One Model To Learn Them All

- Łukasz Kaiser (Google Brain)
- Aidan N. Gomez (University of Toronto)
- Noam Shazeer (Google Brain)
- Ashish Vaswani (Google Brain)
- Niki Parmar (Google Research)
- Llion Jones (Google Research)
- Jakob Uszkoreit (Google Research)

Abstract

- ディープラーニングは音声認識や画像分類、翻訳などの多くの分野で素晴らしい結果を残している。
- しかしながら、それぞれの問題に対するディープなモデルを得るためには、構造の調査と長時間のチューニングが必要である。
- そこで、多くの問題に対して良い結果を出すための 1 つのモデルを提案する。
- このモデルは ImageNet、多言語翻訳、画像キャプション、音声認識、構文解析の訓練をしている。
- 多数の領域からブロックを組み込んだ構造になっていて、畳み込み層と attention 機構、スパースな層を含む。
- 計算ブロックのそれぞれは、訓練したタスクのサブセットに対して重要になる。
- 面白いことに、あるタスクに対して重要でないブロックを加えてもパフォーマンスの低下はなく、多くの場合で全てのタスクへの精度が向上した。
- さらに、少ないデータしかないタスクでも他のタスクと組み合わせて訓練することで、大きな恩恵が得られることを示した。（多少精度は劣化する）

1. Introduction

近年の深層ニューラルネットワークの成功は、コンピュータビジョン[13]から音声認識[8]などの多くの分野に及んでいる。畳み込みネットワークは、視覚に関連するタスクに優れていますが、自然言語処理タスク、例えば機械翻訳では、リカレントニューラルネットワークが成功していることが証明されています[27,3,4]。しかし、いずれの場合も、ネットワークは設計されており、手元の問題に特化して調整されています。これは、この新しい努力のたびにこの努力を繰り返す必要があるため、深い学習の影響を制限します。また、人間の脳の一般的な性質とは非常に異なります。人間の脳は、多くの異なるタスクを学び、伝達学習の恩恵を受けることができます。自然な問題が発生します：

複数のドメイン間でタスクを解決するための統一された深い学習モデルを作成できますか？

マルチタスクモデルに関する質問は、深い学習文献の多くの論文で研究されています。自然言語処理モデルは、以前からマルチタスク手法の恩恵を受けることが示されており[6]、最近、機械翻訳モデルは複数のランゲージ[18]で訓練されたときにゼロショット学習を示すようにさえ示されている。音声認識は、顔の目印の検出[31]のようないくつかの視覚問題を有するので、マルチタスク訓練[24]の恩恵を受けることが示されている。しかし、これらのモデルはすべて、同じドメインの他

のタスクで訓練されています。翻訳タスクは、他の翻訳タスク、他のビジョンタスクを伴うビジョンタスク、他のスピーチタスクを伴うスピーチタスクで訓練されます。マルチモーダル学習は、教師なし設定[22]と先験的な無関係なタスク[22]として使用されるときの学習表現を改善することが示されている。しかし、競争力のあるマルチタスクマルチモーダルモデルは提案されていないので、上記の質問は未解決のままである。

この作業では、さまざまなドメインから複数のタスクを同時に学習できる単一の深い学習モデルであるマルチモデルアーキテクチャを導入することで、上記の質問に積極的に答えていくステップを踏み出します。具体的には、次の8つのコーパスで同時にマルチモデルを訓練します。

（1）WSJ音声コーパス[7]

（2）ImageNetデータセット[23]

（3）COCO画像キャプションデータセット[14]

（4）WSJ解析データセット[17]

（5）WMT英語 - ドイツ語翻訳コーパス

（6）上記の逆：ドイツ語 - 英語の翻訳。

（7）WMT英語 - フランス語翻訳コーパス

（8）上記の逆：ドイツ語 - フランス語の翻訳。

モデルは、上記のすべてのタスクを学習し、優れたパフォーマンスを達成します。現時点では最先端ではありませんが、近年研究された多くのタスク特有のモデルを上回ります（詳細はセクション3を参照してください）。図1は、モデルから直接取り出したデコードを示しています。画像にキャプションを付け、分類し、フランス語とドイツ語に翻訳し、構文解析ツリーを構築できることは明らかです。マルチモデルは最初のステップに過ぎず、将来的には調整され、改善されるでしょうが、2つの重要な洞察はそれを完全に機能させるために不可欠であり、この作業の主な貢献です。

小さなモダリティ固有のサブネットワークは統一された表現に変換され、そこから戻ってきます。

画像、音波、テキストなど、大きさや大きさが大きく異なる入力データを訓練できるようにするには、入力を共同表現空間に変換するサブネットワークが必要です。これらのサブネットワークは、各モダリティ（画像、音声、テキスト）に固有のものであり、これらの外部ドメインと統一された表現の間の変換を定義するものです。我々はモダリティネットを計算上最小限に抑えるように設計し、重いフィーチャ抽出を促進し、大部分の計算がモデルのドメイン非依存体内で実行されるようにします。我々のモデルは自己回帰的であるため、モダリティネットは入力を統一表現に変換し、後でこの表現から出力空間に変換する必要があります。 2つの設計上の決定が重要でした。

- *統一表現は可変サイズです。*固定サイズの表現は魅力的で実装が簡単ですが、ボトルネックが発生し、モデルのパフォーマンスが制限されます。
- *同じドメインの異なるタスクは、モダリティネットを共有します。*すべてのタスクにサブネットワークを作成することは避け、すべての入力モダリティに対してサブネットワークを作成することを優先します。たとえば、すべての翻訳タスクは、どの言語ペアに関係なく、同じモダリティ - ネット（およびボキャブラリ）を共有します。これにより、タスク間の一般化が促進され、新しいタスクを即座に追加することができます。

異なる種類の計算ブロックは、さまざまな問題の良い結果を得るために重要です。

MultiModelの本体には、複数のドメインからのビルディングブロックが組み込まれています。我々は、深さ方向に分離可能な畳み込み、注目メカニズム、および希薄にゲートされた混合エキスパート層を使用する。これらのブロックは、異なるドメインに属し、他のドメインのタスクでこれまでには研究されていなかった論文に導入されました。例えば、分離可能な畳み込みはXceptionアーキテクチャ[5]に導入され、以前はテキスト処理や音声処理には適用されていなかった。一方、疎結合の専門家の混合物[21]は言語処理タスクのために導入されており、画像の問題については研究されていない。これらのメカニズムのそれぞれは、導入されたドメインにとって本当に重要であることがわかりました。たとえば、画像関連のものよりも言語関連のタスクでは、はるかに重要です。しかし興味深いことに、これらの計算ブロックを追加することは、設計されていないタスクであっても、パフォーマンスを損なうことはありません。実際には、注目度とエキスパート層の両方が、ImageNet上のMultiModelのパフォーマンスを若干向上させることがわかりました。

2. MultiModel Architecture

マルチモデルは、図2に示すように、いくつかの小型モダリティネット、エンコーダ、I / Oミキサ、自己回帰デコーダで構成されています。すでに述べたように、エンコーダとデコーダは、3つの重要な計算ブロックさまざまな問題を横断して：

（1）畳み込みは、モデルが局所パターンを検出し、空間全体にわたって一般化することを可能にする。

（2）注意層は、モデルの性能を向上させるために特定の要素に集中することを可能にする。

（3）疎結合の専門家の混合物は、過剰な計算コストをかけずにモデル容量を与える。

まず、これらの3つのブロックのそれぞれのアーキテクチャを記述し、次に、エンコーダ、デコーダ、およびモダリティネットのアーキテクチャを紹介します。

2.1 Convolutional Blocks

ローカル計算を実行するために、ReLUの非線形性と正規化を伴う畳み込みのブロックを使用します。畳み込みのブロックは、テンソルの形状[バッチサイズ、シーケンス長、フィーチャチャンネル]を入力として取得し、次のように処理された同じ形状のテンソルを返します。

畳み込み演算では、[15]と同様の方法で[5]で導入された深度分離可能な畳み込みを使用します。深度分離可能な畳み込みは、伝統的な畳み込みのパラメータおよび計算上効率的な変形である。それらは、各フィーチャチャネル上のコンボリューションによって個別に定義され、その後、所望のフィーチャ深さに投影するためのポイントワイドコンボリューションが定義されます。完全な定義については、読者に[5]を参照されたい。ここでは、ストライドsで入力テンソルxに適用され、因子d（[30]参照）で拡張されたサイズh×wのfカーネルに対応する重みW h×w を持つ深度分離可能な畳み込みをSepConvd、s、f（W、x）である。拡張dまたはストライドsが1に等しいか、または出力サイズfが入力のフィーチャ深度に等しい場合、ストライド、拡張および出力サイズの添え字は省略されることに注意してください。

`入力`のReLUアクティベーション、続いてSepConv、それに続くレイヤー正規化の3つのコンポーネントで構成されるブロックで、畳み込みを使用します。層正規化[2]は、下の層のh個の隠れた単位に対して作用し、各バッチの例について層ごとの統計を計算し、それに応じて正規化する。これらの正規化されたユニットは、スカラー学習されたパラメータGおよびBによってそれぞれスケーリングされ、シフトされ、非線形性によって活性化される最終ユニットを生成する。したがって、完全畳み込みステップは、以下のように定義される。

$$ConvStep_{d,s,f}\left(W,x\right)=LN\left(SepConv_{d,s,f}\left(W,ReLU\left(x\right)\right)\right)$$

畳み込みステップは、図3に示すように、それらを積み重ね、残りの接続を追加することによってブロックに構成されます。スタック入力と第2および第4の畳み込みステップの出力との間に2つのスキップ接続を持つ4つの畳み込みブロックのスタックを使用し、最初の2つは3×1のカーネルを有し、次の2つは15×1のカーネルを有し、最後のカーネルは8で拡張されて広い受容野を提供する。また、各ブロックの最後に40％のドロップアウトを追加するので、完全なブロックは次のように定義されます。

$$\begin{aligned}
hidden1\left(x\right)&=ConvStep\left(W_{h1}^{3\times1},x\right)\\
hidden2\left(x\right)&=x+ConvStep\left(W_{h2}^{3\times1},hidden1\left(x\right)\right)\\
hidden3\left(x\right)&=ConvStep\left(W_{h3}^{15\times1},hidden2\left(x\right)\right)\\
hidden4\left(x\right)&=ConvStep_{d=8}\left(W_{h4}^{15\times1},hidden3(x)\right)\\
ConvBlock\left(x\right)&=\begin{cases}Dropout\left(hidden4\left(x\right),0.4\right) & \text{during training}\\ hidden4\left(x\right) & \text{otherwise}\end{cases}
\end{aligned}$$

2.2 Attention Blocks

注意のため、図3に示すように、[3]に似た[1]に似たマルチヘッドドットプロダクトアテンションメカニズムを使用します。アテンションレイヤへの入力は、 *ソース*テンソルと *ターゲット*テンソルの2つのテンソルです形状[バッチサイズ、シーケンス長、フィーチャチャネル]を持つ *ターゲット*テンソルは、タイミング信号で加算的に構成され、2つの畳み込みブロックを使用して混合されます。この混合されたテンソルは、図3に示すように、各ヘッドを表すg = 8の別々のテンソルに分割された入力を持つドット積の注意であるマルチヘッドドット積の注意を使用して自立する。この注意メカニズムと以前に使用されたものとの主な違い。彼らは、コンテンツベースの注意を自分の位置に基づいて集中させることができます。これらは、正弦曲線と余弦曲線を連結して作成されます。

$$\begin{aligned}
\Delta\left(2d\right)&=1e4^{-\frac{2d}{depth}}\\
timing\left(t,\left[2d,2d+1\right]\right)&=\left[\sin(t\Delta\left(2d\right))\left|\right|_2\cos(t\Delta\left(2d\right))\right]
\end{aligned}$$

ここで[a || db]はd次元に沿ったaとbの連結を表す。 *ソース*テンソルは、最終的に2つの異なる点畳み込みを通ってメモリキーKおよび値Vを生成し、クエリキー、メモリキーおよびメモリ値は、自己介入 *ターゲット*と *ソース*との間のアテンションメカニズムを適用するために使用される（図3 ）。

2.3 Mixture-of-Experts Blocks

エキスパートの混合層は、いくつかの簡単なフィードフォワードニューラルネットワーク（専門家）と訓練可能なゲーティングネットワークで構成されています各入力を処理する専門家の疎結合。ここで説明したアーキテクチャーを正確に使用するため、詳細については読者に[21]を参照してください。特に、トレーニング中に、専門家プール全体からk = 4のエキスパートを選択し、[21]のように追加のロードバランシングコストを追加します。私たちのモデルの2つの専門家の混合層のそれぞれでは、8つの問題を共同して訓練する際に240人の専門家のプールを使用し、それぞれの問題を別々に訓練する場合は60人の専門家を使用します。

2.4 Encoder and Mixer and Decoder

マルチモデルの本体は、入力を処理するだけのエンコーダ、符号化された入力を以前の出力と混合するミキサ（自己回帰部）、および入力と混合を処理して新しい出力を生成するデコーダの3つの部分から構成されます。

エンコーダ、ミキサおよびデコーダは、ByteNet [11]またはWaveNet [29]などのシーケンスモデルに対して以前の完全畳み込みシーケンスと同様に構成されていますが、使用される計算ブロックは異なります。図3にそのアーキテクチャーを示します。そこに見られるように、エンコーダーは、中間に6つの反復畳み込みブロック（前に説明した）とエキスパートの混合レイヤーで構成されています。ミキサーは、アテンションブロックと2つの畳み込みブロックで構成されています。デコーダは4ブロックの畳み込みと注意で構成され、中間にエキスパート層が混在しています。重要なのは、ミキサーとデコーダの畳み込みは左に埋め込まれているため、将来は情報にアクセスできないということです。これにより、モデルは自己回帰的になり、この畳み込み自己回帰生成スキームは、長期依存性を確立することができる入力および過去出力に対して大きな受容野を提供する。

デコーダが同じモダリティであっても異なるタスクの出力を生成できるように、*To-English*や*To-Parse-Tree*などのコマンドトークンでデコードを開始します。訓練中に各トークンに対応する埋め込みベクトルを学習する。

2.5 Modality Nets

私たちは、言語（テキストデータ）、画像、オーディオ、およびカテゴリデータのための4つのモダリティネットを持っています。

2.5.1 Language modality net

私たちの言語ベースのデータはすべて、[25]の方法に従って、8kサブワード単位で同じ語彙を使用してトークン化されます。言語入力モダリティは、終了トークンで終わる一連のトークンを取ります。この一連のトークンは、学習された埋め込みを使用してボディの正しい次元にマッピングされます。出力側では、言語モダリティはボディのデコードされた出力を取り、学習された線形マッピングを実行し、続いてSoftmaxを実行し、トークンボキャブラリ上の確率分布をもたらす。

$$
\begin{aligned}
LanguageModality_{in}\left(x, W_E\right) &= W_E \cdot x \\
LanguageModality_{out}\left(x, W_S\right) &= Softmax(W_S \cdot x)
\end{aligned}
$$

2.5.2 Image modality net

画像入力モダリティは、Xception入力フロー[5]に類似しています。入力画像の特徴の深度は、ConvResと呼ばれる残余畳み込みブロックを使用して徐々に深くなり、次のように定義されます。

$$
\begin{aligned}
c1(x, F) &= ConvStep_{f=F}(W^{3\times3}, x) \\
c2\left(x, F\right) &= ConvStep_{f=F}\left(W^{3\times3}, c1\left(x, F\right)\right) \\
p1\left(x, F\right) &= MaxPool_2\left(\left[3 \times 3\right], c2\left(x, F\right)\right) \\
ConvRes\left(x, F\right) &= p1\left(x, F\right) + ConvStep_{s=2}\left(W^{1\times1}, x\right)
\end{aligned}
$$

ここで、MaxPools（[h×w]、x）は、ストライドsおよびウィンドウ形状[h×w]を有するx上の最大プール層である。ネットワーク深度d（d = 1024を使用）を持つImageModality入力フローは、次のように定義されます。

$$
\begin{aligned}
h1\left(x\right) &= ConvStep_{s=2,f=32}\left(W^{3\times3}, x\right) \\
h2\left(x\right) &= ConvStep_{f=64}\left(W^{3\times3}, h1\left(x\right)\right) \\
r1\left(x\right) &= ConvRes\left(h2\left(x\right), 128\right) \\
r2\left(x\right) &= ConvRes\left(r1\left(x\right), 256\right) \\
ImageModality_{in}\left(x\right) &= ConvRes\left(r2\left(x\right), d\right)
\end{aligned}
$$

2.5.3 Categorical modality net

カテゴリ別出力モダリティは、Xception出口フロー[5]に類似しています。ネットワーク入力が画像またはスペクトルオーディオデータのような2次元データである場合、モデル本体からの1次元出力は、最初に2次元に再成形され、続いてプログレッシブダウンサンプリングが行われる。

$$
\begin{aligned}
skip(x) &= ConvStep_{S=2}(W^{3\times3}_{skip}, x) \\
h1(x) &= ConvStep(W^{3\times3}_{h1}, x) \\
h2(x) &= ConvStep(W^{3\times3}_{h2}, h1(x)) \\
h3(x) &= skip(x) + MaxPool_2([3\times3], h2(x)) \\
h4(x) &= ConvStep_{f=1536}(W^{3\times3}_{h4}, h3(x)) \\
h5(x) &= ConvStep_{f=2048}(W^{3\times3}, h4(x)) \\
h6(x) &= GlobalAvgPool(ReLU(h5(x))) \\
CategoricalModality_{out}(x) &= PointwiseConv(W^{classes}, h6(x))
\end{aligned}
$$

GlobalAvgPolは、すべての空間的および時間的次元にわたって取られた平均を表す。

2.5.4 Audio modality net

時間の経過とともに、または2次元スペクトログラムとして、1次元波形の形でオーディオ入力を受け入れます。波形入力とスペクトル入力の両方のモダリティでは、ImageInputModality（セクション2.5.2）の8つのConvResブロックのスタックを使用します。 i番目のブロックの形式はli = ConvRes（li-1,2i）です。スペクトルモダリティは周波数ビン次元に沿ってストライドを実行せず、スペクトル領域で完全な解像度を維持します。

2.6 Related Models

マルチモデル・アーキテクチャは、ニューラル・マシン変換に適用されるイヤリング・エンコーダ・デコーダ・アーキテクチャから引き出される。初期のシーケンス - シーケンス変換モデル[27,3,4]は、長い短期記憶セルを有するリカレントニューラルネットワーク（RNNs）を使用した[9]。畳み込みアーキテクチャは、[10]以降の[19]から始まる単語レベルの神経機械翻訳において良好な結果をもたらした。これらの初期のモデルは、畳み込みの上に標準のRNNを使用して出力を生成し、特に長い文章では、RNNシーケンス - シーケンスモデルのようにパフォーマンスを傷つけるボトルネックがありました[27,4]。このボトルネックのない完全畳み込み神経機械翻訳は、[16、11]に示されている。 [16]（Extended Neural GPU）のモデルは、ゲートされた畳み込みレイヤーの反復スタックを使用していましたが、[11]（ByteNet）のモデルは再帰を伴わず、デコーダで左パッド付き畳み込みを使用しました。 WaveNet [29]で導入され、MultiModelでも使用されているこのアイデアは、効率を大幅に向上させます。深さ方向に分離可能な畳み込みは、Sifre [26]によって最初に研究され、その後、Xceptionを用いて大規模画像分類で良好な結果を得るために用いられた[5]。

3. Experiments

TensorFlowを使用して前述のマルチモデルアーキテクチャを実装し、さまざまな構成でトレーニングしました。下で報告されたすべてのトレーニングでは、同じハイパーパラメータセットとAdamオプティマイザ[12]をグラデーションクリッピングで使用しました。実装をリリースします

私たちのセットアップの詳細とすべての使用されたハイパーパラメータと一緒にオープンソースとして。私たちは次の質問に答えるために実験に焦点を当てました：

（1）マルチモデルは、最先端の結果から8つのタスクで同時にどのくらい訓練されていますか？
（2）8つのタスクのトレーニングは、それぞれのタスクのトレーニングとはどのようにして同時に個別に比較されますか？

（3）上記の異なる計算ブロックは、どのように異なるタスクに影響を与えますか？

上記の質問に答えて、私たちは常に8つの問題すべてを考慮するとは限りません。特に4つの翻訳問題は非常によく似た動作をするため、それぞれの比較ですべての問題を含めることにしましたが、さまざまな問題に焦点を当てました。

質問（1）に答えるために、8つの問題のマルチモデルのパフォーマンスを表1の最新の結果と比較します。マルチモデルのハイパーパラメータのチューニングにまだ多くの時間を費やしていないため、そこに見られる違いは、より多くのチューニングでより小さくなるでしょう。達成された結果は、昨年報告されたExtended Neural GPUの結果で改善した英語 - フランス語の翻訳など、タスク固有のモデルが重いチューニングなしで得られる結果と似ています[16]。

質問（2）に答えるために、我々は訓練されたマルチモデルと、単一のタスク上で別々に訓練されたマルチモデルとを比較する。 8つのタスクを共同して訓練するとき、我々はモデルの共有パラメータを用いて各タスクについて別々の訓練を受けました。 1つのタスクを訓練する際には、この作業で同じような手順で1つの訓練だけを使用しました。同じモデルの異なるインスタンス化を比較しているので、負の対数不一致とトークン単位の精度（開発セットで測定）の2つの内部メトリックを報告します。表2の結果からわかるように、ジョイント8-問題モデルは、大きなタスクでは単一モデルと同様に機能し、構文解析などのデータが利用できないタスクではより優れていることがあります。

表2に示されている構文解析の大幅な改善は、翻訳タスクでの多数のテキストデータを考慮すると驚くべきことではありません。しかし、一見無関係なタスクであるImageNetだけで解析を訓練することで改善が得られるのではないかと思っていました。表3に見られるように、これは事実です。パフォーマンスの違いは重要であり、ドロップアウトと早期停止の両方を使用するため、オーバーフィッティングには関係しないと推測します。むしろ、ImageNetと解析のような一見無関係のタスク間でさえ、いくつかの転送学習を可能にする異なるタスク間で共有される計算プリミティブがあるようです。

質問（3）に答えるために、エキスパート混合層を持たないトレーニングや注意メカニズムを持たないトレーニングがどのようにしてさまざまな問題のパフォーマンスに影響を与えるかを確認します。これらのメカニズムは機械翻訳を念頭に置いて設計されているため、英語 - フランス語の翻訳を確認します。しかし、ImageNetも含まれています。これは、これらのブロックの恩恵を最小限に抑えるという問題です。実際、これらのブロックを削除すると、この作業で本当に役に立たなかった場合、ImageNet単独のパフォーマンスが向上することが期待できます。対照的に、表4では、これらのブロックがパフォーマンスに影響を与えないか、またはパフォーマンスをわずかに向上させることを示しています。これは、異なる計算ブロックを混合することが、実際には多くのさまざまなタスクのパフォーマンスを向上させる良い方法であると結論づけます。

4. Conclusions

初めて、深い単一の学習モデルが、複数のドメインから多数の大規模なタスクを共同で学習できることを初めて実証します。成功への鍵は、可能な限り多くのパラメータを共有し、異なるドメインの計算ブロックを一緒に使用するマルチモーダルアーキテクチャを設計することにあります。私たちは、このモデルが、大量の利用可能なデータを持つタスクからデータが限られているタスクへの移転学習を示しているため、これはより一般的な深い学習アーキテクチャの面白い将来の作業への道を切り開くものと確信しています。

References

[1] Niki Parmar Jakob Uszkoreit Llion Jones Aidan N. Gomez Lukasz Kaiser Illia PolosukhinAshish Vaswani, Noam Shazeer. Attention is all you need. arXiv preprint arXiv:1706.03762,2017.

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. arXiv preprintarXiv:1607.06450, 2016.

[3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointlylearning to align and translate. CoRR, abs/1409.0473, 2014. URL http://arxiv.org/abs/1409.0473.

[4] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Fethi Bougares, Holger Schwenk,and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statisticalmachine translation. CoRR, abs/1406.1078, 2014. URL http://arxiv.org/abs/1406.1078.

[5] Francois Chollet. Xception: Deep learning with depthwise separable convolutions. arXivpreprint arXiv:1610.02357, 2016.

[6] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: deepneural networks with multitask learning. In Proceedings of the 25th International Conferenceon Machine learning, pages 160–167, 2008.

[7] Linguistic Data Consortium et al. Csr-ii (wsj1) complete. Linguistic Data Consortium, Philadelphia, vol. LDC94S13A, 1994.

[8] George E. Dahl, Dong Yu, Li Deng, and Alex Acero. Context-dependent pre-trained deepneural networks for large-vocabulary speech recognition. IEEE Transactions on Audio, Speech& Language Processing, 20(1):30–42, 2012.

[9] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.

[10] Nal Kalchbrenner and Phil Blunsom. Recurrent continuous translation models. InProceedings EMNLP 2013, pages 1700–1709, 2013. URL http://nal.co/papers/KalchbrennerBlunsom_EMNLP13.

[11] Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Ko-ray Kavukcuoglu. Neural machine translation in linear time. arXiv preprint arXiv:1610.10099,2016.

[12] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. CoRR,abs/1412.6980, 2014. URL http://arxiv.org/abs/1412.6980.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. Imagenet classification with deepconvolutional neural network. In Advances in Neural Information Processing Systems, 2012.

[14] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, JamesHays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO:common objects in context. CoRR, abs/1405.0312, 2014. URL http://arxiv.org/abs/1405.0312.

[15] Francois Chollet Łukasz Kaiser, Aidan N. Gomez. Depthwise separable convolutions for neuralmachine translation. arXiv preprint arXiv:1706.03059, 2017.

[16] Samy Bengio Łukasz Kaiser. Can active memory replace attention? In Advances in NeuralInformation Processing Systems, (NIPS), 2016.

[17] Mitchell P Marcus, Beatrice Santorini, Mary Ann Marcinkiewicz, and Ann Taylor. Treebank-3ldc99t42. CD-ROM. Philadelphia, Penn.: Linguistic Data Consortium, 1999.

[18] Quoc V. Le Maxim Krikun Yonghui Wu Zhifeng Chen Nikhil Thorat Fernanda Viégas MartinWattenberg Greg Corrado Macduff Hughes Jeffrey Dean Melvin Johnson, Mike Schuster.Google's multilingual neural machine translation system: Enabling zero-shot translation. arXivpreprint arXiv:1611.04558, 2016.

[19] Fandong Meng, Zhengdong Lu, Mingxuan Wang, Hang Li, Wenbin Jiang, and Qun Liu.Encoding source language with convolutional neural network for machine translation. In ACL,pages 20–30, 2015.

[20] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng.Multimodal deep learning. In Proceedings of ICML'11, pages 689–696, 2011.

[21] Krzysztof Maziarz Andy Davis Quoc Le Geoffrey Hinton Jeff Dean Noam Shazeer, Aza-lia Mirhoseini. Outrageously large neural networks: The sparsely-gated mixture-of-expertslayer. arXiv preprint 1701.06538, 2017.

[22] Bernardino Romera-Paredes, Andreas Argyriou, Nadia Berthouze, and Massimiliano Pontil.Exploiting unrelated tasks in multi-task learning. In JMLR Proceedings of AISTATS'12, pages951–959, 2012.

[23] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, ZhihengHuang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei.ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision(IJCV), 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[24] Michael L. Seltzer and Jasha Droppo. Multi-task learning in deep neural networks for improvedphoneme recognition. In Proceedings of the IEEE International Conference on Acoustics,Speech and Signal Processing (ICASSP'13), 2013.

[25] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare wordswith subword units. CoRR, 2015.

[26] Laurent Sifre and Stéphane Mallat. Rotation, scaling and deformation invariant scattering fortexture discrimination. In 2013 IEEE Conference on Computer Vision and Pattern Recognition,Portland, OR, USA, June 23-28, 2013, pages 1233–1240, 2013.

[27] Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. Sequence to sequence learning with neuralnetworks. In Advances in Neural Information Processing Systems, pages 3104–3112, 2014.URL http://arxiv.org/abs/1409.3215.

[28] Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke. Inception-v4, inception-resnet andthe impact of residual connections on learning. CoRR, abs/1602.07261, 2016.

[29] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, AlexGraves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generativemodel for raw audio. CoRR abs/1609.03499, 2016.

[30] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. arXivpreprint arXiv:1511.07122, 2015.

[31] Loy C.C. Tang X. Zhang Z., Luo P. Facial landmark detection by deep multi-task learning. InProceedings of ECCV'14, 2014.