の出力を用いて、ユーザのベクトル表現を構成する手法を提案する。ベクトルに線形写像を施すことで個人の特性を抽出できるかを確認する実験により、提案手法のベクトルに個人の知識や興味の特徴、他人との交流傾向の特徴が同時に埋め込まれていることを示す。

キーワード ユーザエンベディング, 特徴量抽出, ベクトル表現, チーミング, マッチング

1 はじめに

近年オンライン上での人々の活動が増加している。たとえば SNS を通じたユーザの情報発信やチャットシステムを用いた多人数のコミュニケーションが日々大量に行われている。これらのデータを活用することでユーザ興味を理解、行動予測、状態変化の検出、商品や広告のレコメンドや他ユーザとのマッチング戦略の策定などが行えると考えられる。

マッチング戦略における一つの問題として、組織におけるチーミングの問題がある。チーミングとは人々のグループ活動においてチームのメンバー構成やグループ構造を与えること、チームを作ることを意味する。組織において効率的な共同作業を行うためには適切なチーミングが必要になる。社会科学分野におけるチーミングの研究ではチームの評価としてどんな尺度が重要になるかの研究や、どのようなメンバーを集めればパフォーマンスが向上するかなど、複数の角度から研究が進められている [1-4]。チームにおけるメンバーの組み合わせの最適性の研究では個性が重要になるが、その方法論は確立されておらず、社会科学の分野では個性はアンケートを用いて評価することが主流となっている。しかしながら個性は時間とともに変化すると考えられるため、日々集められるデータから自動的に収集できることが望ましい。

チーミングを行うためにユーザ特性を SNS やチャットなどの個人のデータから数値化することが考えられるが、どのように数値化するかが問題となる。この解決方法の一つとして、埋め

込み学習が存在する。ある対象の埋め込み (エンベディング) とは対象をその特徴を表すベクトルに割り当てる、もしくは変換する操作のことである。すなわちユーザエンベディングとは各ユーザに対してその特徴を示すベクトルを割り当てることを示す。これまで上記の目的のためにユーザエンベディングの研究が多数行われてきた [5,6]。特に SNS の投稿テキストや投稿画像などに対して自然言語処理や画像処理の埋め込み手法を活用することでユーザのベクトル表現を獲得することが盛んである。

本研究では会社や学校におけるチーミングやペアマッチング戦略のためにユーザ特性をベクトル化することを目的とする。個人のオンライン上の活動において、特にマルチユーザーチャットのデータは個人の興味や交流の傾向を多く含んでいると考えられる。本論文ではチャンネルごとに区分けされたチャットシステムにおけるチャットデータに焦点を当てる。

従来研究では、ユーザの投稿テキストやアクティビティから個人の知識や興味などといった特性をベクトル化する手法 [7-9] や、他人とのフォロー関係のグラフ構造から交流特性を抽出する手法 [10,11] などは別々に議論されてきた。しかしながらこれらはマッチングやチーミングへの利用シーンではどちらも重要な要素であり、チャットデータから双方の情報を合わせて適切にベクトルに埋め込むことが求められる。

そこで本研究では個人の知識や興味の特徴と他人との交流傾向をベクトルに埋め込むことを目標とする。これを達成するために、本論文ではチャット履歴のデータベースからチャットパッチを構成し、チャットパッチから複数のタスクを同時に解

ニューラルネットワークの中間層の出力を用いて、ユーザのベクトル表現を構成する手法を提案する。ここでチャットパッチとは、チャットのチャンネル情報とチャット履歴の一部を取り出して構成したデータフレームのことである。提案手法で解く具体的なタスクは、一部をマスクしたチャットパッチから元のチャットパッチを復元するタスク、チャットパッチのメンバーリストからそれぞれの発言量を予測するタスクの2つである。チャットパッチにはユーザとユーザ、ユーザと発言の共起性情報が含まれていることから、これに対して上記の2つのタスクを実施することでユーザの知識や興味、交流傾向を埋め込むことができる。本論文ではベクトルに線形写像を施すことで個人の特徴を抽出できるかを確認する実験により、提案手法のベクトルに個人の知識や興味の特徴、他人との交流傾向の特徴が同時に埋め込まれていることを示す。

2 関連研究

自然言語処理の分野では単語や文章のベクトルに埋め込む方法としてこれまで様々な手法が提案されてきた。近年ではニューラルネットワークを用いた手法が多数提案されており、特にその中でも BERT が注目されている [12]。BERT は大量のデータを用いて事前学習を行った後に、タスクに応じて適切なファインチューニングを行うことで様々なタスクで高い性能を発揮することが明らかにされている。

BERT は Attention 機構を用いた Transformer [13] の構造をベースとしている。同様に Transformer モデルの構造を応用したモデルとして XLNet [14], RoBERTa [15], ALBERT [16], T5 [17], GPT-3 [18] などの高い性能を誇るモデルが複数発表されている。自然言語処理のみならず、Transformer モデルは画像処理や映像処理の分野にも活用されており、ViT [19], ViViT [20] といった高い性能を持つモデルが発表されている。このような有効性・発展性を考慮して本研究でも Transformer の構造を用いたニューラルネットを使用する。

また、グラフエンベディングにおける分野でもニューラルネットワークを用いた埋め込み手法が盛んに研究されている。例えば、グラフのノード埋め込みを行うモデルでは Skip-gram [21] を用いる DeepWalk [22] や Node2Vec [23], Autoencoder を用いる SDNE [24], Transformer を用いる GTN [25] などが存在する。これらを使用するにはグラフ構造を定義する必要があり、この定義によってエンベディング結果は大きく変化するため、目的に応じてグラフ構造の設計方法を考える必要がある。また、本研究ではチャットデータからユーザの知識や興味と交流の傾向の特徴をベクトルに埋め込むことを考えるが、グラフ構造を用いて表現しやすいのは誰と誰が交流しているかといった交流傾向が主であり、ユーザの知識や興味を埋め込むことは容易ではない。そのため、本論文ではグラフエンベディング手法には焦点を当てない。

チャットのテキスト情報から、ユーザのベクトル表現を得る単純な手法として LSI (Latent Semantic Indexing) [26] が考えられる。LSI は使用単語の頻度情報からベクトルを構成す

る手法である。これをユーザエンベディングに使用した場合、ユーザが使用する単語の情報からユーザの知識や興味といった特徴を埋め込むことができると考えられる。また、ユーザが使用する単語の語彙量からユーザが普段どれだけ様々なトピックについて話すかなどといった情報を推論することができ、ここからユーザがどのくらい活発に話すか、どれくらい様々なチャンネルに参加するかなどの交流特性を埋め込むことができると考えられる。同様に LDA (Latent dirichlet allocation) [27] や Doc2Vec [28] などでも LSI と類似した情報を埋め込むことができると考えられる。本研究では最も単純な LSI を比較手法として使用する。

3 提案手法

この章では本研究で提案するチャットシステムにおけるチャット履歴データからユーザ埋め込み学習を行う手法について説明する。ユーザのベクトル表現は、提案するニューラルネットの学習を実施した後に、ニューラルネットに対してユーザ ID を入力した際の中間層の出力とする。

3.1 ユーザ埋め込みの学習方法

提案手法の学習方法を図 1 に示す。

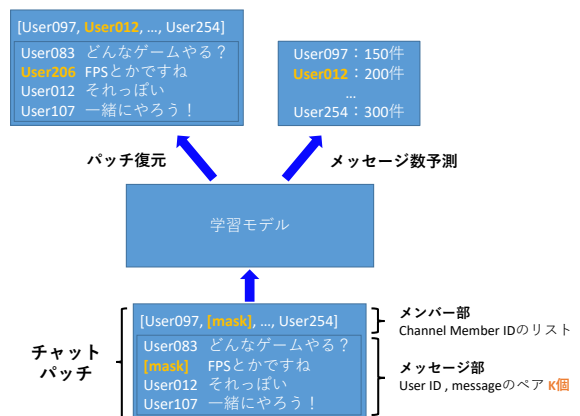


図 1 ユーザエンベディングのための提案手法における学習方法の概要。チャットパッチを入力し、パッチ復元とメッセージ数予測のタスクを解く学習を行う。

提案手法ではチャンネルごとに分けられたチャットシステムのチャット履歴のデータベースから、まずチャットパッチを作成する。チャットパッチはメンバー部とメッセージ部の2つのブロックで構成される。メンバー部はチャンネルメンバー ID リストで構成される。メッセージ部はチャンネルで発言されたメッセージの発言したユーザ ID とメッセージのテキストのペア K 個で構成される (図 1 の例では $K = 4$)。ただしこのメッセージ部は、あるチャンネル上で発言されたメッセージを時系列順に並べた際に連続するメッセージ K 個で構成される。

この作成したチャットパッチのユーザ ID の一部をマスクし

たデータを入力とし、2つのタスクを同時に解く。

■ **パッチ復元**：チャットパッチ $\mathbf{c} \in \mathcal{C}$ の要素の一部を置き換えた $\tilde{\mathbf{c}}$ から \mathbf{c} を復元するタスク。

■ **メッセージ数予測**：チャットパッチのユーザーリスト $\mathbf{u} \in \mathcal{L}(U)$ に対して、チャットパッチのチャンネル上でのユーザー u_i の発言回数 $t_i \in \mathbb{Z}_{\geq 0}$ ($i = 1, \dots, \dim(\mathbf{u})$) を推定するタスク。ここで \mathcal{C} は記号集合 \mathcal{S} の元からなる長さ M の有限列の全体集合であり、すなわちチャットパッチの全体集合を表す。また、 $\mathcal{L}(U)$ は U の元からなる順列の全体集合であり、 U はユーザーの全体集合を表す。

このタスクの学習によって、学習モデルが次のような情報を学習し、中間層の出力としてそれらの情報を保持することが期待できる。

■ **知識特性**：あるユーザーがどんな知識を持っているか、あるユーザーがどんなものに興味があるかなどの情報

■ **交流特性**：あるユーザーがどの程度活発にチャンネルに参加するか、どのユーザーが誰とどこでどれだけ話しやすいかなどの情報

ここで期待される効果は次のような仮説から説明される。

- ユーザーの知識や興味はユーザーの発言に生じる。この共起性からチャットパッチのメッセージ部の中のマスクされたユーザー ID をチャットパッチ上の情報から予測するタスクを行うことで、そのユーザーの知識特性を獲得することができる。

- ある同一のものに対して知識や興味ある人々はグループとして集まりやすい。この共起性からチャットパッチのメンバー部の中のマスクされたユーザー ID を予測するタスクを行うことで、どのようなユーザーたちが共通の知識や興味を持っているかといった知識特性を獲得できる。

- ユーザーの活発性はチャンネルの参加回数やチャンネル内での発言割合から特徴付け、分類することができる（例えば少人数ではよく話すが、大人数では話づらいなど）。ここからあるチャンネル内で誰がどの程度話すかを学習することで、活発性の交流特性を獲得できる。

- ユーザーは親しい、もしくは交流しやすい人のメッセージの直後にメッセージを送りやすい。この共起性からチャットパッチのメッセージ部の中のマスクされたユーザー ID を復元することで誰の発言に対して誰が反応しやすいかを学習し、交流特性を獲得できる。

3.2 モデル構成

ここでは前節で与えた学習を行うためのニューラルネットワークの構成を述べる。

全体構成

構成図を図 2, 3 に示す。モデルは大まかにエンベディング部・Transformer-Encoder 部・ヘッダ部の 3 つのブロックから構成され、ヘッダ部はさらに 3 つのブロックから構成される。エンベディング部・Transformer-Encoder 部の基本的な構成は BERT と同様であり、入力とヘッダ及び出力のみが異なる。ユーザーのベクトル表現は、学習済みモデルのエンベディング部へユーザー

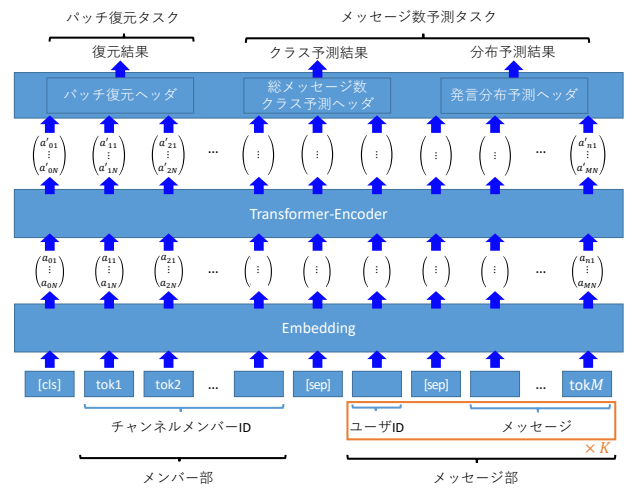


図 2 提案モデルのアーキテクチャ。エンベディング部・Transformer-Encoder 部・ヘッダ部の 3 つのブロックから構成される。ユーザーエンベディングの結果はエンベディング部の出力を用いる。

ID のトークンを入力した際の出力ベクトルとして得る。

基本的な構成として BERT モデルを利用する理由は次の通りである。まず BERT は自然言語処理における単語の埋め込みにおいて利用される。単語は一般的に文章によって意味が異なり、これを考慮して BERT は単語に対して常に一定の出力値を返さず、文章によって異なる出力値を出すように設計されている。対して人間も単語と同様に、場面によって個人の発言内容、立場や役割が異なるため、この特性を考慮して本研究では BERT モデルを用いる。しかしながら、これではユーザーを表現するベクトルの結果が逐一変わってしまう。そこで本研究ではニューラルネットワークの最下層のエンベディング部にユーザー ID を入力した際の出力ベクトルを埋め込み結果として使用する。

提案モデルのヘッダ部はパッチ復元、発言分布予測、総メッセージ数クラス予測の 3 つのブロックからなる。パッチ復元ヘッダはパッチ復元タスクに対応し、チャットの復元結果を出力する。このヘッダは BERT における Masked Language Model のヘッダと同様の構成である。一方で残る 2 つのヘッダはメッセージ数予測タスクに対応している。発言分布予測ヘッダはチャットパッチのメンバー部にある各メンバーのチャンネルでの発言確率を出力し、総メッセージ数クラス予測ヘッダはチャットパッチが構成されたチャンネルにおける総メッセージ数を予測し、あらかじめ設定されたメッセージ数に対応するクラスの番号を出力する。

このモデルの入出力の流れは次の通りである。

(1) チャットパッチを最小単位のトークンに分割する。ただし、ユーザー ID は 1 つのトークンとして扱い、メッセージ部のテキスト情報は SentencePiece [29] を用いてサブワードに分割する。

(2) 分割したトークンを 1 列に並べたトークン列を構成する。このときチャンネルメンバー ID リスト、ユーザー ID、メッセージの各区切れ位置に特殊トークン “[sep]” を挿入しセパレートをする。また、トークン列の最大の長さを M とし、作成し

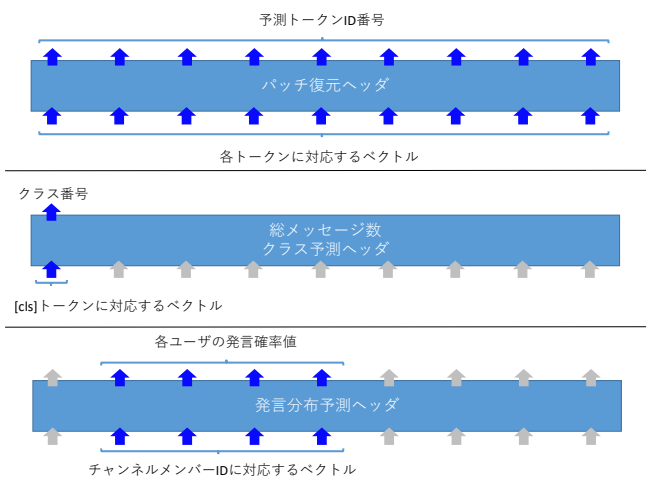


図3 提案モデルにおける3種類のヘッダの入出力。

たトークン列の長さが M になるようにトークン列の後ろに特殊トークン “[pad]” を追加する。

(3) 構成したトークン列のユーザ ID のトークンを一部ランダムに特殊トークン “[mask]” に置き換えマスクする。チャットパッチのメンバー部、メッセージ部はそれぞれ P_α , P_β の確率で置き換える。またこの置き換えたトークンの中から、さらに確率 P_R でランダムに別のトークンに置き換える操作を行う。

(4) マスクしたトークン列のトークンをそれぞれ対応する ID に置き換え、トークン ID 列を構成する。

(5) トークン ID 列をエンベディング部へ通して各 ID を N 次元のベクトルに変換する。

(6) 各ベクトルを Transformer 部とヘッダ部を順に通し、マスクしたトークン列を復元した元のトークン列の予測結果と各メンバーの発言確率と総メッセージ数のクラスを出力する。

エンベディング部の詳細

エンベディング部ではトークン ID を one-hot ベクトルに変換し、それらに線形変換を行い N 次元のベクトルに射影して Token エンベディングを生成する。特に、ユーザ ID のトークンを入力した際の Token エンベディングがユーザエンベディングの結果となる。

さらに学習時のモデルのエンベディング部の構造は BERT と同様に構成する。すなわち学習時には Token エンベディングに Segment エンベディングと Position エンベディングを足し合わせてエンベディングを構成する。Segment エンベディングは、0-ベクトル、1-ベクトルを用い、[sep] トークンを挟むごとに 0 と 1 を反転させる。ただし、ここでは整数 c が成分として並んだベクトルを c -ベクトルと表記している。Position エンベディングはトークン位置 m に対応して m -ベクトルを用いる。

ヘッダ部の詳細

パッチ復元ヘッダおよび発言分布予測ヘッダは、入力側から順にアフィン変換、GELU 活性化、Layer-Normalization、アフィン変換の順で変換する構成とする。総メッセージ数クラス予測ヘッダはアフィン変換のみで変換する構成とする。また、

メッセージ数のクラスは “ 10^1 未満”, “ 10^1 以上 10^2 未満”, “ 10^2 以上 10^3 未満”, “ 10^2 以上 10^4 未満”, “ 10^4 以上” の 5 つで構成する。

4 評価実験

ここでは提案手法でユーザのベクトルを作成し、そのベクトルの評価実験を行う。ここでは次の手法を比較する。

- LSI (Latent Semantic Indexing)
- 提案手法 A: 入力チャットパッチのメンバー部のみ、出力がメッセージ予測のみの提案手法
- 提案手法 B: 入力チャットパッチのメンバー部のみ、出力がパッチ復元・メッセージ予測の提案手法
- 提案手法 C: 入力チャットパッチのメンバー部・メッセージ部、出力がパッチ復元・メッセージ予測の提案手法

4.1 使用するチャットデータ

まず使用するデータについて述べる。ここでは学習データとしてチャンネル数 2726, ユーザ人数 6399, メッセージ数 30 万件のチャット履歴のデータベースを用いる。また学習データと別に、評価データとして 106 件のチャンネルのデータを用いる。使用するチャットデータの特性として、趣味に関するチャットを行うチャンネルが主であり、またチャットシステム上ではユーザが自由にチャンネルの出入りができる。

4.2 LSI の学習方法について

LSI での学習、ベクトル生成方法について述べる。まずユーザが送信したすべてのメッセージに対して、形態素解析して名詞のみを取り出し、その羅列からユーザのドキュメントを作成する。これをすべてのユーザに対して行い、そこから Bag of Words の行列を構成し、さらに TF-IDF を用いて重みづけをする。この行列に対して LSI を実施して $N = 768$ へ次元削減し、最終的に行列の行からユーザのベクトル表現を得る。

4.3 提案手法の学習方法について

モデル学習の初期値は、まずエンベディング部のユーザ ID をエンベディングする部分のパラメータは乱数で初期値を設定し、それ以外のパラメータは日本語データ学習済み BERT モデルのパラメータを初期値とする。この状態で上で述べたチャット履歴のデータベースから 10 万件のチャットパッチをランダムでサンプリングし、学習を行った。この際、各種設定値は $K = 4, M = 512, N = 768, P_\alpha = 0.75, P_\beta = 0.15, P_R = 0.10$ とした。

4.4 知識特性の埋め込み評価

抽出したユーザのベクトル表現に対して知識特性の埋め込み評価を行う。ここでは「あるチャンネルに参加しているユーザはそのチャンネルの議題に対しての知識または興味がある」を仮定する。この仮定からユーザの知識特性がユーザのベクトル表現に適切に埋め込まれている場合、このベクトル表現を用いて、あるチャンネルが与えられたときにそのチャンネルに参加

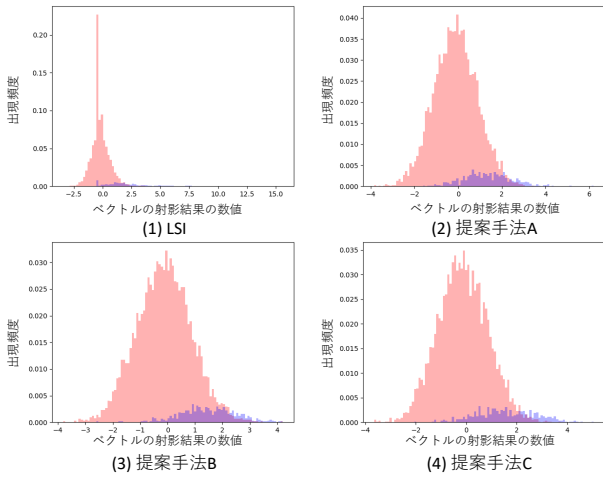


図 4 音楽に関するチャンネルへの参加に対するユーザのベクトル表現の線形判別分析法での射影結果の出現頻度分布. どの手法も理想的な線形射影であっても完全な分離ができておらず重なる部分が存在するため, 本研究では分離度として p_{ij} を定義して比較を行う.

するかどうかを予測することができると考えられる.

そこでユーザがある与えられたチャンネルに参加するかどうかをユーザのベクトルから抽出できるかを確認する. 具体的にはベクトルを理想的な線形写像で射影することでユーザがチャンネルに参加するかどうかの予測ができるかの精度評価を行う. すなわち線形写像における精度上限を測る. ここではフィッシャーの線形判別分析法 [30] を用いてユーザのベクトルを射影し, チャンネルへの参加状況を予測する. あるチャンネルに対して, ユーザがチャンネルに参加していない場合はクラス 1, ユーザがチャンネルに参加している場合はクラス 2 としてラベル付けし, そのラベルとユーザのベクトル表現の対応から線形写像を求め, ユーザのベクトルを 1 次元に射影する. この射影結果のスカラ値から適切にクラス分類ができるかの精度を求める.

例として, 音楽に関するチャンネルへの参加についてのユーザの射影結果 (分類結果) を図 4 に示す. 横軸はユーザのベクトル表現の 1 次元への射影結果の値であり, 縦軸は出現頻度を表す. 図 4 の 2 つのヒストグラムは重なりが存在するため, 理想的な線形写像した後であっても各手法のベクトルは完全にクラスごとに分離できていないことがわかる. 同様に他のチャンネルに対して実験を行った場合でも, 各手法でヒストグラムから明らかな分離を確認できるような結果は得られなかった.

そこで本研究ではその分離度合いを, クラス i のデータが生成された際にクラス j と判別できる確率を用いて測る. クラス k に属するユーザの射影結果のスカラ値 x に対して, これらがある区間 Δx に含まれる度数を $h_k(\Delta x)$ としてヒストグラムを生成する. このとき, 1 次元データの判別確率 p_{ij}^1 は次の計算式で求めることができる.

$$p_{ij}^1 = \sum_{\Delta x} \frac{h_i(\Delta x)}{\sum_{\Delta x} h_i(\Delta x)} \cdot \frac{h_j(\Delta x)}{\sum_k h_k(\Delta x)} \quad (1)$$

ここで総和 $\sum_{\Delta x}$ は, ヒストグラムのすべての Δx がヒストグ

ラムの階級を 1 回ずつとるように計算する. 特にここではチャンネルに参加するユーザが与えられたときに正しく参加すると判別できる確率 p_{22}^1 が重要となる.

各手法に対する p_{ij}^1 の評価データに対する平均値を表 1 にまとめる. 表 1 から, 提案手法が LSI よりも一般的にチャンネルへの参加確率を正しく分類できるような埋め込みを行っていることがわかる. また, 精度に関して提案手法 A, B, C を比べると $A < B < C$ となっている. ここから入力にメッセージ部を入れること, 出力でパッチ復元を行うことが, それぞれ知識特性の埋め込みに寄与していることがわかる.

4.5 交流特性の埋め込み評価

抽出したユーザのベクトル表現に対して交流特性の埋め込み評価を行う. ここではベクトルからのユーザの活発度の予測, モデルを用いたメッセージ数予測精度に関する 2 つの評価実験を行う.

ユーザの活発度の予測

ベクトルからのユーザの活発度の予測の実験について述べる. ユーザのベクトルから次の情報が抽出できるかを確認する.

- 参加チャンネル数: 参加しているチャンネルの個数
- 平均メッセージ回数: メッセージ送信回数の各チャンネルでの平均値
- 接触ユーザ数: 接触したことがあるユーザ数 (ユーザが他人のメッセージの直後に同チャンネルでメッセージを送信した場合に接触したと定義する)

具体的には 4.4 節の知識特性の埋め込み評価の実験と同様に, ベクトルを理想的な線形写像で射影することで, 各情報に関してユーザがクラス分類可能あるかを判定する. すなわち線形写像における精度上限を測る. ここではフィッシャーの線形判別分析法を用いてユーザのベクトル射影し, そこからユーザの活発度に関するクラスを予測する. 各情報に関して 3 つのクラスのラベル付けを行い, そのラベルとユーザのベクトル表現の対応から線形写像を求め, そのラベルとユーザのベクトル表現の対応から線形写像を求め, ユーザのベクトルを 2 次元に射影する. この射影結果のベクトル値から適切にクラス分類ができるかの精度を求める.

ユーザのベクトル表現を参加チャンネル数, 平均メッセージ回数, 接触ユーザ数に対して射影した結果をそれぞれ図 5, 6, 7 に示す. どの図に対してもクラス間の重なりが存在するため, 理想的な線形写像した後であっても各手法のベクトルは完全にクラスごとに分離できていない.

そこで 4.4 節の知識特性の埋め込み評価の実験と同様に, 本研究ではその分離度合いを, クラス i のデータが生成された際にクラス j と判別できる確率を用いて測ることとする. クラス k に属するユーザの射影結果の 2 次元のベクトル値 \mathbf{x} に対して, これらがある面積区間 ΔA に含まれる度数を $h_k(\Delta A)$ としてヒストグラムを生成する. このとき 2 次元データの判別確率 p_{ij}^2 は次の計算式で求めることができる.

表 1 スカラー値 x で表されるクラス i のユーザをクラス j と判断する確率 p_{ij}^1 の評価データ 106 チャンネルに対する平均値. ここではクラス 1 が “チャンネルに参加してしない”, クラス 2 が “参加している” を表す. 特にチャンネルに参加するユーザが与えられたときに正しく参加すると判別できる確率 p_{22}^1 が重要となる. 結果としては提案手法 C が精度が最も高く, 知識特性をより埋め込んでいるといえる.

手法	学習時の入力	学習時の出力	p_{11}^1 平均値	p_{22}^1 平均値	Acc 平均値
ランダム	–	–	0.9830	0.0170	0.9669
LSI	–	–	0.9899	0.5394	0.9791
提案手法 A	メンバー部	メッセージ数予測	0.9914	0.6402	0.9830
提案手法 B	メンバー部	メッセージ数予測, パッチ復元	0.9916	0.6552	0.9837
提案手法 C	メンバー部, メッセージ部	メッセージ数予測, パッチ復元	0.9920	0.6697	0.9844

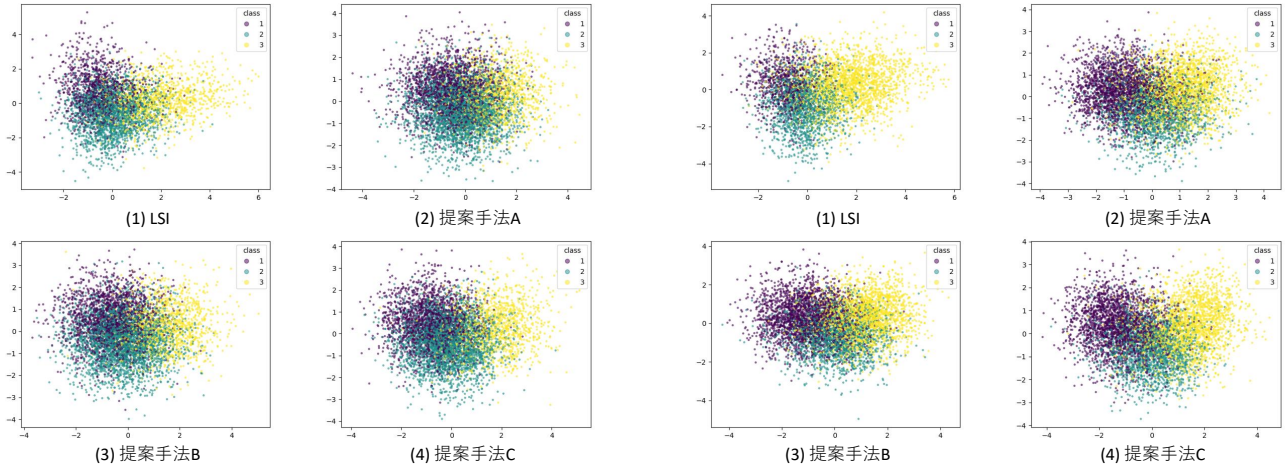


図 5 参加チャンネル数に対するユーザのベクトル表現の線形判別分析法での射影結果. 各クラスは参加チャンネル数が, クラス 1: 0 以上 10 未満, クラス 2: 10 以上 20 未満, クラス 3: 20 以上のユーザを意味する.

図 7 接触ユーザ数に対するユーザのベクトル表現の線形判別分析法での射影結果. 各クラスは接触ユーザ数が, クラス 1: 0 以上 5 未満, クラス 2: 5 以上 20 未満, クラス 3: 20 以上のユーザを意味する.

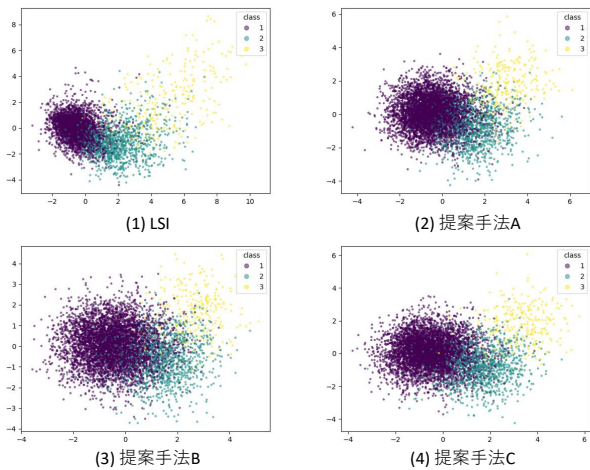


図 6 平均メッセージ回数に対するユーザのベクトル表現の線形判別分析法での射影結果. 各クラスは平均メッセージ回数が, クラス 1: 0 以上 10 未満, クラス 2: 10 以上 100 未満, クラス 3: 100 以上のユーザを意味する.

$$p_{ij}^2 = \sum_{\Delta A} \frac{h_i(\Delta A)}{\sum_{\Delta A} h_i(\Delta A)} \cdot \frac{h_j(\Delta A)}{\sum_k h_k(\Delta A)} \quad (2)$$

ここで総和 $\sum_{\Delta A}$ は, ヒストグラムのすべての ΔA がヒストグ

ラムの階級を 1 回ずつとるように計算する.

各手法に対する p_{ij}^2 の値を表 2 にまとめる. 表 2 から, 提案手法が LSI と同等, もしくはそれ以上に正しくクラス分類できるような埋め込みを行っていることがわかる. 平均メッセージ回数と接触ユーザ数のクラス分類では LSI は精度が高いが, これは LSI があるユーザが使用する語彙量からそのユーザの活発度を測れているからであると考えられる. しかしながら参加チャンネル数といった語彙量だけでは判断しづらいクラス分類では分類精度は低く, 比べて提案手法が精度が高くなっている. また, 精度に関して提案手法 A, B, C を比べると $A < B < C$ となっている. ここから入力にメッセージ部を入れること, 出力でパッチ復元を行うことが, それぞれ交流特性の活発度の埋め込みに寄与していることがわかる.

モデルを用いたメッセージ数予測精度

モデルを用いたメッセージ数予測精度評価の実験について述べる. ここではベクトルにユーザが誰とどこでどれだけ話しやすいかなどの情報をうまく埋め込んでいるかの評価を目的とし, 学習したモデルのメッセージ数予測の精度を評価する. LSI は比較ができないため, 本評価では除外し, 提案手法 A, B, C のみの比較を行う.

表 2 2次元のベクトル値 α で表されるクラス i のユーザをクラス j と判断する確率 p_{ij}^2 の値.

提案手法 C は一般的に精度が高く、交流特性の活発度をより埋め込んでいるといえる.

分類内容	手法	学習時の入力	学習時の出力	p_{11}^2	p_{22}^1	p_{33}^2	Acc
参加チャンネル数	ランダム	–	–	0.3157	0.4834	0.2010	0.3737
	LSI	–	–	0.5971	0.6887	0.7351	0.6691
	提案手法 A	メンバー部	メッセージ数予測	0.6596	0.7304	0.6797	0.6979
	提案手法 B	メンバー部	メッセージ数予測, パッチ復元	0.6828	0.7440	0.7415	0.7242
	提案手法 C	メンバー部, メッセージ部	メッセージ数予測, パッチ復元	0.6885	0.7464	0.7657	0.7320
平均メッセージ回数	ランダム	–	–	0.7736	0.1847	0.0417	0.6343
	LSI	–	–	0.9521	0.7917	0.8866	0.9198
	提案手法 A	メンバー部	メッセージ数予測	0.9278	0.6977	0.8923	0.8838
	提案手法 B	メンバー部	メッセージ数予測, パッチ復元	0.9393	0.7426	0.8984	0.9013
	提案手法 C	メンバー部, メッセージ部	メッセージ数予測, パッチ復元	0.9441	0.7667	0.9425	0.9112
接触ユーザ数	ランダム	–	–	0.3655	0.2926	0.3419	0.3361
	LSI	–	–	0.7977	0.6790	0.8362	0.7762
	提案手法 A	メンバー部	メッセージ数予測	0.78045	0.6624	0.7640	0.7403
	提案手法 B	メンバー部	メッセージ数予測, パッチ復元	0.7751	0.6468	0.7625	0.7332
	提案手法 C	メンバー部, メッセージ部	メッセージ数予測, パッチ復元	0.8137	0.6879	0.8141	0.7770

評価方法は未知のチャンネルのメンバーリストのみから、そのチャンネルの総メッセージ数クラス予測と発言分布予測を行い、その結果を各手法で比較する。総メッセージ数クラス予測はクラス予測精度、発言分布予測は正解データとの KL ダイバージェンスの値を評価指標とする。ここでは評価データの 106 件のうち 25 件のチャンネルに関して予測精度を確認する。これは学習済みモデルを用いてメッセージ数予測の実験を適切に行うためには、評価データのチャンネルに含まれるユーザがすべて学習データに含まれている必要があるためである。

実験結果を表 3 にまとめる。精度に関して提案手法 A, B, C を比べると、ここでは $C < A < B$ となった。 $A < B$ の結果から、パッチ復元のタスクがメッセージ数予測の精度向上に寄与していることがわかる。一方で $C < A < B$ の結果から学習モデルをそのまま使用する場合には手法 C ではなく手法 B を使用したほうが良いことがわかる。これは手法 C の学習時には手法 A, B と入力異なり、メッセージ部が追加されているため、入力異なるこの実験においては不利であったことが示唆される。そのためメッセージ数予測の精度向上には学習データにメッセージ部が存在しないデータを混ぜるか、学習後にファインチューニングを行う必要があると考えられる。

5 まとめ

本研究ではオンライン上のチャット履歴から個人の特性を抽出し、ユーザのベクトル表現を構成することを目的として、個人の知識や興味の特徴と他人との交流傾向をベクトルに埋め込む方法を考えた。本論文ではチャットのデータベースから構成した一部をマスクしたチャットパッチをからパッチ復元とメッセージ数予測を行うニューラルネットワークを構成することで、その中間層の出力を用いてユーザエンベディングする手法を提案した。

実験では、知識特性と交流特性の 2 つの観点から LSI での埋め込み、提案手法を少しずつ変更した提案手法 A, B, C での埋

め込みに対して評価を行った。知識特性の埋め込み評価では、ある与えられたチャンネルに参加するかどうかをユーザのベクトルから線形写像を用いて取り出すことができるかを評価し、ここでは提案手法 C が最も精度が良いことが明らかとなった。また交流特性の埋め込み評価ではユーザの活発度の予測とメッセージ数予測精度の評価を行った。ユーザの活発度の予測ではユーザのベクトルから線形写像を用いてユーザの活発度の情報を取り出せるかを評価し、ここでも提案手法 C が最も精度が良いことが明らかとなった。しかしながらメッセージ数予測精度の評価では提案手法 B が最も精度が良い結果となった。

これらの結果から、提案手法のパッチ復元とメッセージ数予測によって知識特性や交流特性におけるユーザの活発度をベクトルに埋め込めることがわかった。一方で今回の提案モデルをメッセージ数予測にタスクにそのまま使用する場合には、学習時にチャットパッチにおけるメッセージ部のパッチ復元は除外すべきという結果が得られた。これは学習時の入力にメッセージ部が追加されており、評価時と入力異なるためだと考えられる。そのためメッセージ数予測の精度向上には学習後にファインチューニングを行うなど、さらなる検討が必要であることが示唆された。

文 献

- [1] Amy C Edmondson and Zhike Lei. Psychological safety: The history, renaissance, and future of an interpersonal construct. *Annu. Rev. Organ. Psychol. Organ. Behav.*, Vol. 1, No. 1, pp. 23–43, 2014.
- [2] Astrid C Homan, Claudia Buengeler, Robert A Eckhoff, Wendy P van Ginkel, and Sven C Voelpel. The interplay of diversity training and diversity beliefs on team creativity in nationality diverse teams. *Journal of Applied Psychology*, Vol. 100, No. 5, p. 1456, 2015.
- [3] Hannes Leroy, Inga J Hoever, Kathleen Vangronsveld, and Anja Van den Broeck. How team averages in authentic living and perspective-taking personalities relate to team information elaboration and team performance. *Journal of Applied Psychology*, Vol. 106, No. 3, p. 364, 2021.

表 3 評価データ B の 25 チャンネルに対するメッセージ数予測の評価値の平均値. 提案手法 B が最も高く, 本評価では提案手法 C の精度が低い.

手法	学習時の入力	学習時の出力	総メッセージ数クラス予測 Acc	発言分布予測 KL
提案手法 A	メンバー部	メッセージ数予測	0.7368	0.3646
提案手法 B	メンバー部	メッセージ数予測, バッチ復元	0.7368	0.3373
提案手法 C	メンバー部, メッセージ部	メッセージ数予測, バッチ復元	0.5790	0.5297

- [4] Hannes Leroy, Claudia Buengeler, Marlies Veestraeten, Meir Shemla, and Inga J. Hoever. Fostering team creativity through team-focused inclusion: The role of leader harvesting the benefits of diversity and cultivating value-in-diversity beliefs. *Group & Organization Management*, p. 10596011211009683, 2021.
- [5] Shimei Pan and Tao Ding. Social media-based user embedding: A literature review. *arXiv preprint arXiv:1907.00725*, 2019.
- [6] Fatema Hasan, Kevin S Xu, James R Foulds, and Shimei Pan. Learning user embeddings from temporal social media data: A survey. *arXiv preprint arXiv:2105.07996*, 2021.
- [7] Jitao Sang, Dongyuan Lu, and Changsheng Xu. A probabilistic framework for temporal user modeling on microblogs. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management*, pp. 961–970, 2015.
- [8] Shangsong Liang, Xiangliang Zhang, Zhaochun Ren, and Evangelos Kanoulas. Dynamic embeddings for user profiling in twitter. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1764–1773, 2018.
- [9] Shenglin Zhao, Tong Zhao, Irwin King, and Michael R Lyu. Geo-teaser: Geo-temporal sequential embedding rank for point-of-interest recommendation. In *Proceedings of the 26th international conference on world wide web companion*, pp. 153–162, 2017.
- [10] Yun Xiong, Yao Zhang, Hanjie Fu, Wei Wang, Yangyong Zhu, and S Yu Philip. Dyngraphgan: Dynamic graph embedding via generative adversarial networks. In *International Conference on Database Systems for Advanced Applications*, pp. 536–552. Springer, 2019.
- [11] Maoguo Gong, Shunfei Ji, Yu Xie, Yuan Gao, and AK Qin. Exploring temporal information for dynamic network embedding. *IEEE Transactions on Knowledge and Data Engineering*, 2020.
- [12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- [13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.
- [14] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, Vol. 32, , 2019.
- [15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.
- [16] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.
- [17] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.
- [18] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.
- [19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- [20] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. *arXiv preprint arXiv:2103.15691*, 2021.
- [21] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [22] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 701–710, 2014.
- [23] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855–864, 2016.
- [24] Daixin Wang, Peng Cui, and Wenwu Zhu. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1225–1234, 2016.
- [25] Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J Kim. Graph transformer networks. *Advances in Neural Information Processing Systems*, Vol. 32, pp. 11983–11993, 2019.
- [26] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, Vol. 41, No. 6, pp. 391–407, 1990.
- [27] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, Vol. 3, No. Jan, pp. 993–1022, 2003.
- [28] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International conference on machine learning*, pp. 1188–1196. PMLR, 2014.
- [29] Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics.
- [30] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, Vol. 4. Springer, 2006.