—論文題目—

関係推論能力を有する Neural Turing Machine

.

指導教授

萩原 将文 教授

慶應義塾大学 理工学部 情報工学科

令和 4 年度

学籍番号 61913674

東明 鴻希

目 次

あらまし

Neural Turing Machine(NTM) のような外部メモリを持つニューラルネットワークモデルは情報を長期的に保存し活用する能力を持つが、メモリ内容間の複雑な推論を行う能力に課題がある。関係ネットワークはエンティティ間の関係を計算でき、高度な推論能力を必要とするタスクを解決可能である。本論文では NTM のメモリ構造を保ったまま、メモリ内容に関係推論を適用することを提案する。これにより既存の関係推論ネットワークよりも高度な忘却方式、説明可能性および拡張性を持つモデルを実現する。 想定される課題に対処するため、二段階の順序整理モジュールも新しく提案する。一段階目のグラフアテンション構造は時間的に隣接する項目間の推論を行う。二段階目のモジュールはメモリをソートしたのちに全項目間の関係を計算し、関係情報専用のメモリに保存する。実験結果は

第1章

はじめに

ニューラルネットワークモデルの一つである Recurrent Neural Networks(RNN) は、時系列データの学習に広く用いられる。入力の順伝播を繰り返す最もシンプルな RNN では、長時間の入力にわたって逆伝播を計算する中で勾配の値が極端に小さくあるいは大きくなるという問題が発生する。Long-Short Term Memory(LSTM)[] はゲーティング機構によりこれらの問題に対処した。また、メモリセルの存在により長時間にわたり情報を保持する能力が向上した。しかしメモリセルのサイズや複雑さの制限により、多くの情報を長期保持する能力にはまだ限界がある [要出典]。

メモリネットワークはメモリとして利用可能な行列を有し、各時間ステップでの入力情報をメモリに読み書きする能力を学習する。End-To-End Memory Networks[1] は入力の埋め込みを外部メモリとして保持することで、入力中の長い時間を隔てた関係を解釈できる。しかし入力系列の各項目を全て保存する必要があるため、時間方向へのスケーリング能力に課題がある。

Neural Turing Machine(NTM) [3] が有するメモリは固定長であるため、入力の長期化に伴う計算量の増加を回避できる。メモリスロットの数を超えるサイズの入力系列に対応できるのは、メモリの特定の行を忘却/上書きする能力による。Differentiable Neural Computer(DNC)[4] は NTM の読み出し/書き込み重みの計算部分をより複雑にしたモデルである。追加モジュールの一つである usage vector は、各スロットの使用率を記憶する隠れ状態である。書き込み重みの算出に利用することで、空いているスロットに優先的に書き込める。別の追加モジュールである temporal link matrix は、スロット間の前後関係を記憶す

るメモリである。読み出し重みの計算に利用することで、着目するスロットの前後の時間で書き込まれたスロットも考慮できる。Convertible Short-Term and Long-Term Memory in DNC (CSLM) [5] は、DNC の長期記憶能力をさらに改善する。メモリの各スロットには重要度が割り振られており、これは学習を通して変化可能である。重要度の割り当てが低いスロットは忘却されやすく、高いスロットはその逆になる。これは DNC のメモリがスロット単位で長期記憶と短期記憶に分けられており、その割り当ては適応的かつ学習可能であることを意味する。

これら NTM ベースのモデルは学習を通して書き込みの位置や読み出す内容を最適化できる。しかしメモリの項目間の関係を計算する機構を持たないため、最短経路探索のような入力実体間の関係推論を要求するタスクでの性能に劣る。

関係ネットワークは入力や記憶内の実体間の関係推論能力を持つ。関係の計算は self-attention[Transformer] をベースにした手法で行われる。Relational recurrent neural networks (R-RNN) [6] で提案された Relational Memory Core(RMC) は、毎時間ステップにおいて入力と全メモリスロット間での関係情報を計算し、その情報でメモリを更新する。Self-Attentive Associative Memory (SAM) [8] で提案された SAM-based Two-memory Model(STM) は、入力を保存する項目メモリと、保存した項目間の関係情報を保存する関係メモリの 2 種類のメモリを持つ。項目を忠実に復元することを要求するタスクにおいて、関係メモリのみを持つ RMC よりも高い精度を示した。NTM のように項目がメモリスロット間で区分けされるモデルと異なり、SAM の 2 つのメモリは分散型のメモリである。項目メモリは入力の outer product を保存する自己連想記憶、関係メモリは項目メモリから抽出した項目間の outer product を保存する相互連想記憶として実装されており、各入力がメモリ全体に分散するためである。

本論文では NTM のメモリ構造を保ったまま関係推論モジュールを追加し、メモリ項目間の関係推論能力を追加することを提案する。関係メモリには RMC を用いるため、提案ネットワークは NTM を項目メモリ、RMC を関係メモリとして持つ 2 メモリモデルとなる。同じ 2 メモリモデルである SAM と比較すると、項目メモリが非分散型メモリであり各行で記憶内容が区分けされているあ

る点が異なる。これにより各入力項目を選択的に忘却・上書きすることが可能になり、R-RNN や SAM のような分散記憶+LSTM 方式の忘却よりも高度な忘却が行えると考えられる。従って長期記憶保持が必要なタスクでより優れた性能を示すことが期待される。CSLM が各スロットに忘却強度を割り振ったように、読み書き・忘却方式に拡張性・柔軟性があることも利点である。またこれらのメモリの読み書きはアテンションに由来する説明可能性がある。アテンション係数を可視化することで読み書きした番地を追跡可能である。

提案ネットワークに考えられる課題として、NTM 内部で項目の並び順に一貫性が無くなることで、関係メモリの学習を阻害することが予想される。これは入力との類似度に応じてスロットの忘却・上書きを行う機構や、DNC では空いた場所から優先的に書き込む機構が関係推論の計算と独立していることによる。この問題は入力が長期化し、忘却が頻繁に行われる場合により顕著になると考えられる。メモリ全体を常に一貫性をもって解釈するために、以下の 2 種類のモジュールを提案する。全体としてはこれらのモジュールが項目メモリの順序を整理・解釈した後に関係メモリに入力する構造となる。

１．DNC の時間リンク行列はスロット間の時間的な前後関係を表すグラフと捉えられる。これを元にメモリ項目にグラフアテンション (GAT)[9] をかけることで、時間的に局所的な情報 (文脈) を考慮した情報へと解釈できる。

２．各スロットの書き込み頻度をもとにソートする。

End-To-End Memory Networks[1] のようにメモリ内でデータが時系列で並ぶモデルや SAM[8] のように項目がメモリ全体に分散するモデルでは学習を通してメモリの解釈は一貫していると考えられるため、順序整理モジュールは新しい取り組みである。{要サーベイ}

この論文の貢献予定は以下に示す通りである

１．NTM に関係推論能力を付与する。このとき関係メモリの入力から一貫性が失われることが予想される。この課題を GAT と書き込み頻度ソートの 2 種類のモジュールにより解決する。

２．非分散型のメモリの実装により、分散型メモリよりも長期記憶が必要なタスクに適している。

３． 非分散型のメモリにはCSLMに代表されるように拡張性・柔軟性がある。またアテンションを用いる読み書きにより、メモリ操作は説明可能性を持つ。

第2章

関連研究

2.1 Neural Turing Machine

NTM の構造は 3 つのコンポーネントに分けられる。
１．情報を保存するメモリ M_t は $N \times W$ 次元の行列からなり、これは W 次元の項目を N スロット分保存できる。
２．コントローラは毎時間ステップで入力 x_t を受け取り、隠れ状態 h_t を計算する。任意の RNN がコントローラとして採用可能だが、大抵 LSTM が用いられる。
３．読み出し/書き込みヘッドはコントローラ出力 h_t とメモリ M_t の内容から読み出し/書き込み重みを計算し、メモリへの読み書きを行う。読み出し/書き込み重みはそれぞれ各スロットへの書き込み/読み出しの強度を表す N 次元のアテンション係数である。この重みを計算する操作を ”アドレス指定 ” 操作と呼ぶ。

2.1.1 節ではヘッドにおける読み書きを説明する。2.1.2 節ではアドレス指定を説明する。

2.1.1 読み出し/書き込みヘッド

NTM は読み出し用のヘッドと書き込み用のヘッドそれぞれを 1 つ以上持ち、ヘッドごとにアドレス指定及び読み書きが行われる。読み/書き重みをそれぞれ w_t^r,w_t^w と表し、これらは 2.1.2 節で説明するアドレス指定操作を用いて計算され

る。w_t^r,w_t^w は式〜、〜を満たすアテンション係数である。

$$\sum_i w_t(i) = 1 \tag{2.1}$$

$$0 \leq w_t(i) \leq 1, \forall i \tag{2.2}$$

ここで $w_t(i)$ は w_t の i 番目の要素である。メモリからの読み出しは式〜のように計算され、読み出しベクトル r_t を得る。

$$r_t = \sum_i w_t(i) M_t(i) \tag{2.3}$$

$M_t(i)$ は M_t の i 行目を表す。書き込みは式〜による忘却、式〜による加算の順で計算され、M_{t-1} を M_t に更新する。忘却ベクトル e_t、書き込みベクトル a_t はコントローラ出力からの変換で得られる。ただし忘却ベクトルの各要素は (0,1) の範囲にある。

$$M'_t(i) = M_{t-1}(i)[1 - w_t(i)e_t] \tag{2.4}$$

$$M_t(i) = M'_t(i) + w_t(i)a_t \tag{2.5}$$

2.1.2 アドレス指定操作

2.2 Relational Memory Core

関係メモリのベースとして使用した、[R-RNN] で提案された RMC を説明する。RMC は毎ステップで入力 x_t とメモリ M_{t-1} の各項目間の関係情報を計算し、そのステップでのメモリ成分 $\tilde{M}$ を用意する。LSTM ベースのゲーティングを利用して、$\tilde{M}$ を M_{t-1} に合成し M_t を得る。$\tilde{M}$ の計算はマルチヘッドドット積アテンション [Transformer] を用いて行われる。RMC は項目メモリ M_i と入力 x_t からアテンションを計算するために、クエリ・キー・バリューを計算する為の訓練可能な線形層を有する。それぞれを W_q,W_k,W_v と表現すると、$\tilde{M}$ は式〜のようにして計算される。

$$\tilde{M} = softmax(\frac{M_{t-1}W_q([M_{t-1}; x_t]W_k)^T}{\sqrt{d_k}})[M_{t-1}; x_t]W_v \tag{2.6}$$

ここで d_k はキーベクトルの次元　、[M;x] は M に x_t を新たな行として連結した $(N+1) \times M$ 次元の行列を表す。クエリ行列の計算では [M:x] ではなく M を入力することに注意が必要である。これは $\tilde{M}$ の次元を M_t と等しくすることを目的としている。

ヘッドが複数存在する時は、ヘッドごとに独立な線形層を用いたアテンション計算結果を結合し最終的な $\tilde{M}$ を得る。各ヘッドの計算結果を $\tilde{M}_1, \tilde{M}_2 \cdots \tilde{M}_h$ と表すとき、それぞれの次元は $N \times (M/h)$ であり、$\tilde{M}_t = [\tilde{M}_1 : \cdots : \tilde{M}_h]$ とすることで M_t と同じ次元の $\tilde{M}$ を得る。[:] は行方向の連結を表す。

$\tilde{M}$ により M_t を更新するために LSTM を利用する。M_t の各行を 2D-LSTM の各メモリセルとして実装することで、式 x〜y によって更新される。m_i,$\tilde{m_i}$ はそれぞれ M_t,$\tilde{M}$ の i 番目の行を表す。

$$= \tag{2.7}$$

$$= \tag{2.8}$$

$$= \tag{2.9}$$

式〜において下線部が示す箇所は LSTM からの変更部分である。関数 g は既存研究 [R-RNN] に従い、MLP + layer normalization として実装した。パラメータ は各 m_i について共通する。

第3章

NTMへの関係推論能力の付与

提案ネットワークは2種類のメモリを持つモデルとして構成される。アーキテクチャの全体図を図〜に示す。ネットワークの構造は大きく3つのモジュールに分けられる。1つ目は入力の保存・忘却・読み出しを行う項目メモリ M_t^I とそのオペレータ。2つ目は項目メモリの項目間の関係推論を行い、計算した関係情報を保存する関係メモリ M_t^R とそのオペレータ。3つ目は項目メモリの内容を関係メモリに入力する前に項目メモリの項目間順序を整理する順序整理モジュールである。ネットワークはコントローラLSTMからの出力 o_t^I、項目メモリからの読み出しベクトル r_t^I、関係メモリからの読み出しベクトル r_t^R を結合したのちlogistic sigmoid活性化を行い最終的な出力とする。

項目メモリには2.1節で導入したNTMを用いる。関係メモリには2.2節で導入したRMCに変更を加えたものを利用し、この変更は3.1節で説明する。3.2節では新しく提案する順序整理モジュールを2種類説明する。

3.1　提案ネットワーク中の関係メモリ

RMCに変更を加え、項目メモリの各項目間の関係情報を保存する関係メモリとして提案ネットワークに実装する。時間tにおける関係情報の計算のために、式〜における入力 x_t とメモリ M_t^R にまたがるアテンションを項目メモリ M_t^I と関係メモリ M_t^R の連結に対するアテンションに拡張する。変更後は式〜が示すようにして $\tilde{M}$ が計算される。

また M_{t-1}^{R} の更新時には、入力 x_t の代わりに項目メモリへの書き込みベクトル w_t を利用する。従って式〜は式〜のように変更される。

3.2 順序整理モジュール

3.2 節では項目メモリの順序を解釈・整理する 2 種類の機構を説明する。3.1 節で述べた拡張により、NTM のメモリを関係メモリへの入力とすることが可能となる。しかしこの実装では項目メモリにおける忘却や上書きは関係メモリの更新と独立して実行されるため、関係メモリへの入力の一貫性が損なわれる問題が予想される。そこで項目メモリを整理し一貫性を持った入力を生成する順序整理モジュールを提案する。

3.2.1 書き込み頻度ソート

項目メモリのヘッドによるメモリの忘却・上書きを疑似的に読み取る手段として、書き込み頻度によるソートを提案する。時間 t における各行の書き込み頻度 $q_t(N^I$ 次元ベクトル) は、式〜が示すように $t = 1 \sim t$ での書き込み重みの総和とする。

関係メモリは式〜において M_t^I の代わりに f_q に基づいてソートされた M_t^I を入力として受け取る。

3.2.2 時間リンク行列を利用したグラフアテンション

DNC[] では各ステップの内容を保存するだけでなく、入力系列内での前後関係の情報を保存するために時間リンク行列を実装している。時間リンク行列 L_t は式〜によって示されるように、時間的に隣接する書き込み重みの outer production を保存する。

第一項は項目の上書きが発生した時、その位置のリンクをリセットするための項である。p_t は式〜のようにして計算される。

提案モジュールでは時間リンク行列を各行が時間的に隣接する程度を表現するグラフと解釈して、グラフアテンション (GAT)[] による各項目の変換を試みる。これにより関係メモリに入力される各項目の特徴量に時間的な順序情報が内包されることが期待できる。グラフアテンションによる解釈は各行の位置を変更しないため、2.4.1 節の書き込み頻度ソートと併用できる。

第4章

評価実験

4.1　実験概要

複数のタスクでネットワークを評価する。実験の目的を以下に示す

・4.2 節では順序整理モジュールの有効性を ablation study により検証する

・4.3 節では提案する非分散型メモリの長期依存関係の計算能力を分散型の既存手法と比較する

・4.4 節では提案ネットワークにおける関係推論能力を評価する

・4.5 節ではより複雑な関係推論を必要とするタスクにおける性能をベースラインの関係推論ネットワークと比較する

全てのタスクにおいてネットワークは LSTM コントローラを採用し、ロジスティックシグモイド出力層を有する。クロスエントロピーを損失関数として訓練し、最適化手法としては Adam を用いた。また、各入力系列の開始時にネットワークの隠れ状態はリセットされる。各タスクにおけるネットワークのパラメータ、データのバッチサイズ、学習率を表 X に示す。

4.2　実験１　Ablation Study

4.1.1 節及び 4.1.2 節ではネットワークの基礎的な記憶と項目整理の能力を評価する。NTM[] の評価で用いられたシンプルなアルゴリズムタスクを説明する。

4.2.1 associative recall

一定の長さのランダムに生成されたバイナリベクトルからなる系列を 1 アイテムとする。始めにネットワークにはランダムな数のアイテムが入力される。入力アイテム系列が提示された後、クエリとして入力アイテムのうちの一つが改めて入力される。この時ターゲットとして、入力アイテムの中でクエリアイテムの次に提示されたアイテムを取る。各アイテムの間および入力とクエリの間にはデリミタを表すベクトルが挿入される。NTM[] 中の設定に従い、ベクトルの次元を 6,1 アイテムの長さを 3 とし、アイテムの数は一様分布を用いて 2-6 のいずれかに決定する。

タスクはクエリアイテムをメモリから検索する、入力ベクトルを忠実に復元するといったメモリネットワークの基礎的な能力を要求する。加えて、入力アイテムの順序の情報を保存する能力も要求する。

4.2.2 priority sort

入力系列はランダムなバイナリベクトルに、[-1,1] の範囲から一様分布に従い決定した乱数を優先度として付加したものからなる。ターゲットは入力系列をこの優先度に従ってソートした系列の一部とする。NTM[] 中の設定に従い入力系列の長さは 20 ベクトル, ターゲットは系列の中から優先度が高い順に 16 ベクトルとする。

このタスクはネットワークが入力をソートする能力を評価する。

4.2.3 実験結果

表 4.1 に

4.3 実験2　priority sort における長期入力系列の検証

4.2 節のタスクにおいて入力系列のサイズを増加させたときの精度の推移を観察する。この実験により提案ネットワークが長期依存関係を扱う能力を評価する。提案ネットワークは入力系列が長期にわたるとき、分散型項目メモリを持つ SAM よりも正答率が高くなることが期待される。

4.4 実験3　Nth-farthest

この実験の目的は提案手法が NTM に関係推論能力を付与出来るかを評価することである。既存研究 [R-RNN][SAM] は RMC や SAM のような関係推論能力を持つモデルが 90 以上の精度を記録した一方で、NTM や DNC の精度は 30 を超えないことを示している。提案ネットワークが NTM メモリ項目間の関係を計算できる場合、NTM よりも大きく改善された精度を示すことが期待される。

入力系列はランダムに生成されたバイナリベクトルからなる。ベクトルの次元を d、系列長を l で表したとき、タスクの要求は入力系列中のあるベクトル m から n 番目に遠いベクトルを見つける事である。m, n は入力系列ごとにランダムに決定される。1 ステップあたりの入力はバイナリベクトル、ベクトルの ID、m、n を連結したベクトルからなり、$ID, m, n \in \{1, 2, \cdots, l\}$ は one-hot エンコーディングにより表現されるため、最終的な入力の次元は $d + 3l$ になる。既存研究 [R-RNN] に従い、$d = 16, l = 8$ として実験を行った。

このタスクは入力の読み書きやソートといったタスクよりも複雑な処理をネットワークに要求する。ネットワークは m と全入力のペアの距離を計算しソートを行う必要がある。距離は入力間の関係情報の一形態である。入力そのものをソートするタスクと異なり、関係情報のソートの為に関係メモリを活用する必要がある。

4.5 実験4 babi dataset

表 4.1: あああといいいの予測誤差

	2019		2018		2017	
モデル	ああ	いい	ああ	いい	ああ	いい
Naive	**1**	1	**1**	1	**1**	1
TCN	1.0895	0.9032	1.4791	**0.9198**	1.2888	0.8555
LSTM	1.0384	0.9295	1.4917	0.9725	1.1627	0.8541
提案手法	1.0977	**0.8698**	1.3824	0.9439	1.2061	**0.8516**

第5章

結論

本論文では

記法メモ

Wavenet[1] は音声波形を時系列データとして自己回帰モデルで学習することによって，人間の声のような自然な音声を生成することができる．時点 t における観測値を x_t，$\boldsymbol{x} = \{x_1, ..., x_T\}$ を観測値の全体集合とする．このとき，波形の同時確率は 条件付き確率の積として以下のよう表現される．

$$p(\boldsymbol{x}) = \prod_{t=1}^{T} p(x_t | x_1, ..., x_{t-1}) \tag{5.1}$$

つまり，x_t は前時点の全てにおけるサンプルに条件づけられる．

図 5.1 に因果的畳み込み層のスタックを示す．

[2] は．

今後の課題を以下に挙げる．

- の向上

  必要がある．

- への応用

  を行いたい．

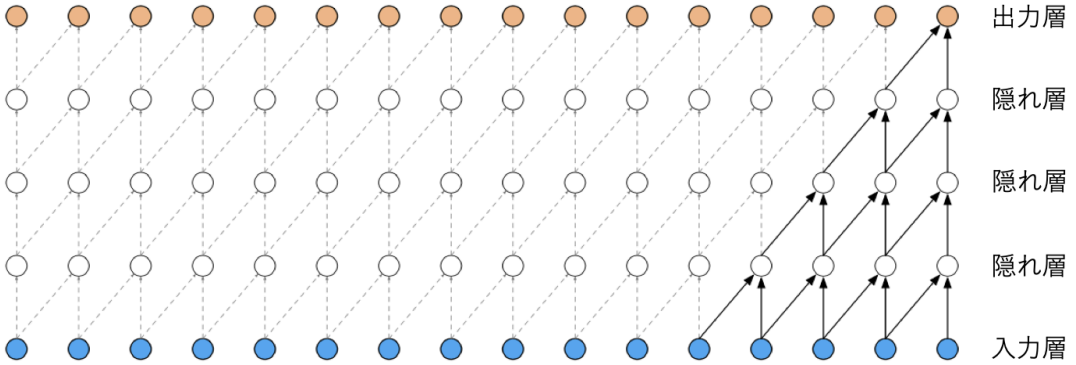

図 5.1: 因果的畳み込み層

- の改善

  今後，取り組みたい．

謝辞

本研究を行うにあたり親身に相談に乗っていただき，ご指導してくださった萩原将文教授，ならびに共に問題解決，議論，相談に付き合ってくださった研究室の先輩方，同期の皆様に深く感謝いたします．誠にありがとうございました．

参考文献

[1] Aaron van den Oord et al. “Wavenet: A generative model for raw audio”. In: *arXiv preprint arXiv:1609.03499* (2016).

[2] Biing Hwang Juang and Laurence R Rabiner. “Hidden Markov models for speech recognition”. In: *Technometrics* 33.3 (1991), pp. 251–272.

付録 A

データの前処理

A.1　ヒストリカルデータ

為

付録Ｂ

実験結果詳細

B.1　の予測

第

付録C

のモデル化

C.1　異なる

あ

付録D

のヒストグラム

D.1　異なる期間

図