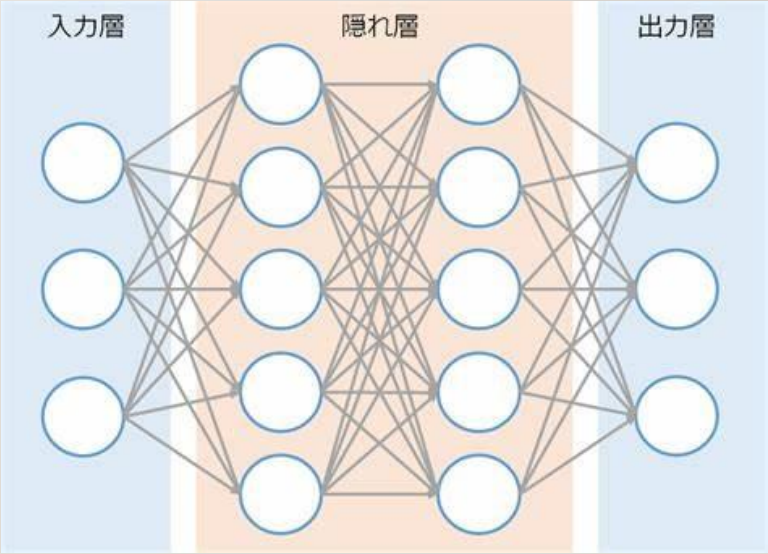
ZCOMP：ベクトル拡張を用いた深層ニューラルネットワークの交差層のメモリフットプリントの削減

飯沼柾斗　2290170008-0　ihpc B4 (M0)

深層ニューラルネットワーク(DNN)

- 深層ニューラルネットワークは、画像認識、機械学習、自然言語処理、音声認識のアプリケーションの方法として流行してきている
- 目的に応じて多くの種類があり、それぞれが特定のアプリケーションに適している



DNNに適したハードウェア

CPU	GPU	FPGA
・広く用いられており、新規導入の必要性が低くコストパフォーマンスが良い ・柔軟性が高く非DNNタスクと統合しやすい	・計算スループットが大きく、演算が高速なので膨大な計算をするDNNと相性が良い ・高価	・DNN分野全体からするとニッチ ・低消費電力、低コストなことが多く、モバイル分野に強い

DNNによるハードウェアの負荷

- クラス分けの精度を上げるために、より多くのネットワーク層とトレーニングパラメータが用いられるようになった
- 性能が上がるにつれて、ハードウェアに対する負荷が増加してきた

メモリ面

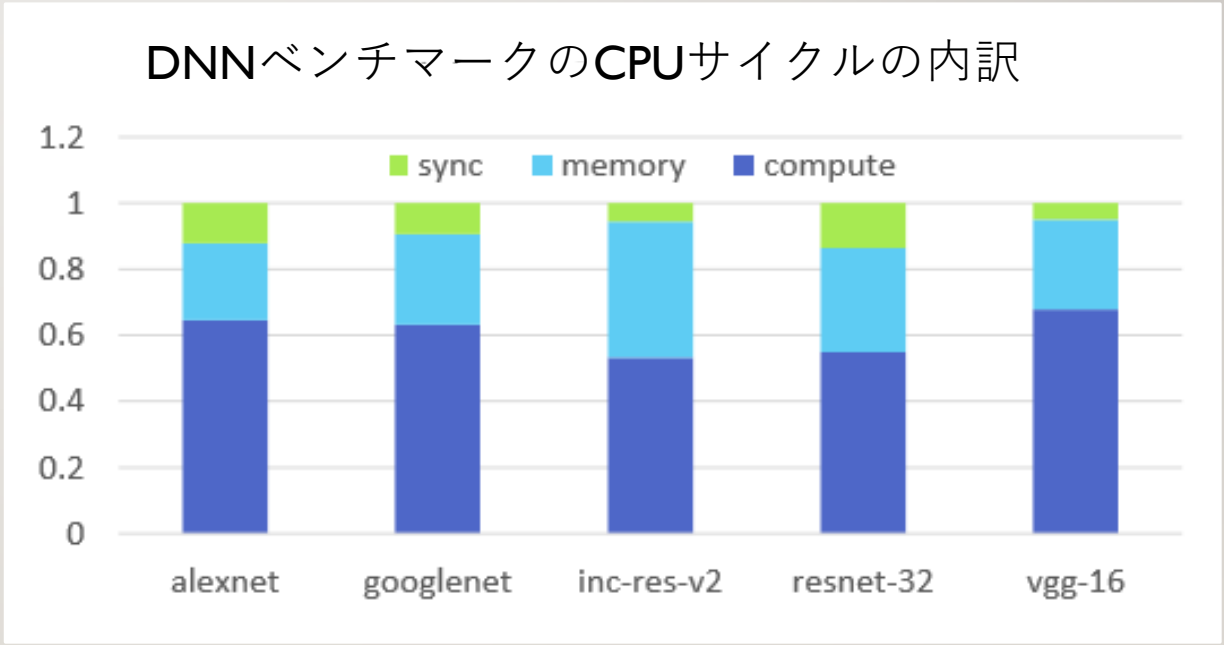
ネットワーク層で通信される大規模な中間データにより、メモリの容量と帯域幅を圧迫してしまう

演算量

レイヤ、カーネルの増加により、演算量が膨大になってしまう

DNNによるハードウェア負荷の割合

- DNNワークロードの実行時間の大部分(24%～41%)がメモリアクセスによるストールに費やされている
- 今後、計算量は改善していく傾向にある一方、バッチサイズと活性化マップは大きくなる
- DNNに合わせたメモリの最適化が必要とされてくる



メモリ階層への負荷の削減

先行研究では、カスタマイズされたデータパスを介して、クロスレイヤのデータの分散性を活用してメモリ圧迫の軽減を実現した

↓ しかし

汎用マルチプロセッサではデータを動的に圧縮、拡張することが困難

DNNのデータの偏り (Sparsity of DNN Data)

Model Sparsity

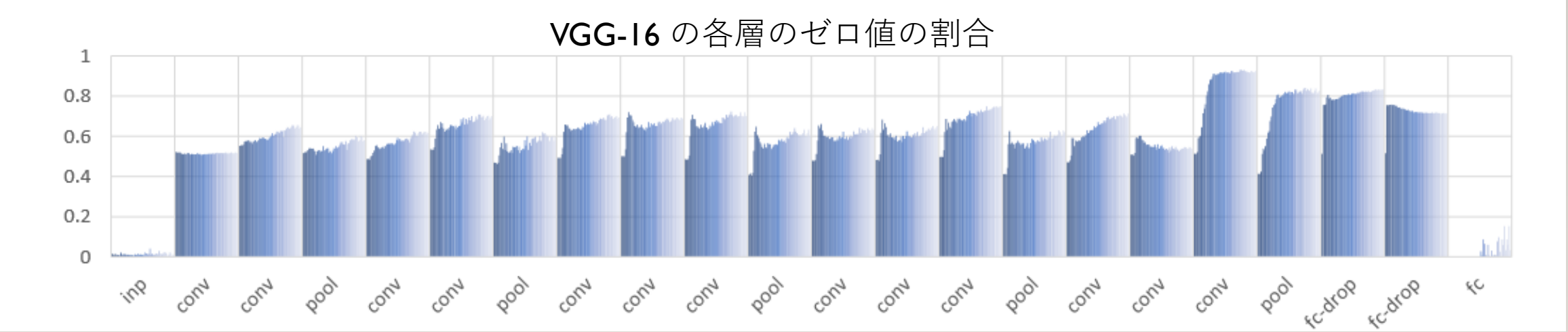
- 冗長性、ゼロ値、重みの高精度化で発生
- 圧縮は訓練されたモデルを入力としてオフラインで解析、処理、圧縮され、より小さなモデルを実現する

Feature Map Sparsity

- Reluなどの活性化関数によりゼロ値が大量にできる。またはアルゴリズムの都合で出力の一部が破棄されゼロになることで発生
- 訓練、推論のどちらに置いても入力に一意に、動的に生まれるため、オフライン処理が困難

各層でのデータの偏り

- すべてのネットワーク層で特徴マップのデータの偏り(ゼロ)が存在することがわかる
- プーリング層では入力データの偏りが減少しますが、畳み込み層ではほとんどの場合、偏りが増加する



中間データの特徴

DNNの中間データは順次ストリーム化されるか、

レイヤ間で規則的にリシェイプされるか

中間データは逐次圧縮/拡張することが可能

DNNクロスレイヤ通信用CPUベクトルISA拡張機能
ZCOMPを提案

ZCOMP概要

- ゼロ値圧縮/展開をコンパクトに表現し、メタデータの生成、保存、検索を完全に自動化することで、複数の余分な命令実行レジスタの使用が不要になる
- メモリに書き込まれる前のクロスレイヤーデータを動的に圧縮/拡張するために、推論と訓練の両方を対象とすることができる
- 特徴マップの圧縮率は動的に変化するので、レイヤに通すまでメモリ空間が割り当てられない
- ZCOMPは仮想メモリフットプリントの削減を目的としておらず、可能であれば、元の仮想メモリ割り当てを変更することなく、実際の物理メモリ空間の使用量を削減することを目標としている

並列実行

- ZCOMPの圧縮/展開をベクタ毎に行うとき、次のベクタの操作に前のベクタが必要
- 次のベクタを処理するために、現在のベクタの処理を完了する必要がある
- したがって、ZCOMPのスループットは、並列実行をサポートしていないと問題になることがある

結論

- 従来の物はカスタマイズされたデータパスとメモリ階層を使用して特徴マップを圧縮するが、汎用プロセッサでは困難
- CPUのクロスレイヤメモリフットプリントを緩和するために、ZCOMPベクトルISA拡張機能を導入
- ZCOMP命令は、ハイスループットな並列実行、キャッシュ階層や仮想メモリとの透過的な相互作用、柔軟なソフトウェアインターフェイスを提供する