時系列に並んだ複数のアナリストレポートを用いた株価動向予測

Stock Price Trend Forecast using Multiple Timeseries Analyst Reports

鈴木 雅弘 *1　坂地 泰紀 *1　和泉 潔 *1　石川 康 *2
Masahiro Suzuki　Hiroki Sakaji　Kiyoshi Izumi　Yasushi Ishikawa

*1東京大学大学院 工学系研究科　*2日興アセットマネジメント株式会社
School of Engineering, The University of Tokyo　Nikko Asset Management Co., Ltd.

本研究では，時系列に並んだ複数のアナリストレポートを用いて，アナリストの予想純利益と株価動向を予測する手法を提案する．1 時点のアナリストレポートの本文に加え，同じアナリストが同じ銘柄について過去に書いたアナリストレポートの本文も入力する．2 つのレポートの発行日の間隔の日数による減衰を反映した，Transformer Encoder を用いたモジュールを適用する．その結果，複数レポートの入力による精度向上の可能性を示した．

1. はじめに

投資家は投資のために，株価だけでなく企業の売上高，収益，経営状況や為替など多くの情報を調べる必要がある．機関投資家が行うような高度な財務分析は個人投資家には難しく，個人投資家にとっては理解しやすく投資に役立てやすい情報が求められる．個人投資家の増加に伴い，近年金融ドメインにおいて伝統的に用いられてきた財務情報や経済指標などのデータに対して，今まで利活用の少なかったオルタナティブデータの利用が進んでいる．オルタナティブデータとしては，POS 売上データ，IR やニュースなどのテキスト，記者会見動画や衛星写真など様々な領域に広がる．これらのオルタナティブデータは，機関投資家にとっては新しいリターンの源泉となりうる．

投資判断に活用しやすいオルタナティブデータの情報源として，注目を集めているのがアナリストレポートである．アナリストレポートは，ニュースやプレスリリース，株価の評価，マクロ経済のトレンドなどを考慮に入れたそれぞれの銘柄の評価を，証券会社に所属する，金融のプロフェッショナルであるアナリストが書いたレポートである．アナリストレポートには決算状況や株価，会社発表の情報などの事実だけでなく，これらの情報をベースにしたアナリストの将来の業績予測や，アナリストが企業に足を運ぶなどして獲得した独自の情報が含まれる．これらの情報はアナリストレポート特有の有益な情報である．アナリストレポート本文の有用性として，平松らによりアナリストレポートの文面に表れるセンチメント (レポートのトーン) が株価に関して有益な情報を保持している可能性が高いことが報告されている [1].

アナリストレポートの著者であるアナリストは，証券会社ごとにセクターや業種によって分かれ，同じアナリストが継続的に同じ企業についてアナリストレポートを発行することが多い．株価リターンに対するアナリストの株価レーティングや業績予想の相関性は低く，これらの有効性が失われていることが報告されているものの，短期的にはこれらの業績予想に市場が反応することが報告されている [2]．アナリストのレーティング変更や業績予想の変更は市場に即座に反応されるため，微小な変化の場合はこれらの数値に反映されない一方で，テキストへの書きぶりに反映されうる．そのため，以前に同じアナリストによって書かれたアナリストレポートを参照し過去と現在の書きぶりの変化を追うことで，レーティングや予想純利益などの数値として表出しない微小の変化を検出することができると仮定する．

本研究では対象のアナリストレポート 1 本だけでなく，同じアナリストが同じ銘柄について書いた過去のアナリストレポートも用いることで，予測実験における有効性を確かめる．予測実験として 2 つの実験を行う．1 つ目として，アナリストの予想した純利益の将来の変化率の予測 (純利益予測) を行う．2 つ目として，超過リターンの正負の予測 (株価動向予測) を行う．

連絡先: 113-8656 東京都文京区本郷 7-3-1 東京大学大学院工学系研究科システム創成学専攻 和泉研究室 鈴木雅弘
b2019msuzuki@socsim.org

2. 使用データ

本研究ではアナリストレポートの文章と，アナリストが予想した純利益の値，超過リターンの値を用いる．

2.1 アナリストレポート

国内の大手証券会社が 2016 年 1 月から 2020 年 9 月にかけて発行した 75,440 本のアナリストレポートを用いる．これらのアナリストレポートの中から，同じアナリストが同じ銘柄について書いたものを抽出し，時系列に並べる．時系列に 3 つ並んだアナリストレポートの組み合わせを 1 つの入力データとする．また，時系列順に隣接する 2 つのアナリストレポートの，発行日の間隔 (日数) も入力データとして取得する．合計で 67,220 件のデータセットが構築された．

2.2 純利益予測に用いるデータ

本研究では，アナリストの予想した純利益が将来において上昇するか，下降するかを予測するために，アナリストが予想した純利益の変化率を用いる．最初にアナリストの予想した純利益を求める．ある時点 t における対象銘柄の先 12ヶ月の予想純利益 (Forecasted Net Income) を NI(t) とおく．NI(t) は，時点 t において最新のレポートにおける今期予想純利益と来期予想純利益の按分によって算出する．按分を用いる理由としては，今期予想と来期予想のどちらかのみを使う形にすることで決算期を跨ぐタイミングでジャンプが生じてしまい，予想利益の対象期間が一定にならないためである．2022 年 1 月 30 日時点の例を考える．アナリストは多くの 3 月決算の企業について，2021 年 4 月から 2022 年 3 月までの今期純利益と，2022 年 4 月から 2023 年 3 月までの来期純利益を予想する．そのため，2022 年 1 月 30 日から先 12ヶ月の予想純利益は，2022 年 3 月までの今期 2ヶ月と 2022 年 4 月から 2023 年

1 月までの来期 10ヶ月の予想純利益の按分として求められる．NI′ を今期予想純利益，NI″ を来期予想純利益とすると，NI(t) は式 (1) によって求められる．

$$\mathrm{NI}(t) = \mathrm{NI}' \times \frac{2}{12} + \mathrm{NI}'' \times \frac{10}{12} \tag{1}$$

アナリストレポートの発行日の翌日から 3ヶ月後 (60 営業日後) の，アナリストの予想純利益の変化率 (Future Rate of change) を FR(60) とすると，これは式 (2) によって求められる．

$$\mathrm{FR}(60) = \begin{cases} \dfrac{\mathrm{NI}(t+60) - \mathrm{NI}(t)}{|\mathrm{NI}(t)|} & (\mathrm{NI}(t) \neq 0) \\ \mathrm{sgn}(\mathrm{NI}(t+60) - \mathrm{NI}(t)) & (\mathrm{NI}(t) = 0) \end{cases} \tag{2}$$

ここで sgn(·) は符号関数で，入力が正のときに 1，0 のときに 0，負のときに-1 を返す．本研究では，アナリストレポートが四半期に 1 回発行されることが多いことから，3ヶ月後 (60 営業日後) のアナリストの予想純利益の変化率 FR(60) を用いる．実験では FR(60) の大小によって 2 値分類を行う．FR(60) の大小の閾値は，使用データの中央値を用いる．

2.3 株価動向予測に用いるデータ

株価動向予測では，各銘柄のリターンがベンチマークを上回る部分である超過リターンの正負の予測を行う．そのため，各銘柄の株価とベンチマークの東証株価指数 (TOPIX) を用いる．アナリストレポートの発行日翌日とその 3ヶ月後 (60 営業日後) の，各銘柄の株価と TOPIX を取得し，これらをそれぞれ C_0, C_{60}, T_0, T_{60} とする．各指標は終値を用いる．アナリストレポートは市場への影響を避けるため 1 日の取引終了後に発行され，アナリストレポートからの情報は発行日の翌日に市場に織り込まれる．このことから本研究では発行日翌日の値を用いる．3ヶ月後 (60 営業日後) の超過リターン (Excess Return) を ER(60) とすると，これは式 (3) によって計算される．

$$\mathrm{ER}(60) = \frac{(C_{60} - C_0)}{C_0} - \frac{(T_{60} - T_0)}{T_0} \tag{3}$$

TOPIX のようなベンチマークと比較して相対的に評価される機関投資家にとって，単純なリターンよりも超過リターンに対する予測性が重要である．このことから本研究では超過リターンを用いる．それぞれのアナリストレポートに対して超過リターンについて正と負の 2 値の分類を行う．

3. 手法概要

本研究では，BERT[3]，Transformer[4]，Attention[5]，MLP を組み合わせた手法を用いる．複数のアナリストレポートを入力する場合，2 つのアナリストレポートの発行日数の間隔を入力する場合としない場合に分けて実験を行う．本研究では，BERT の他に用いる主なネットワークである Cross Transformer Module (CTM) を提案し，その内容について 3.1 節にて述べる．CTM や Transformer の Encoder 部分 (TE) を用い，3.2 節では 2 レポートを入力する場合，3.3 節では 3 レポートを入力する場合についてそれぞれ用いる手法を述べる．

3.1 Cross Transformer Module (CTM)

図 1 に Cross Transformer Module (CTM) の概要を示す．Transformer の Encoder 側で用いられている Layer を Transformer Encoder とする．Transformer の入力となる Query，Key，Value の行列をそれぞれ $Q \in \mathbb{R}^{d\times n}, K \in \mathbb{R}^{d\times n}, V \in \mathbb{R}^{d\times n}$ とする．ここで n は入力となる，BERT の出力の最終層の長さ，d は入力となる BERT の出力の最終層の隠れ層の次元数である．Transformer Encoder の出力を $\mathrm{Transformer}(Q,K,V) \in \mathbb{R}^{d\times n}$ と表す．図 1 の Transformer Encoder の出力 $\mathrm{TE}_{t-d} \in \mathbb{R}^{d\times n}$(左側)，$\mathrm{TE}_t \in \mathbb{R}^{d\times n}$(右側) は，$t-d, t$ 時点のアナリストレポートの本文に BERT を適用して得られる $H_{t-d} \in \mathbb{R}^d, H_t \in \mathbb{R}^d$ を用いて，それぞれ式 (4)，式 (5) のように表される．

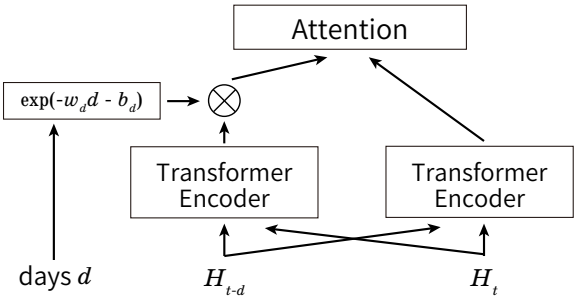

図 1: 本実験で使用する Cross Transformer Module (CTM) の概要図．アナリストレポートを入力した BERT から出力された H_t と一期前 (d 日前) のアナリストレポートの出力 H_{t-d} は，2 つの Transformer Encoder に入力される．2 レポート間の日数 d が入力される場合，$\exp(-w_d d - b_d)$ に変換されたのち，H_{t-d} 側の Transformer Encoder の出力に乗算される．2 つの Transformer Encoder の出力は結合され Attention によって重みづけされる．

$$\mathrm{TE}_{t-d} = \mathrm{Transformer}(H_{t-d}, H_t, H_t) \tag{4}$$

$$\mathrm{TE}_t = \mathrm{Transformer}(H_t, H_{t-d}, H_{t-d}) \tag{5}$$

以降では，Transformer Encoder による式 (5) の処理を $\mathrm{TE}(H_t, H_{t-d}, H_{t-d})$ と表す．式 (6) のように，2 つの Transformer Encoder の出力を次元数を変換し結合する．その際，2 つのレポートの発行日の間隔 d を入力する場合，$t-d$ 時点から日数 d が経過することで，$t-d$ 時点のレポートの情報の価値が減衰することを表現する．そのために，d は学習可能なパラメータ $w_d \in \mathbb{R}, b_d \in \mathbb{R}$ を用いて $\exp(-w_d d - b_d)$ とした上で TE_{t-d} に乗算して入力する．

$$C = [\mathrm{Flatten}(\mathrm{TE}_t); \mathrm{Flatten}(\exp(-w_d d - b_d)\mathrm{TE}_{t-d})] \tag{6}$$

ここで，Flatten(·) は次元数を (d,n) から $d\times n$ に変換する操作であり，$C \in \mathbb{R}^{dn\times 2}$ である．式 (7) のように，Attention の重み $a_{\mathrm{CTM}} \in \mathbb{R}^2$ によって重みづけされ，CTM の出力を得る．

$$\mathrm{CTM}(H_t, H_{t-d}, d) = \mathrm{Reshape}(a_{\mathrm{CTM}} C^{\mathsf{T}}) \tag{7}$$

ここで，Reshape(·) は次元数を $d\times n$ から (d,n) に変換する操作を指し，Attention の重み $a_{\mathrm{CTM}} \in \mathbb{R}^2$ は式 (8) によって表される．

$$a_{\mathrm{CTM}} = w_1 C + b_1 \tag{8}$$

ここで，$w_1 \in \mathbb{R}^{dn}, b_1 \in \mathbb{R}^{dn}$ は学習可能なパラメータである．以上より，2 つのアナリストレポートを適用した BERT からの出力 H_t, H_{t-d} と 2 つのアナリストレポートの発行日の間隔 d から，Cross Transformer Module (CTM) の出力 $\mathrm{CTM}(H_t, H_{t-d}, d)$ を求める．

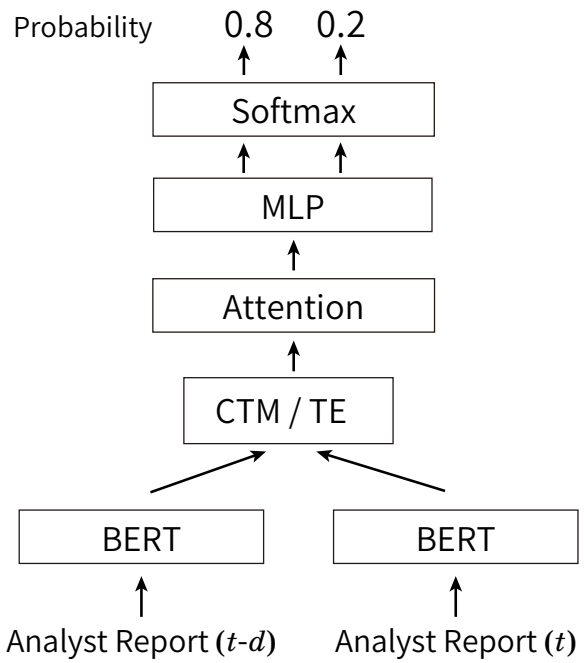

図 2: 2 時点のアナリストレポートを入力する際に用いるネットワークの概要図．CTM/TE は CTM(Cross Transformer Module) または TE (Transformer Encoder) のどちらかを用いることを示す．Transformer を用いる場合，2 つのレポート間の日数 d は用いない．

3.2 2 レポートを入力する場合

図 2 に，2 つのアナリストレポートを入力する場合に用いるネットワークの概要を示す．時点 $t, t-d$ におけるアナリストレポートのトークン列をそれぞれ $\boldsymbol{w}_t, \boldsymbol{w}_{t-d}$ とする．ここで，入力トークン列から得られる BERT の最終層の隠れ層の出力を $\mathrm{BERT}(\cdot) \in \mathbb{R}^{d\times n}$ と表す．ここで n は入力トークン列の長さ，d は入力となる BERT の出力の最終層の隠れ層の次元数である．$\boldsymbol{w}_t, \boldsymbol{w}_{t-d}$ に重みを共有している BERT を適用し，式 (9)，(10) より $H_t \in \mathbb{R}^{d\times n}, H_{t-d} \in \mathbb{R}^{d\times n}$ を得る．

$$H_t = \mathrm{BERT}(\boldsymbol{w}_t) \tag{9}$$

$$H_{t-d} = \mathrm{BERT}(\boldsymbol{w}_{t-d}) \tag{10}$$

それぞれの BERT の最終層の出力は，CTM または TE (Transformer Encoder) に入力される．TE が使用される場合式 (11) によって，CTM が使用される場合式 (12) によって，$H \in \mathbb{R}^{d\times n}$ が得られる．

$$H = \mathrm{TE}(H_t, H_{t-d}, H_{t-d}) \tag{11}$$

$$H = \mathrm{CTM}(H_t, H_{t-d}, d) \tag{12}$$

ここで d は，2 つのアナリストレポートの発行日の間隔の日数である．また $\mathrm{TE}(H_t, H_{t-d}, H_{t-d})$ は，式 (5) における $\mathrm{Transformer}(H_t, H_{t-d}, H_{t-d})$ と同じ操作を指す．CTM や TE の出力 H は，式 (13) のように Attention の重み $a_{\mathrm{text}} \in \mathbb{R}^n$ によってトークン位置の方向に重みづけされる．

$$h_{\mathrm{text}} = a_{\mathrm{text}} H^{\mathsf{T}} \tag{13}$$

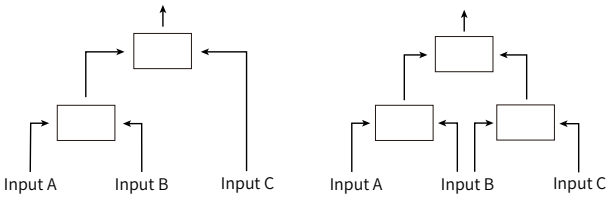

図 3: 3 時点のアナリストレポートを統合的に処理する際の概要図．(左) トーナメント型．最初に左側の Input A と InputB を処理する．この出力と，Input C の出力を処理し，その出力を最終的な出力とする．(右) 隣接型．隣接する入力の組 (Input A と Input B，Input B と Input C) を，それぞれ処理する．これら 2 組の出力を再度処理し，最終的な出力とする．

a_{text} は式 (14) によって求められる．

$$a_{\mathrm{text}} = w_2 H + b_2 \tag{14}$$

ここで，$w_2 \in \mathbb{R}^d, b_2 \in \mathbb{R}^d$ は学習可能なパラメーターである．$h_{\mathrm{text}} \in \mathbb{R}^d$ はその後 3 層の MLP で処理され，2 値分類の出力 $y \in \mathbb{R}^2$ を得る．

$$r = \mathrm{ReLU}(W_r \cdot h_{\mathrm{text}} + b_r) \tag{15}$$

$$y = \mathrm{softmax}(W_y \cdot r + b_y) \tag{16}$$

ここで $W_r \in \mathbb{R}^{d\times d}, W_y \in \mathbb{R}^{2\times d}$ は重み行列，$b_r \in \mathbb{R}^d, b_y \in \mathbb{R}^2$ はバイアスベクトルで，どちらも学習可能なパラメーターである．

3.3 3 レポートを入力する場合

3 レポートを入力する場合，どの 2 つの情報を比較するかにより，トーナメント型と隣接型に分けることができる．トーナメント型と隣接型の概念図を図 3 に示す．トーナメント型では，まず時系列順に過去の 2 つ (図 3 左における Input A と Input B) を処理する．その出力と時系列順に最新の入力 (図 3 左における Input C) を処理することで出力を得る．隣接型では，まず時系列順に隣接する 2 つの入力の組 (図 3 右における Input A と Input B，Input B と Input C の 2 組) をそれぞれ処理する．それぞれの組からの出力を，さらに処理することで出力を得る．Attention 以降の処理については 3.2 節と同じであるため，以降は Attention より前の処理に着目して述べる．

図 4 にトーナメント形式で 3 レポートを処理する際の概要を示す．$t, t-d_1, t-d_2$ 時点のアナリストレポートのトークン列 $\boldsymbol{w}_t, \boldsymbol{w}_{t-d_1}, \boldsymbol{w}_{t-d_2}$ は，それぞれ重みが共有された BERT で式 (17)，(18)，(19) のように処理される．

$$H_t = \mathrm{BERT}(\boldsymbol{w}_t) \tag{17}$$

$$H_{t-d_1} = \mathrm{BERT}(\boldsymbol{w}_{t-d_1}) \tag{18}$$

$$H_{t-d_2} = \mathrm{BERT}(\boldsymbol{w}_{t-d_2}) \tag{19}$$

まず $t-d_1, t-d_2$ の 2 時点のレポートの BERT の出力 $H_{t-d_1} \in \mathbb{R}^{d\times n}, H_{t-d_2} \in \mathbb{R}^{d\times n}$ が TE または CTM によってそれぞれ式 (20), (21) のように処理される．ここで n は入力トークン列の長さ，d は入力となる BERT の出力の最終層の隠れ層の次元数である．

$$H'' = \mathrm{TE}(H_{t-d_1}, H_{t-d_2}, H_{t-d_2}) \tag{20}$$

$$H'' = \mathrm{CTM}(H_{t-d_1}, H_{t-d_2}, d_2 - d_1) \tag{21}$$

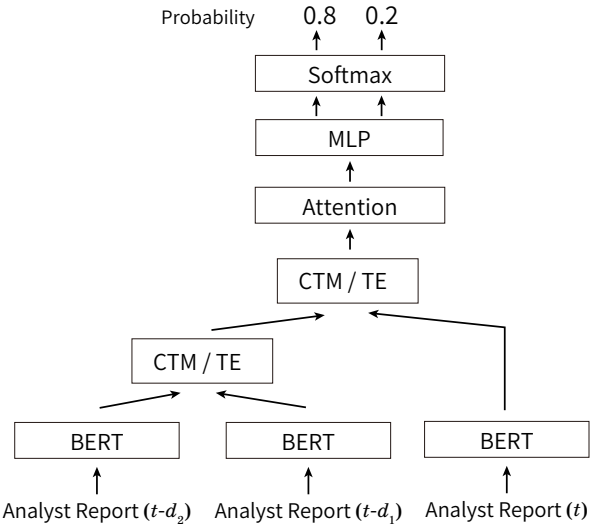

図 4: 3 時点のアナリストレポートをトーナメント形式で処理する際に用いるネットワークの概要図．$0 < d_1 < d_2$ である．CTM/TE は CTM(Cross Transformer Module) または TE (Transformer Encoder) のどちらかを用いることを示す．Transformer を用いる場合，2 つのレポート間の日数 d は用いない．3 つの BERT と，3 つの CTM または TE の重みはそれぞれ共有される．

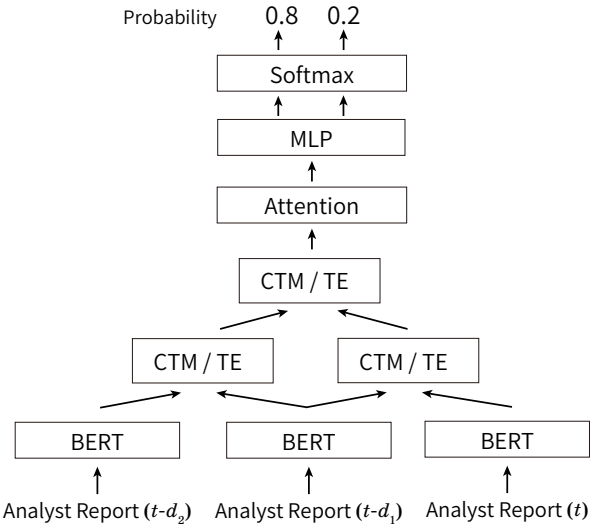

図 5: 3 時点のアナリストレポートを隣接形式で入力する際に用いるネットワークの概要図．$0 < d_1 < d_2$ である．CTM/TE は CTM(Cross Transformer Module) または TE (Transformer Encoder) のどちらかを用いることを示す．Transformer を用いる場合，2 つのレポート間の日数 d は用いない．3 つの BERT と，3 つの CTM または TE の重みはそれぞれ共有される．

この出力 $H'' \in \mathbb{R}^{d\times n}$ と，t 時点のレポートの BERT の出力 $H_t \in \mathbb{R}^{d\times n}$ を同様に TE または CTM によってそれぞれ式 (22)，(23) のように処理する．

$$H = \mathrm{TE}(H_t, H'', H'') \tag{22}$$

$$H = \mathrm{CTM}(H_t, H'', d_2 - d_1) \tag{23}$$

2 つの CTM または TE の重みは共有される．H を得た以降の処理は，3.2 節の式 (13) 以降と同じ処理を行う．

図 5 に隣接形式で 3 レポートを処理する際の概要を示す．トーナメント形式と同様，$t, t-d_1, t-d_2$ 時点のレポートは，それぞれ重みが共有された BERT で処理され，それぞれ $H_t \in \mathbb{R}^{d\times n}, H_{t-d_1} \in \mathbb{R}^{d\times n}, H_{t-d_2} \in \mathbb{R}^{d\times n}$ を得る．$t-d_1, t-d_2$ の 2 時点のレポートの BERT の出力 H_{t-d_1}, H_{t-d_2} はトーナメント型と同様式 (20), (21) のように TE または CTM によって処理される．$t, t-d_1$ の 2 時点のレポートの BERT の出力 H_t, H_{t-d_1} も式 (24), (25) のようにそれぞれ TE または CTM によって処理される．

$$H' = \mathrm{TE}(H_t, H_{t-d_1}, H_{t-d_1}) \tag{24}$$

$$H' = \mathrm{CTM}(H_t, H_{t-d_1}, d_1) \tag{25}$$

2 つの TE または CTM の出力 H'', H' を，式 (26)，(27) のようにさらに TE または CTM に入力し処理する．

$$H = \mathrm{TE}(H', H'', H'') \tag{26}$$

$$H = \mathrm{CTM}(H', H'', 0) \tag{27}$$

3 つの TE または CTM の重みは全て共有される．H を得た以降の処理は，3.2 節の式 (13) 以降と同じ処理を行う．

4. 実験

本研究では，同じアナリストが同じ銘柄について書いた複数のアナリストレポートと，そのアナリストレポートの発行日の間隔を入力とし，2.2 節で述べたアナリストが予想した純利益の変化率の大小と，2.3 節で述べた超過リターンの正負の 2 値分類の予測を行う．構築したアナリストレポートによるデータセットに対し，入力するレポートの本数，CTM(Cross Transformer Module) と TE(Transformer Encoder) のどちらを使用するか，発行日の間隔を入力するかについて組み合わせ，表 1 のように 10 通りの入力方法を純利益予測と超過リターン予測のそれぞれに対して行う．データセットのうち，時系列順に新しい方から 20%をテストデータに割り当てる．残ったデータセットのうち 85%(全体のデータセットの 68%) を学習データに，15%(全体のデータセットの 12%) を開発データに割り当てる．BERT の事前学習モデルには，東北大学が公開している Wikipedia によって事前学習を行った base モデル [*1] を用いる．

5. 結果と考察

結果を表 1 に示す．両実験において，最も精度が高かったのはどちらもアナリストレポート 3 本をトーナメント型で入力し，TE (Transformer Encoder) を用い，アナリストレポートの発行日の間隔日数を入力しないものとなった．複数のレポートを用い，最新のレポートが他のレポートより出力に近くなるトーナメント型によって効率的に処理される．そのため複数レポートの活用には効果がある可能性がある．また，2 つの入力の情報は CTM ではなくより簡単な構造の TE での処理で十分なことによって TE による精度が高くなったと考えられる．一方で，アナリストレポートの発行日の間隔を入力しても，精

*1 https://huggingface.co/cl-tohoku/bert-base-japanese

表 1: 実験の入力方法とその結果．評価方法は Macro-F1 である．3 レポート数を入力する場合，3.3 節で述べたトーナメント型と隣接型の 2 つの入力方法で実験を行う．CTM は Cross Transformer Module，TE は Transformer Encoder を指す．レポート数が 1 の入力は，BERT と MLP のみを組み合わせたネットワークで処理する．太字は各実験において最も精度が高かったことを示す．

レポート数	CTM/TE	発行日間隔の入力	純利益予測の F1	超過リターン予測の F1
1	なし	なし	.585	.512
2	TE	なし	.577	.512
2	CTM	あり	.587	.514
2	CTM	なし	.578	.515
3(トーナメント型)	TE	なし	**.589**	**.523**
3(トーナメント型)	CTM	あり	.588	.515
3(トーナメント型)	CTM	なし	.581	.494
3(隣接型)	TE	なし	.560	.508
3(隣接型)	CTM	あり	.571	.502
3(隣接型)	CTM	なし	.583	.508

度の向上は見られなかった．この理由としては 2 つ考えられる．1 つはアナリストレポートの多くが四半期に一度発行されるため，レポートの発行日の間隔が入力データセットごとに大きく異なることがなかったためである．2 つ目としては，日数が情報の減衰性の大きさにほとんど寄与しないことが考えられる．

アナリストレポート 3 本を隣接型で入力したものは，アナリストレポート 1 本を入力したものより精度が低かった．これは，隣接型において入力の中で最新のアナリストレポートが必ず 2 回 CTM や TE を経由し，出力まで遠くなるため最新の情報が伝達しにくいことが理由であると考えられる．そのため，複数時点の入力を用いる際は，RNN のように最新時点の入力が最も出力に近くなるように入力した方が良い可能性が高い．

6. まとめ

本研究では，対象のアナリストレポート 1 本だけでなく，同じアナリストが同じ銘柄について書いた過去のアナリストレポートの活用を目的に，純利益予測と超過リターン予測による実験を行いその有効性を検証した．また複数レポートの統合手法として，TE(Transformer Encoder) を組み合わせた CTM(Cross Transformer Module) を提案した．その際，2 つのレポートの発行日の間隔を CTM に入力し，TE の出力に乗算することで過去のレポートの情報の価値の減衰を表現した．実験の結果，3 本のアナリストレポートをトーナメント型の TE によって処理したモデルの精度が両実験において最も高くなった．複数のレポートの活用の効果の可能性が現れた．また，最新の情報が効率的に処理されるトーナメント型と CTM に比べ簡素な構造の TE の組み合わせが効果的だった．その一方で，アナリストレポートの発行日の間隔の日数の入力や隣接型の入力による効果は見られたなかった．

今後の課題として，決算短信や有価証券報告書などの，アナリストレポート以外に定期的に発行される金融テキストでの実験が考えられる．過去のテキストとの内容の差分を抽出したり，それによって過去と異なる情報に着目でき，効率的な処理が行われることが期待される．

参考文献

[1] 平松賢士, 三輪宏太郎, 酒井浩之, 坂地泰紀. アナリストレポートのトーンの情報価値. 証券アナリストジャーナル, Vol. 59, No. 2, pp. 86–97, 2021.

[2] 伊藤拓之. 証券アナリストの役割と市場の反応, 2010. https://www.nli-research.co.jp/files/topics/38649_ext_18_0.pdf.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, Vol. abs/1810.04805, , 2018.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30 (NeurIPS)*, pp. 5998–6008. Curran Associates, Inc., 2017.

[5] Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In *The Proceedings of 3rd International Conference on Learning Representations (ICLR 2015)*, 2015.