統計データ検索のための FT-Transformer によるランキング学習

岡本　卓† 宮森　恒†

† 京都産業大学大学院 先端情報学研究科　〒 603–8555 京都府京都市北区上賀茂本山
E-mail: †{i2086042,miya}@cc.kyoto-su.ac.jp

あらまし　本稿では，FT-Transformer によるランキング学習を用いた，統計データに対するアドホック検索の問題に取り組む．近年，様々な団体が保有している公共的データを，オープンデータとして有効活用するための基盤整備が進んでおり，オープンデータの一種である統計データに対するアドホック検索基盤の重要性も高まっている．統計データは一般に表形式で記載されており，表の構造や大きさは多種多様なものが存在しているが，近年それら統計データに対するアドホック検索の性能評価を可能とするデータセットが整備されてきた．本稿では，このデータセットを活用し，表の構造や大きさが多様な統計データに対して，Transformer アーキテクチャに基づくランキング学習の性能を分析する．実験では，BM25 をはじめ，MLP や ResNet ベースの従来手法との比較を行い，課題を明らかにする．

キーワード　統計データ，アドホック検索，Transformer，オープンデータ，深層学習，情報検索

1 はじめに

近年，様々な団体が保有している公共的なデータを，オープンデータとして有効活用するための基盤整備が進んでおり，オープンデータの一種である統計データに対するアドホック検索基盤の重要性が高まっている．例えば，統計データは，社会問題になっているフェイクニュースに対処するための事実確認 (ファクトチェック) において重要な役割を果たすと考えられている．一般に，政府や業界団体等が公開している統計データは一定の品質が担保されていると考えられるため，真偽が不確かな情報とそれら統計データを比較照合することで，情報の整合性の有無などを確認できるためである．

統計データを用いた事実確認 (ファクトチェック) 支援を実現するためには，事実確認したいテキストを入力として，関連する統計データを取得する段階 (第 1 段階) と，取得した統計データの中から回答となる箇所を特定し出力する段階 (第 2 段階) の 2 つのステップが必要となる．本稿では，このうち最初の段階である，統計データの文書集合からクエリに関連する文書を取得するアドホック検索技術を研究の対象とする．

従来研究では，文書中の表検索や表理解に関する研究が活発に行われてきたものの，統計データそのものを対象としたアドホック検索についてはこれまでほとんど扱われてこなかった．統計データに対するアドホック検索では，統計データ本体とそのメタデータの組を一文書として扱い，本稿ではこれを統計文書と呼ぶこととする．統計データはタイトルなどを除くと基本的に数値の並びで構成されており，表のサイズや構造の複雑さが多様であるという特徴がある．また，メタデータは表のタイトルや簡単な説明が記述されており，文書長が短いという特徴がある．

統計文書に対するランキング手法として，BM25 を採用することが考えられる．BM25 は，テキスト文書に対する最も成功したランキング手法の一つであり，統計文書のテキスト部分を手がかりにランキングすればある程度の性能を達成することは期待できる．しかし，BM25 のみでは，統計データのもつ構造や数値の並びなどがもつ特徴を十分に活用できているとは言い難く，それらも適切に活用したランキング手法を考案することが更なるランキング性能の向上のためには必要である．

主要国の食料自給率（カロリーベース食料自給率）（単位：パーセント）

国名	1965	1970	1975	1980	1985	1990	1995	2000	2005	2010	2015
アメリカ	117	112	146	151	142	129	129	125	123	135	129
カナダ	152	109	143	156	176	187	163	161	173	225	255
ドイツ	66	68	73	76	85	93	88	96	85	93	93
スペイン	96	93	98	102	95	96	73	96	73	92	83
フランス	109	104	117	131	135	142	131	132	129	130	132
イタリア	88	79	83	80	77	72	77	73	70	62	62
オランダ	69	65	72	72	73	78	72	70	62	68	64
スウェーデン	90	81	99	94	98	113	79	89	81	72	77
イギリス	45	46	48	65	72	75	76	74	69	69	71
スイス	－	－	－	－	－	－	－	59	57	52	51
オーストラリア	199	206	230	212	242	233	261	280	245	182	214
韓国	－	80	－	70	－	63	51	51	45	47	43
日本	73	60	54	53	53	48	43	40	40	39	39

図 1　従来研究で使用される統計データの例

人口　１２６，２２７千人（令和２年１０月１日現在）
純旅客　　３千人（推計値）

類別・品目別	国内 生産量	外国貿易 輸入量	外国貿易 輸出量	在庫の増減量	国内消費仕向量
1．穀類	9,360	23,898	110	350	32,054
a．米	8,145	814	110	248	7,857
	(a) (381)				744
	(b) (33)				
b．小麦	949	5,521	0	58	6,412
c．大麦	201	1,649	0	40	1,810
d．裸麦	20	25	0	9	36
e．とうもろこし	0	15,366	0	△ 44	15,410
f．こうりゃん	0	379	0	33	346
g．その他の雑穀	45	144	0	6	183

図 2　本研究で使用される統計データの例

本稿では，統計文書に対するアドホック検索の問題に取り組み，表の構造や大きさが多様な統計データを含むデータセットを用いて，Transformer アーキテクチャに基づくランキング手法の性能を分析する．本稿で用いるデータセットは，NTCIR-15 DataSearch タスク [1] で提供されたものであり，日本の政府統

計ポータルサイト (e-Stat) で収集された多種多様な統計文書で構成されている．従来の表検索や表理解の研究では，図 1 に示すように構造が単純で機械的な解釈が容易な表形式のデータが主に用いられていた．一方，本稿では，図 2 に示すようにヘッダや中身の構造や対応関係が単純でなく，機械的な解釈が難しいものが多く含まれたデータセットを用いる．

本稿では，Transformer アーキテクチャを表形式データ用に修正した，FT-Transformer [2] を用いる手法を提案する．その際，ユーザが意図する内容を適切に捉えるため，統計文書をカテゴリで絞り込むカテゴリ検索を適用する．また，検索の手がかりを増やすために，統計データのヘッダ部分を抽出するデータ補強を用いる．

統計文書に対するアドホック検索の先行研究においては，統計文書のメタデータのみを用いたランキング手法 [3] [4] [5] が多く，統計データ本体の内容を活用した事例は少ない．しかし，メタデータのみを使用したランキング手法では，従来のテキスト文書を対象としたアドホック検索でのランキングと比べて同等以上の性能を示すには至っていない．本稿では，統計データ本体がもつ手がかりを FT-Transformer の枠組みで活用することで，ランキング性能にどのような影響があるのかを明らかにする．

実験では，NTCIR-15 DataSearch タスク [1] で提供されたデータセットを用いた各検索結果とクエリとの関連性を 3 段階で評価し nDCG@10 で比較することで，提案手法の有効性を検証する．

本稿の貢献は以下の通りである．

（1） 統計文書のアドホック検索の問題において，統計データ本体の情報を活用したランキング手法を提案した．

（2） 構造が単純な表のランキングにおいて有効性が報告されている FT-Transformer を，統計文書のアドホック検索に適用した．

（3） 表のサイズや構造の複雑さが多様である統計データで構成される統計文書データセットを用いてランキング性能を検証した．

本論文の構成は，以下の通りである．2 節で関連研究について述べ，3 節で本稿で使用する統計文書データセットと既存研究の文書との違いについて述べる．4 節で本稿で扱う問題を定式化し，提案手法について詳述する．5 節で実験内容と実験結果，考察を示し，6 節で結論と課題をまとめる．

2 関 連 研 究

本節では，統計文書に対するアドホック検索の関連研究として，2.1 節でテキスト文書を対象としたアドホック検索の研究，2.2 節で統計データを対象とした検索に関する研究，2.3 節で統計データを扱ったデータセットに関する研究を紹介する．

2.1 テキスト文書に対するアドホック検索

情報検索は古くから研究されてきた分野であり，クエリに関連する文書を検索するため，各文書に様々なスコアを付与する手法がいくつも提案されてきた．主な手法としては，クエリと文書の関連する確率を計算する確率モデル [6] やニューラルネットを使用した推論ネットワークモデル [7] などが挙げられる．特に確率モデルの一つである BM25 [8] は，現在でも有用性が高く，様々な検索エンジンにおいて広く利用されている．近年では，推論ネットワークモデルを発展させた深層学習を利用した手法 [9] や自然言語理解の分野で目覚ましい進展をもたらしている BERT [10] を使用したランキング手法 [11] も提案されている．これらの研究が対象とする文書はテキストベースで作成されている．一方，本研究が対象としている統計文書は，文書長の短いテキストベースのメタデータと，ほとんどが数値で記述された表形式の統計データから構成されており，従来のテキスト文書とはその特徴が大きく異なる．

2.2 表データを対象とした研究

表を対象とした検索についてもこれまで多くの研究がなされている．Zhang らは，WikiTables コーパスに基づく表検索のためのデータセットを作成し，クエリと表を複数の意味空間で表現し，それらを様々な類似性尺度でマッチングする手法を提案している [12]．Shraga らは，表の内的類似尺度と表の外的尺度を用いてリランキングする手法を提案している [13]．Chen らは，表から選択された項目とクエリ，関連フィールドを BERT を用いて学習する手法を提案している [14]．これらの研究はどれも WikiTable データセットを用いているが，このデータセットは，Wikipedia 文書の表を用いているため，表のサイズが比較的小さく，表本体のセル値が数値でない通常の単語である割合が比較的大きいという特徴がある．一方，本研究では，e-Stat から収集された統計文書データセットを用いており，表のサイズが比較的大きく，表本体のセル値が数値である割合が大きいという特徴がある点でこれらの従来研究と異なる．

また，表を対象とした表現学習に関する研究も多くなされている．Yin らは，自然言語文と半構造化テーブルの表現を合同で学習する事前学習済み言語モデルを提案しており [15]，Somepalli らは，表形式データの問題を解くため，行と列の両方に対する注意機構を用いて強化された埋め込み手法や，ラベルが少ない場合に使用する，新しい対照的な自己教師付き事前学習法を提案している [16]．Xiang らは，事前学習時に，教師無しで関係表の行と列構造をモデル化するために、構造を考慮した Transformer エンコーダ手法を提案している [17]．これらの研究はどれも 1 つの表を対象にしており，計算された特徴量も 1 つの表でのみしか使用されない．一方，本研究では，統計文書を対象としており，複数の表で特徴量を計算し，比較する必要があるなどの点でこれらの従来研究とは異なる．

2.3 統計データを扱ったデータセットの先行研究

近年になり，統計データを扱ったデータセットの研究が増えてきている．Zhu らは，答えを推測するために，加算，減算，乗算，除算，カウント，比較/ソート，およびそれらの合成などの数値的推論が必要となるデータセットとして，実際の財務報告書からサンプルを抽出し、TAT-QA と名付けた表形式とテ

表 1　統計データのファイル形式の分布

file format	frequency
xls	686436
csv	568042
pdf	49124
xlsx	34794
xlsm	6

キストデータの両方を含む新しい大規模 QA データセットを構築した [18]．Wenhu らは，異種情報の再利用を必要とする新しい大規模な質問応答データセット HybridQA を構築した．各質問は Wikipedia の表と、その表中のハイパーリンクでリンクした複数の自由形式のテキストとで構成した [19]．また，作成した HybridQA から表データと非構造化テキストにまたがるマルチホップ推論を必要とする質問応答を抽出したデータセット OTT-QA も構築した [20]．これらの研究はどれも，異なるデータセットを使用しているが，どのデータセットもヘッダと中身の表が対応しており，機械的な解釈が容易であるという特徴がある．一方，本研究で扱うデータセットは，ヘッダと中身の表が必ずも一致しているとは限らず，機械的な解釈も困難であるという特徴がある点がこれらの従来研究とは異なる．

3　データセット

本稿では，被検索文書集合として，NTCIR-15 Data Search 日本語タスクで提供されたデータセットを使用する (以後，統計文書データセットと呼ぶ)．各文書は，政府統計ポータルサイト (e-Stat) から収集されている．統計データは，xls や csv，pdf などの形式で保存された統計データ本体のファイルに相当する．統計データは，1 つ以 上の表形式のデータを含み，表のタイトルや表のヘッダには数値以外の通常のテキストが記載されているが，表本体は，ほとんどの場合，非常に多くの数値で構成されている．表 1 に，統計データのファイル形式の分布を示す．

メタデータは，e-Stat の統計データの導入ページに記載されているデータを抽出したものである．メタデータは JSON 形式のファイルであり，統計データの id，統計データの導入ページの URL，統計データの title のほか，統計データの簡潔な概要を記述した description，統計データ本体の URL，ファイル形式，ファイル名などを表す変数名と，それぞれに対応する値で構成されている．

表 2 に，統計データにおける数値と数値以外の単語の使用割合の平均と標準偏差，および，統計データの総数を示す．ここで，日本語の単語の使用割合は，統計データに含まれる全文字列を MeCab を用いて単語に分割し，1 単語を構成する文字が全て半角数字である場合に数値を表す単語，それ以外の場合に数値以外の単語と判定した．表 2 より，数値を表す語が日本語統計データでは，平均約 0.79 の割合で使用されていることがわかる．

同様に，表 3 に，メタデータにおける数値と数値以外の単語の使用割合の平均と標準偏差，および，メタデータの 総数を示す．メタデータでは，数値以外の語が平均約 0.9 の割合で使用されており，統計データに比べて，検索の手がかりとしやすい，数値以外の単語の割合が大きいことがわかる．

表 2　統計データにおける数値と数値以外の単語の使用割合

言語	数値		数値以外の文字		統計データの総数
	平均	標準偏差	平均	標準偏差	
日本語	0.786	0.221	0.213	0.221	1,338,402

表 3　メタデータにおける数値と数値以外の単語の使用割合

言語	数値		数値以外の文字		メタデータの総数
	平均	標準偏差	平均	標準偏差	
日本語	0.076	0.023	0.923	0.023	1,338,402

表 4　メタデータで使用される単語長

言語	title		description		title and description	
	平均	標準偏差	平均	標準偏差	平均	標準偏差
日本語	36.2	16.7	20.0	8.2	56.2	21.2

次に，メタデータの文書長について説明する．メタデータにおいては，title，description 変数以外の変数に対応する値は，単なる文書 id や 収集元の URL など，必ずしも検索の手がかりとしやすい内容とはなっていない．そこで本稿では，メタデータの title，description 変数の値で構成される組を，メタデータの文書とみなすこととする．表 4 にメタデータの title，description 変数のそれぞれの値に含まれる単語数の平均と標準偏差，両変数の値に含まれる単語数の平均と標準偏差を示す．ここで，単語数は，当該文字列を MeCab で分割した結果の単語数を表す．

アドホック検索では，従来，Web 文書，学術論文，議論フォーラム，政府や企業の内部文書，ニュースやソーシャルメディアの記事など，様々な文書が被検索文書として用いられてきたが，これらの文書は，主に日常的に用いられる文章などの自然言語で記述されている点が特徴である．例えば，小説などの比較的長い文書長をもつ文書では約 3 万単語 [21] が用いられ，比較的短い文書長の新聞記事では約 330 単語 [22] が用いられている．

一方，本稿で対象とするメタデータで検索の手がかりとしやすい自然言語で記述された変数 title，description の単語数の平均は 56 単語程度と非常に少ない．そのため，メタデータに数値以外の割合が多くても単語数が少ないため，従来のアドホック検索手法をそのまま適用するだけでは，クエリを適切に満たす検索結果を得るのは難しい．

4　提 案 手 法

4.1　問 題 定 義

本節では，本稿で扱う問題を定式化する．まず，クエリ集合 Q，被検索文書集合 D をそれぞれ

$$Q = \{q_i\}, \quad D = \{d_j\} \tag{1}$$

とする．

ここで，クエリ q_i は 1 回の検索で与えられる 1 つ以上の単語列 $w_1^{q_i}, w_2^{q_i}, \ldots, w_{n_{q_i}}^{q_i}$ を表し，1 つの被検索文書 d_j は，1 つ

のメタデータ m_j，および，1 つの統計データ t_j の組として表される．

$$d_j = (m_j, t_j) \tag{2}$$

被検索文書集合 D のうち，クエリ q_i に関連がある文書集合は

$$D^{q_i,+} = \{d_j^{q_i,+}\} \tag{3}$$

と表すこととする．

また，クエリ q_i に関連がある文書集合 $D^{q_i,\ +}$ を，ランキング関数 $rank(q_i, d_j^{q_i,+})$ で降順にソートしたランキングリストを $R_{rank(q_i,d_j^{q_i,+})}$ で表すこととする．

本稿で取り組む問題の目的は，統計文書集合 D から，クエリ q_i に関連がある文書集合の適切なランキング結果 $result_{q_i}$ を取得することである．すなわち，

$$result_{q_i} = R_{rank(q_i,d_j^{q_i,\ +})} \tag{4}$$

とすることである．

4.2 データ補強

メタデータの文書長の短さを補うため，統計データ自身から表のヘッダ情報を抽出し，被検索文書に追加して扱う手法を提案する．ヘッダ情報を抽出し，メタデータを補う手法は先行研究 [23] で有用性がある程度示されている．具体的には，統計データ内の各行あるいは各列における空でないセル数を，行あるいは列ごとに順に調べ，空でないセル数の変化によって列あるいは行ヘッダをそれぞれ抽出する．列ヘッダを抽出する場合の抽出手順をアルゴリズム 1 に示す．まず，入力された統計データを sd とし，直前の行の空でないセル数を格納する変数 $prev$ を 0 で，列ヘッダを格納する変数 hdr_col を空のリストでそれぞれ初期化する．統計データの各行の並びをリストとして格納した $sd.rows$ の要素数を max_row に格納する．

Algorithm 1 Extracting column headers from statistical data

Input: statistical data sd
Output: column headers hdr_col

```
prev = 0
hdr_col = []
max_row = sd.rows.length
for i = 1, ..., max_row do
    curr = sd.rows[i].unempty_cells().length
    if curr > prev then
        hdr_col.append(sd.rows[i].unempty_cells())
    end if
    prev = curr
end for
return hdr_col
```

次に，1 行目から max_row 行まで以下を繰り返す．セルの並びが格納されているリストから，空でないセルのみをフィルタリングして抽出し，その結果をリストとして返すメソッドを $unempty\_cells()$ とする．第 i 行目に含まれるセルのリスト $sd.rows[i]$ に対して，$unempty\_cells()$ を適用し，第 i 行目に含まれる空でないセルのリストを取得する．第 i 行目の空でないセルのリストの要素数 $length$ を，第 i 行目の空でないセル数として $curr$ に格納する．第 $i-1$ 行目の空でないセル数は $prev$ に格納されている．

$curr$ が $prev$ よりも大きければ，リストの最後に別のリストを追加する $append()$ メソッドを用いて，列ヘッダ hdr_col の最後に，第 i 行目の空でないセルのリストを追加する．$prev$ を $curr$ で更新し，次の行の処理に移行する．

繰り返しが終了したら，列ヘッダ hdr_col を h_j^{col} に返す．h_j^{col} には，列ヘッダに該当する，空でないセルを格納した行のリストが格納されている．行ヘッダについては，アルゴリズム 1 の行と列を相互に入れ替えた内容の処理を実行し，ヘッダ h_j^{row} を抽出する．抽出したヘッダ情報をメタデータ m_j に追加することで，メタデータの文書長の短さを補った文書 d_j^{m+h} を作成する．

4.3 カテゴリ検索

ユーザのクエリが意図する検索範囲を適切に反映させるため，被検索文書集合をカテゴリで絞り込む手法を提案する．インデキシング時には，テキスト分類器を用いて各被検索文書にカテゴリを付与し，カテゴリ付きの新たな被検索文書集合として登録する．検索時には，クエリからテキスト分類器を用いてカテゴリを推定し，推定されたカテゴリに属する被検索文書集合に対してのみランキングを実行し，検索結果を返す．カテゴリ集合は，以下の手順で定める．

まず，コミュニティ質問応答 Web サービス Yahoo! 知恵袋のサイト内検索で，“e-Stat” を意味するクエリにより得られる検索結果全て収集する．クエリは，質問と回答のいずれか，あるいは，その両方に含まれる可能性が考えられるため，収集した質問・回答アイテムから，回答に e-Stat へのリンクが含まれているものを抽出し，その各々に対応する質問が属するカテゴリを列挙した．以上により，Yahoo!知恵袋で用いられている 10 のカテゴリからなるカテゴリ集合を定めた．

次に，カテゴリを推定するテキスト分類器を構築する．収集した質問・回答アイテム集合の各アイテムについて，名詞と動詞の品詞をもつ単語を抽出し，該当する単語の fastText による分散表現の平均を各アイテムの特徴ベクトルとした．この特徴ベクトルと正解カテゴリを用いて SVM で学習することでテキスト分類器を構築した．

4.4 統計データ本体を活用した特徴

統計データ本体がもつ手がかりを有効活用するために，統計データ本体の特徴を抽出し，事前学習言語モデルを用いてランキングする手法を提案する．

具体的に表本体の形式的 (formal) 特徴として表 5 に示す 4 種類を用いることとした．これらは，先行研究 [24] で用いられたものである．#Rows, #Cells, #EmptyCells は，各表の本体についての該当数の，1 件の統計データあたりの平均値を表

表 5　表本体のみから得られる形式的特徴 (formal)

表の特徴項目	表の特徴の説明
#Rows	統計データにおける各表本体の行数の平均
#Cols	統計データにおける各表本体の列数の平均
#EmptyCells	統計データにおける各表本体に出現する空セル数の平均
#InLinks	表本体内の各語を RDF の目的語とみなした際の，DBpedia Japanese における該当目的語に対する述語の総数の平均

す．#InLinks は，各表本体内の各語 (主に数値) を RDF の目的語とみなした際の，DBpediaJapanese の RDF における該当目的語に対する述語の総数の，1 件の統計データあたりの平均値を表す．DBpedia Japanese は，日本語 Wikipedia をもとに RDF 形式の LOD(Linked Open Data) を整備したリソースである．

また，表本体の内容 (content) 的特徴として表 6 の 10 種類の特徴を用いる．medRows は，表本体の各行の中央値を集めて得られる値の集合から表ごとの中央値を計算し，その計算結果の値の全ての表についての中央値をとったものである．統計データ t_j における表 $\tau_{j,k}$ の l 行目の各セルの値の集合を $\rho_{j,k,l}$ とし，行ごとの中央値を $v_j^{med,row}$ とおくと，

$$v_j^{med,row} = med^k(med^l(\rho_{j,k,l})) \tag{5}$$

である．ここで，$med^a(x_a)$ は，x_a の添字 a について中央値をとる関数を表す．medCols は，medRows を求める際の行ごとの操作を列ごとの操作に置き換えて得られる値である．normMedRows は，medRows を求める際の $\rho_{j,k,l}$ の各セルの値を，$\rho_{j,k,l}$ 行の合計を 1 としたときの割合に置き換えて得られる値である．normMedCols は，normMedRows を求める際の行ごとの操作を列ごとの操作に置き換えて得られる値である．

dispSeqRows については，表本体の各行のセルの並びを系列と捉え，その変化を表す特徴として計算する．その値は，表本体の各行の各セルについて，窓幅 Δm で差分をとることで得られる値の集合の中央値を計算し，それらを集めて得られる値の集合から表ごとの中央値を計算し，その計算結果の値の全ての表についての中央値をとることで求められる．統計データ t_j における表 $\tau_{j,k}$ の l 行目 m 列目のセルの値を $\chi_{j,k,l,m}$ とし，dispSeqRows の値を $v_j^{seq,row}$ とおくと，

$$v_j^{seq,row} = med^k(med^l(med^m(\{\chi_{j,k,l,m+\Delta m} - \chi_{j,k,l,m}\}))) \tag{6}$$

である．全統計データについての表の行数の平均が 217 であったこととある程度の値の変化を表現できるようにするため，窓幅の最大を 50 と設定し，窓幅は $\Delta m = 1, 25, 50$ として 3 種類の値を求めることとした．

dispSeqCols については，dispSeqRows を求める際の行ごとの操作を列ごとの操作に置き換えて得られる値である．全統計データについての表の列数の平均が 58 であったことから，窓幅の最大を 12 と設定し，窓幅は $\Delta m = 1, 6, 12$ として 3 種類の値を求めることとした．

表 6　表本体のみから得られる内容的特徴 (content)

表の特徴項目	表の特徴の説明
medRows	表本体の各行の中央値を集めて得られる値の集合から表ごとの中央値を計算し，その計算結果の値の全ての表についての中央値をとったもの
medCols	行ごとの中央値の行ごとの操作を列ごとの操作に置き換えて得られる値
normMedRows	行ごとの中央値を求める際の $\{\rho_{j,k,l}\}$ の各セルの値を，$\rho_{j,k,l}$ 行の合計 を 1 としたときの割合
normMedCols	行ごとの割合の行ごとの操作を列ごとの操作に置き換えたもの
dispSeqRows	表本体の各行のセルの並びを系列と捉え，その変化を表す特徴として計算した値 (窓幅 3 種類)
dispSeqCols	行ごとの系列変化の行ごとの操作を列ごとの操作に置き換えたもの (窓幅 3 種類)

表 7　表全体から得られる特徴 (all)

表の特徴項目	表の特徴の説明
hitsLC	統計データ本体の左端の列における総クエリ語頻度
hitsSLC	統計データ本体の左端の 2 列目における総クエリ語頻度
hitsB	統計データ本体に出現するクエリの出現頻度
qInPgTitle	全統計文書に対するメタデータの title で見つかったクエリの比率
LMIR.DIR	ディリクレ平滑化を用いた言語モデルで推定されるクエリ尤度
LMIR.JM	線形補間法を用いた言語モデルで推定されるクエリ尤度
LMIR.ABS	Absolute Discount 平滑化を用いた言語モデルで推定されるクエリ尤度
BM25	クエリと文書の BM25 のスコア
TF-IDF	クエリと文書の TF-IDF のスコア

最後に，表全体から得られる特徴 (all) として，表 7 に示す 9 種類を用いることとした．hitsLC，hitsSLC は，それぞれ表全体の左端と左端の 2 列におけるクエリトークンの出現頻度であり，hitsB は，表全体におけるクエリトークンの出現頻度を表す．qInPgTitle は，メタデータの title の全トークン数に対する，title 中に含まれるクエリトークン数の割合を表す．いずれの値も 1 件の統計データに含まれる 1 つ以上の表に対する平均値を表す．LMIR.DIR，LMIR.JM，LMIR.ABS は，いずれも言語モデルで推定されるクエリ尤度の値を表し，各言語モデルを構築する際に用いる平滑化手法が，それぞれディリクレ平滑化，線形補間法，Absolute Discount 平滑化である点が異なっている．BM25，TF-IDF は，それぞれクエリと文書の BM25，TF-IDF のスコアを表す．

4.5　FT-Transformer による再ランキング手法

表本体がもつ手がかりを有効活用するために，Transformer を表形式データ用に修正した FT-Transformer [25] で得られる特徴から予測されるスコアによる再ランキング手法を試みる．FT-Transformer は，先行研究において表形式のデータセット 11 種類を用いた複数タスクのほとんどで最良の結果を示して

いる．本稿では，多様な大きさや構造をもつ統計文書に対して FT-Transformer がどの程度良好なランキング性能を示すか検証する．

FT-Transformer を用いる再ランキング手法の概要を図 3-図 5 に示す．図 3 に示すように，まず，クエリ，メタデータ，および，統計データの表全体を用いた特徴が特徴トークナイザにより埋め込み T に変換される．次に，[CLS] トークンが付与された埋め込み T_0 が L 層の Transformer により文脈ベクトル T_L に変換される．最後に，[CLS] トークンに該当する最終的な表現を用いて関連度スコアの予測が出力される．

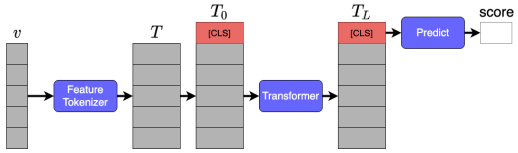

図 3　FT-Transformer のアーキテクチャ

まず，入力には，クエリ，メタデータ，および，統計データの表全体を用いる．ここでは，表 5-表 7 の 23 次元の特徴ベクトルを入力として用いる (実験では比較のため，表 6 を除く 13 次元の特徴ベクトルも用いる)．

入力する p 次元特徴ベクトルを $v_b^{q_i,m_j,t_j} \in \mathbb{R}^p$，埋め込み $T \in \mathbb{R}^{p\times d}$ とし，それぞれの第 p 行成分を $v_{b,p}^{q_i,m_j,t_j} \in \mathbb{R}$, $T_p \in \mathbb{R}^d$ で表すと，

$$T_p = b_p + v_{b,p}^{q_i,m_j,t_j} \cdot W_p \in \mathbb{R}^d \tag{7}$$

である．ここで，b_p, W_p は，それぞれ p 番目の特徴バイアス，重みを表す．埋め込み T は，図 4 に示すように，T_p を縦方向に並べることで得られる．

$$T = vstack[T_1, \ldots, T_p] \tag{8}$$

ここで，$vstack[]$ は引数の要素を縦方向に並べる操作を表す．

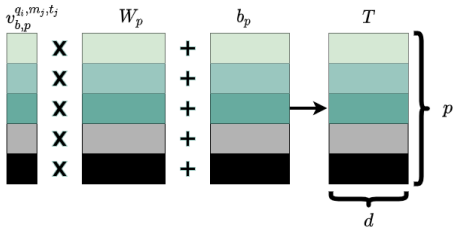

図 4　埋め込み T の計算

次に，埋め込み T に [CLS] トークンを追加した T_0 を作成し，L 層からなる Transformer 層 $F_1, \ldots, F_L$ を適用する．図 5 に，1 層分の Transformer の構成を示す．

$$T_\lambda = F_\lambda(T_{\lambda-1}) \tag{9}$$

最後に，[CLS] トークンに該当する最終的な表現 $T_L^{[CLS]}$ を用いて関連度スコア $score_{FTT}^{q_i,m_j,t_j}$ を得る．

$$score_{FTT}^{q_i,m_j,t_j} = Linear(Relu(LayerNorm(T_L^{[CLS]}))) \tag{10}$$

初期ランキング結果は，関連度スコア $score_{FTT}^{q_i,m_j,t_j}$ を用いて再ランキングされる．

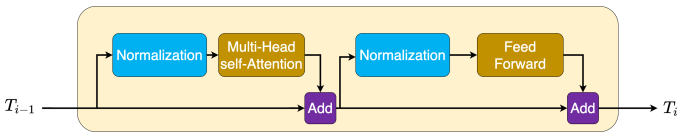

図 5　単一の Transformer 層

5　実　　験

本節では，提案手法の様々な組み合わせで，統計文書データセットに対するランク付けされたリストを生成し，従来のアドホック検索と同様の方法で各手法の有用性を評価する．

5.1　実 験 方 法

評価に用いるクエリとして，NTCIR-15 Data Search タスクで提供されたテストクエリ 96 件を使用する [1]．実験での評価方法について説明する．評価方法は NTCIR-15 Data Search タスクで採用された評価方法に準拠する．はじめに，各テストクエリから得られた検索結果のうち上位 10 件の文書を取得する．評価者には，上位 10 件のうちの 1 文書と，クエリの作成元となった質問文のペアについて，L0,L1,L2 の 3 段階で関連性を評価する作業を指示する (L0: 関連がない， L1: 部分的に関連がある，L2: 関連がある)．作業データとして既知データを加えることで，質の悪い評価者を除外し，品質を保つようにしている．関連度の評価には，クラウドソーシングを利用した．評価した関連度から nDCG@10 によりスコア算出し，手法間の性能を比較する．

5.2　評価対象とする手法

表 8 に，実験で評価対象とする手法の一覧を示す．用いる特徴としては，表本体の形式的特徴 $formal$ と表全体から得られる特徴 all の組み合わせた手法 (以下，F_{base}+FTT で表す)，および，表本体から得られる内容的特徴 (content) のみを用いる手法 (以下，F_{nc}+FTT で表す)，これらを組み合わせる手法 (以下，$F_{base+nc}$+FTT で表す)，F_{base} に BERT を用いた特徴を追加する手法 (以下，$F_{base+bert}$+FTT で表す) をそれぞれ採用する．また，先行研究 [23] から提案手法全てにデータ補強を適用しており，比較手法には，データ補強によるランキング手法 (DA と表す) と比較する．

ただし，それぞれの統計文書全てに対して表の特徴量や FT-Transformer を計算していると時間がかかりすぎるため，カテゴリ検索で検索された文書集合上位 100 件の文書に対してのみ再ランキングを実行するものとする．

5.3　実 験 結 果

表 9 に，ランキング結果を示す．F_{base}+FTT による nDCG@10 の値は 0.254 となった．また，F_{nc}+FTT による

表 8　評価対象とする手法の一覧
Table 8　Combination of evaluation methods

手法名	formal + all	content	bert	FT-Transformer
DA				
F_{base}+FTT	✓			✓
F_{nc}+FTT		✓		✓
$F_{base+nc}$+FTT	✓	✓		✓
$F_{base+bert}$+FTT	✓		✓	✓

表 9　ランキングの評価結果

手法名	nDCG@10
DA	0.394
F_{base}+FTT	0.254
F_{nc}+FTT	0.245
$F_{base+nc}$+FTT	0.253
$F_{base+bert}$+FTT	0.250

結果は 0.245 となり，F_{base}+FTT と比べて 0.009 減少した．これは，表本体の数値やその変化で構成される内容的特徴 content は，形式的な特徴 formal と表全体から得られる特徴 all を合わせた F_{base} よりも，統計文書のランキング性能が低いことを示している．

$F_{base+nc}$+FTT による nDCG@10 の値は，F_{base}+FTT による値に比べ，0.001 減少した．これも，表本体の数値やその変化で構成される内容的特徴 content が統計文書のランキングを向上させるよりはむしろ阻害していることを示している．

最後に，F_{base}+FTT に BERT から得られる特徴 $f_{bert}^{q_i,t_j}$ を追加した $F_{base+bert}$+FTT による nDCG@10 の値は，F_{base}+FTT による値から 0.004 減少した．F_{base} に BERT から得られる特徴を追加した場合の方が，F_{nc} を追加した場合よりも減少幅が大きく，BERT から得られる特徴よりは，表本体から得られる内容的特徴の方が，統計文書のランキングには寄与していることが推察される．

5.4　考　　察

本節では，特徴の組み合わせによる nDCG@10 の悪化が見られた組み合わせについて考察する．図 6 に，各手法で再ランキングされた上位 10 件の文書の関連度の内訳を示す．

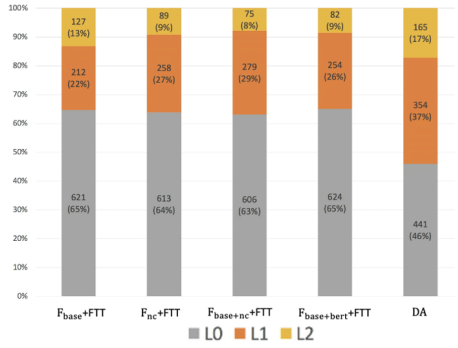

図 6　各手法で再ランキングされた上位 10 件の文書の関連度の内訳

F_{nc}+FTT では，F_{base}+FTT に比べて上位 10 件に含まれる L0 の文書数が減少しているが，L2 の文書数も減少しており，結果的に L1 の文書数が増加している．表本体の数値やその変化を表す内容的特徴は，L0 の文書数を減らす効果があるものの，L2 の文書数も減少させてしまっており，どちらともいえない評価の文書が多く残る傾向がみられる．

表 10 に，F_{base} から $F_{base+nc}$ へ変化させた際の上位 10 件の検索結果の変化を示す．表 10 の結果から，新規に含まれた L2 の文書数よりも，上位 10 件から除外された L2 の文書数が大きいこと，また，L0 の文書数についても同様のことが言えるが，その差が L0 よりも L2 の方がかなり大きいことがわかる．これは，表本体の数値やその変化を表す特徴は，関連度の高い文書のランキングを大きく下げてしまう傾向が強いことを示している．

表 10　F_{base} から $F_{base+nc}$ に変化させた際の上位 10 件の検索結果の変化

検索結果の変化の種別	L0	L1	L2	総数
上位 10 件内で順位が上昇した文書数	35	13	14	62
上位 10 件内で順位が下降した文書数	37	11	6	54
順位に変化がない文書数	11	1	2	14
上位 10 件に新規に含まれた文書数	523	254	53	830
上位 10 件から除外された文書数	533	192	105	830

次に F_{base} と $F_{base+bert}$ を比較する．$F_{base+bert}$+FTT による nDCG@10 の値は，F_{base}+FTT による値から 0.004 減少している．図 6 を確認すると，L2 の文書数が減少し，L0,L1 の文書数が増加していることがわかる．BERT を用いて得られる特徴 $f_{bert}^{q_i,t_j}$ は，統計文書のランキングを向上させるよりはむしろ阻害していることを示している．表 11 に，F_{base} から $F_{base+bert}$ へ変化させた際の上位 10 件の検索結果の変化を示す．表 11 の結果から，新規に含まれた L2 の文書数よりも，上位 10 件から除外された L2 の文書数が大きいことがわかる．これは，BERT から得られる特徴は，関連度の高い文書のランキングを大きく下げてしまう傾向が強いことを示している．

表 11　F_{base} から $F_{base+bert}$ に変化させた際の上位 10 件の検索結果の変化

検索結果の変化の種別	L0	L1	L2	総数
上位 10 件内で順位が上昇した文書数	61	17	12	90
上位 10 件内で順位が下降した文書数	55	15	6	76
順位に変化がない文書数	18	7	2	27
上位 10 件に新規に含まれた文書数	490	215	62	767
上位 10 件から除外された文書数	480	180	107	767

6　ま　と　め

本稿では，統計データの表本体の特徴と，ニューラル検索で用いられるニューラルネットワークモデルによる再ランキング手法を提案し，その性能を検証した．実験では，表本体の内容的特徴と表全体を用いた特徴，BERT を用いた特徴を使用して FT-Transformer による再ランキング手法の性能を検証した．

実験の結果，表本体の形式的特徴と文書の特定ゾーンを用いた特徴，表全体を用いた特徴の組合せに対して FT-Transformer を用いた再ランキングを行う手法が nDCG@10 で 0.254 となり，他の提案手法の組合せの中で最良の結果となった．しかし，先行研究で最良の結果を示したデータ補強を用いる手法の値 0.394 には届かなかった．

今後の課題としては，表本体の特徴抽出の手法を改善することが挙げられる．実験で用いた特徴は比較的単純なものであり，それのみでは nDCG@10 の値が改善しないことから，今後は，表の構造やより端的な内容を表現できる特徴の利用が考えられる．

謝　　辞

本研究の一部は科研費 18K11557 の助成を受けたものである．ここに記して感謝の意を表します．

文　　献

[1] Makoto P. Kato, Hiroaki Ohshima, Ying-Hsang Liu, and Hsin-Liang Chen. Overview of the NTCIR-15 data search task. In *Proceedings of the NTCIR-15 Conference*, 2020.

[2] Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. *CoRR*, Vol. abs/2106.11959, , 2021.

[3] Lya Hulliyyatus Suadaa, Lutfi Rahmatuti Maghfiroh, Isfan Nur Fauzi, and Siti Mariyah. Stis at the ntcir-15 data search task: Document retrieval re-ranking. In *Proceedings of the NTCIR-15 Conference*, 2020.

[4] Ryota Mibayashi, Pham HuuLong, Naoaki Matsumoto, Takehiro Yamamoto, and Hiroaki Ohshima. Uhai at the ntcir-15 data search task. In *Proceedings of the NTCIR-15 Conference*, 2020.

[5] Phuc Nguyen, Kazutoshi Shinoda, Taku Sakamoto, Diana Andreea Petrescu, Hung Nghiep Tran, Atsuhiro Takasu, Akiko Aizawa, and Hideaki Takeda. Nii table linker at the ntcir-15 data search task: Re-ranking with pre-trained contextualized embeddings, data content, entity-centric, and cluster-based approaches. In *Proceedings of the NTCIR-15 Conference*, 2020.

[6] Norbert Fuhr. Probabilistic Models in Information Retrieval. *The Computer Journal*, Vol. 35, No. 3, pp. 243–255, 06 1992.

[7] Howard Turtle and W Bruce Croft. Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems (TOIS)*, Vol. 9, No. 3, pp. 187–222, 1991.

[8] Christopher D.Manning,Prabhakar Raghavan,Hinrich Schutze. *Introduction to information retrieval*. Cambridge University Press, Cambridge, 2008.

[9] Peng Shi, Jinfeng Rao, and Jimmy Lin. Simple attention-based representation learning for ranking short social media posts. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2212–2217, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[11] Zeynep Akkalyoncu Yilmaz, Shengjin Wang, Wei Yang, Haotian Zhang, and Jimmy Lin. Applying BERT to document retrieval with birch. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pp. 19–24, Hong Kong, China, November 2019. Association for Computational Linguistics.

[12] Shuo Zhang and Krisztian Balog. Ad hoc table retrieval using semantic similarity. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, p. 1553–1562, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.

[13] Roee Shraga, Haggai Roitman, Guy Feigenblat, and Mustafa Canim. Ad hoc table retrieval using intrinsic and extrinsic similarities. In *Proceedings of The Web Conference 2020*, WWW '20, p. 2479–2485, New York, NY, USA, 2020. Association for Computing Machinery.

[14] Zhiyu Chen, Mohamed Trabelsi, Jeff Heflin, Yinan Xu, and Brian D. Davison. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, p. 589–598, New York, NY, USA, 2020. Association for Computing Machinery.

[15] Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. Tabert: Pretraining for joint understanding of textual and tabular data. *CoRR*, Vol. abs/2005.08314, , 2020.

[16] Gowthami Somepalli, Micah Goldblum, Avi Schwarzschild, C. Bayan Bruss, and Tom Goldstein. SAINT: improved neural networks for tabular data via row attention and contrastive pre-training. *CoRR*, Vol. abs/2106.01342, , 2021.

[17] Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu. TURL: table understanding through representation learning. *CoRR*, Vol. abs/2006.14806, , 2020.

[18] Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3277–3287, Online, August 2021. Association for Computational Linguistics.

[19] Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Wang. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. *Findings of EMNLP 2020*, 2020.

[20] Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Yang Wang, and William W. Cohen. Open question answering over tables and text. *CoRR*, Vol. abs/2010.10439, , 2020.

[21] Takuto Watarai and Masatoshi Tsuchiya. Developing dataset of Japanese slot filling quizzes designed for evaluation of machine reading comprehension. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pp. 6895–6901, Marseille, France, May 2020. European Language Resources Association.

[22] Mainichi Shimbun. Cd-mainichi shimbun 1995 data collection, nichigai associates, 1996.

[23] 岡本卓, 宮森恒. 被検索文書の絞り込みと補強, クエリ拡張に基づく統計データ向けアドホック検索. 情報処理学会論文誌 データベース, 2021.

[24] Shuo Zhang and Krisztian Balog. Ad hoc table retrieval using semantic similarity. *CoRR*, Vol. abs/1802.06159, , 2018.

[25] Zhiyu Chen, Mohamed Trabelsi, Jeff Heflin, Yinan Xu, and Brian D. Davison. Table search using a deep contextualized language model. *CoRR*, Vol. abs/2005.09207, , 2020.