

小口研究室 研究紹介 (2015年度)

(お茶の水女子大学理学部情報科学科)

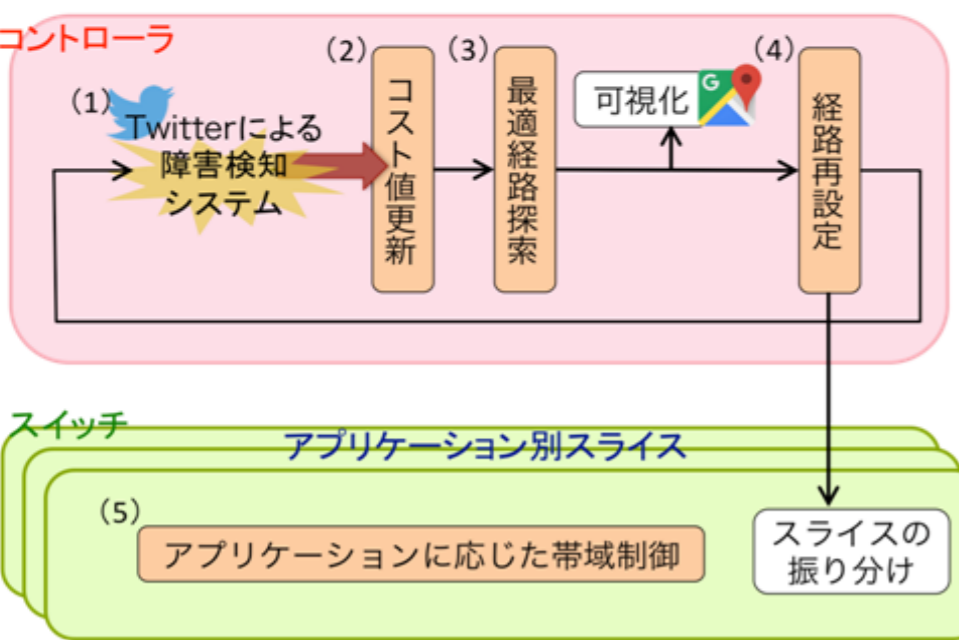
DPN環境におけるSNS情報に基づいたトラフィック制御方式の実装と評価 (研究担当: 柳田 晴香)

研究背景

大規模災害時も繋がるネットワークの必要性
 ◆東日本大震災時以降 SDN(Software Defined Network) による経路制御研究の活性化
 従来の経路制御における問題
 ◆「ネットワーク全体の状態を迅速に把握」困難
 「自動化、NWプログラマビリティ」限界

フルプログラム可能なDPN(Deply Programmable Network) 環境で、SNS情報をもとに経路制御を行うシステムの提案

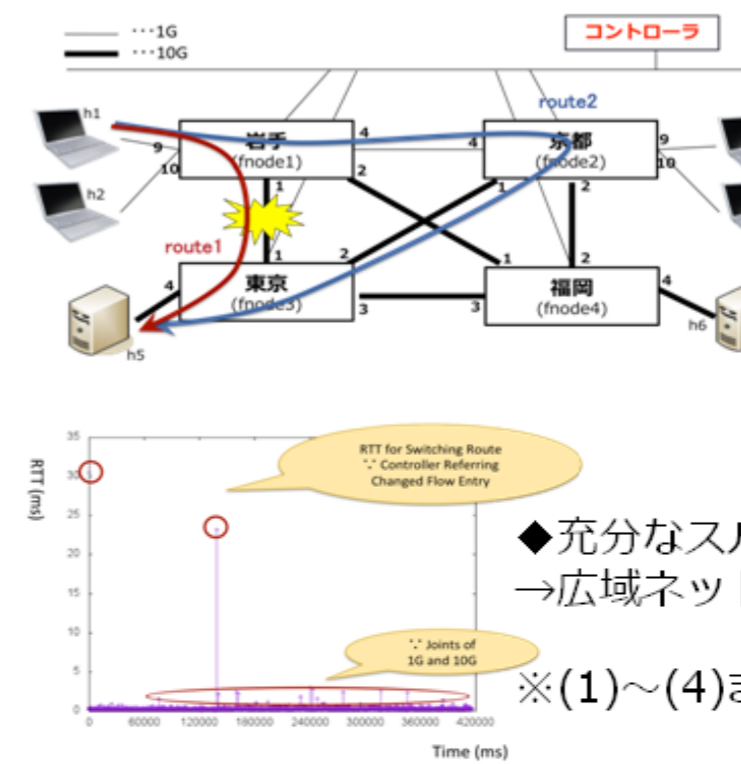
提案システム



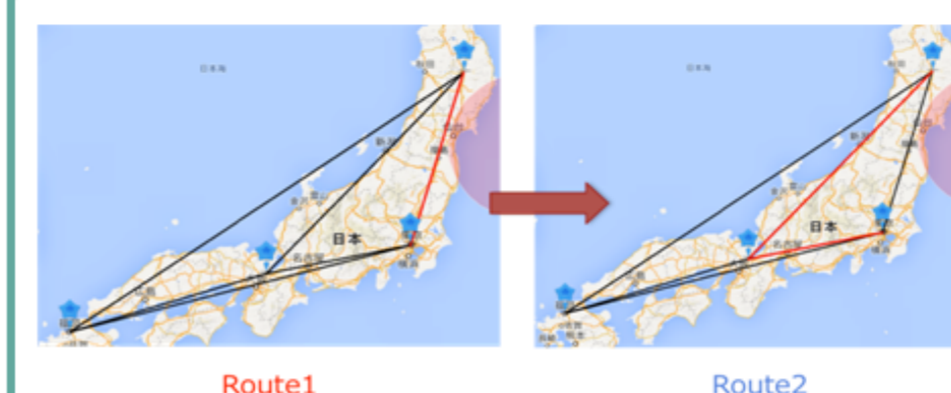
- (1)Twitterのリアルタイム解析よりネットワーク障害を検知
- (2)障害ツイート数と地域毎の優先度に応じてトポロジのリンク間コスト値を更新
- (3)ダイクストラ法より最適経路探索
- (4)最適経路に経路の再設定を行うフローエントリをREST-APIを通してスイッチ側へ投げる
- (5)フルプログラム可能なFLAREスイッチを用いて、アプリケーションの弁別とスライスへの振り分けを行い、スライス毎に経路制御・帯域制御等を行う

FLARE実機実験

| | | |
|---------------|--------|----------------|
| FLARE switch1 | CPU | Core i7-3612QE |
| | Memory | 8GB |
| FLARE switch4 | OS | CentOS 6.4 |
| h1~h4 | CPU | Core i5-4210 M |
| | Memory | 8GB |
| | HDD | SATA 500GB |
| | OS | Ubuntu14.04 |
| h5, h6 | CPU | XeonE3-1241 v3 |
| | Memory | 8GB |
| | HDD | SATA 1TB |
| | OS | Ubuntu14.04 |



経路切替の可視化



今後の課題

- ◆(5)アプリケーション毎のQoS制御の実装と評価
 ◆Twitterより詳細に抽出されたユーザの情報をもとにした制御

ディープラーニングフレームワークCaffeの分散環境への適用 (研究担当: 一瀬 絢衣)

研究背景

◆各種センサの普及
 ◆クラウドコンピューティング技術の習熟

監視カメラなどを用いた一般消費者のライブログの普及

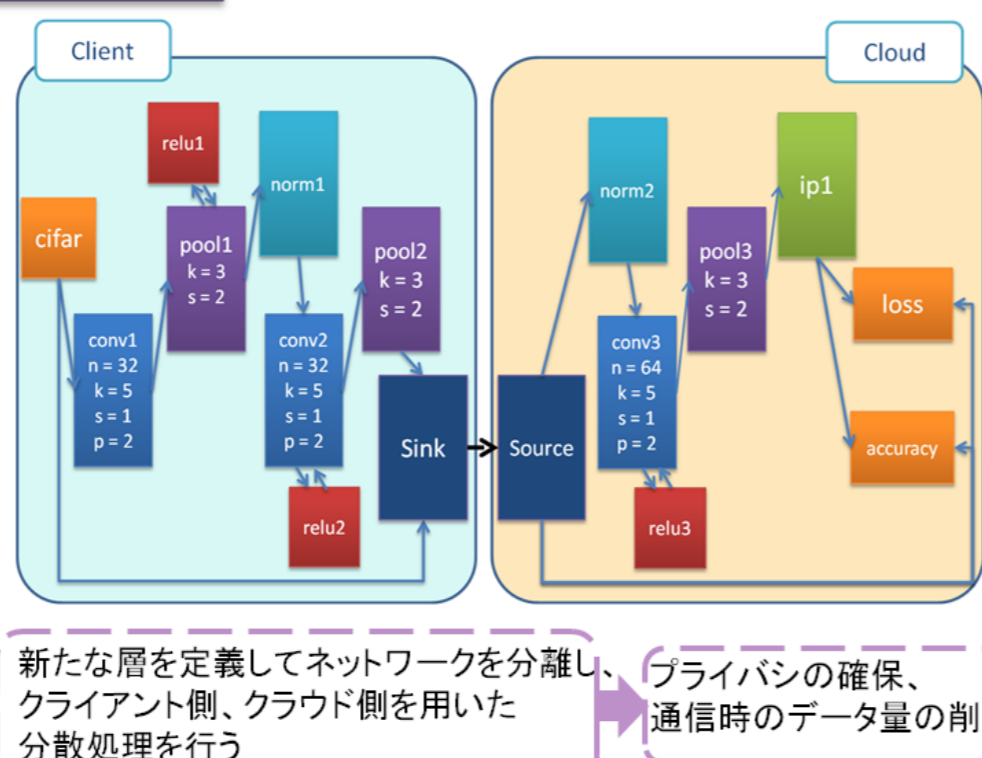
一般家庭で全ての処理を行うことは困難

映像をそのままクラウドへ送信

プライバシー、ネットワーク帯域における問題

ディープラーニングのフレームワークCaffeを分散環境に適用

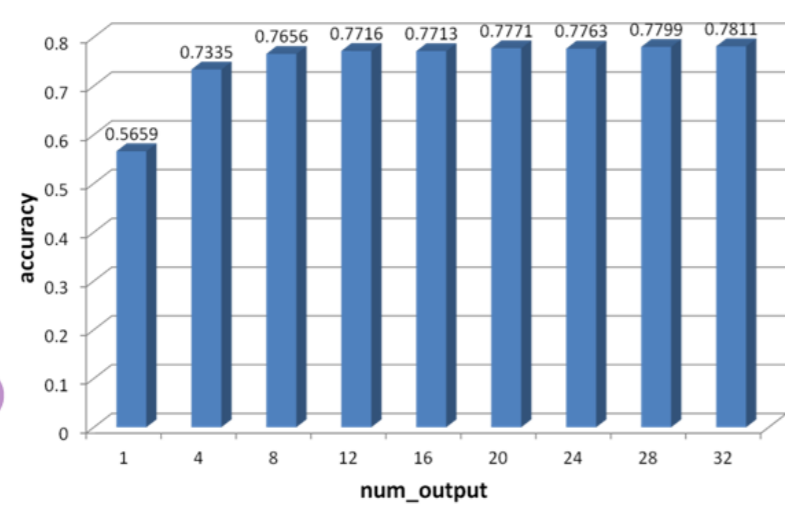
研究概要



識別率の測定

conv2層のフィルタ数を削減し、通信時のデータ量をさらに削減

フィルタ数と識別率の関係を調査

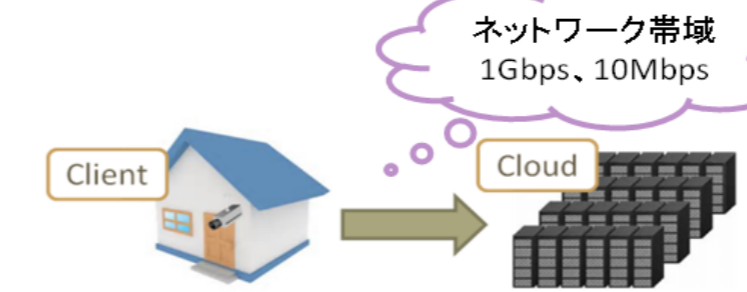


フィルタ数が少ない段階で識別率が収束

高い識別率を保ちながら通信時のデータ量の削減が可能

評価実験

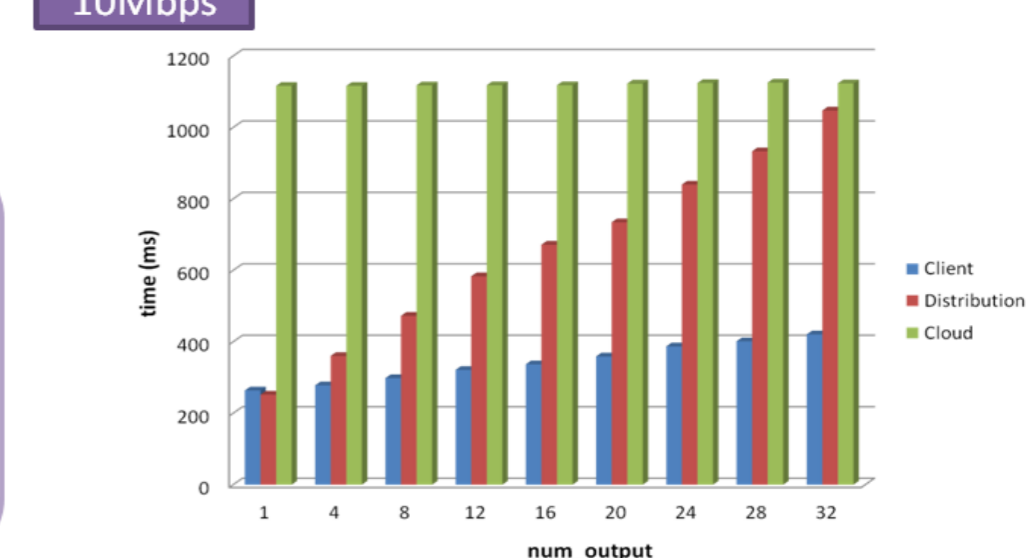
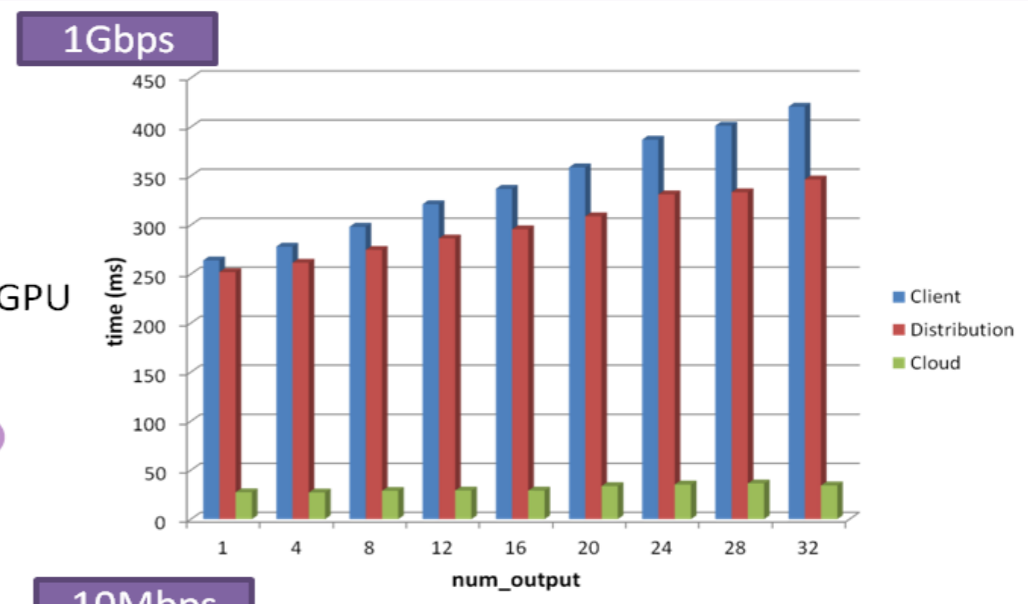
- conv2層のフィルタ数を変化させ、それぞれ1バッチの処理時間を計測
1. クライアントのCPUのみ
 2. クライアントのCPU + クラウドのCPU + GPU
 3. クラウドのCPU + GPU



考察

- ◆一般家庭とクラウドを考慮した通信環境では、映像をそのまま送信するのは効率が悪い
- ◆リソースに限りがあることからクライアント側で全ての処理を行うのは現実的ではない

分散処理が有効



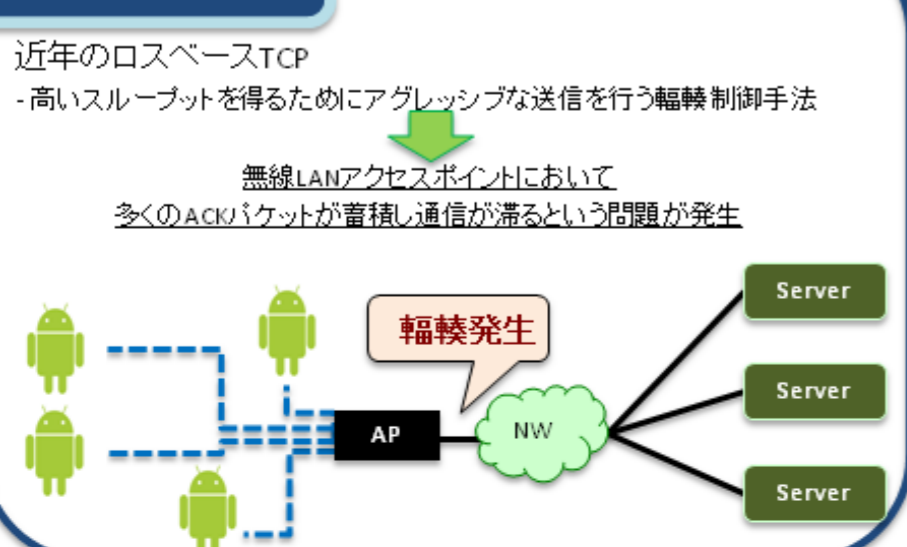
まとめと今後の課題

- まとめ
- ◆Caffeの分散処理によりプライバシー、ネットワーク帯域を考慮したセンサデータ解析処理を検討
 - ◆処理時間の比較により提案手法の有効性を示した

- 今後の予定
- ◆より大きなデータセット、複雑なネットワークモデルを用いた実験
 - ◆クライアント側により現実的な性能の端末を用いた実験

Android端末における通信制御ミドルウェアのタブレット端末への導入と評価 (研究担当: 島田 歩実)

研究背景



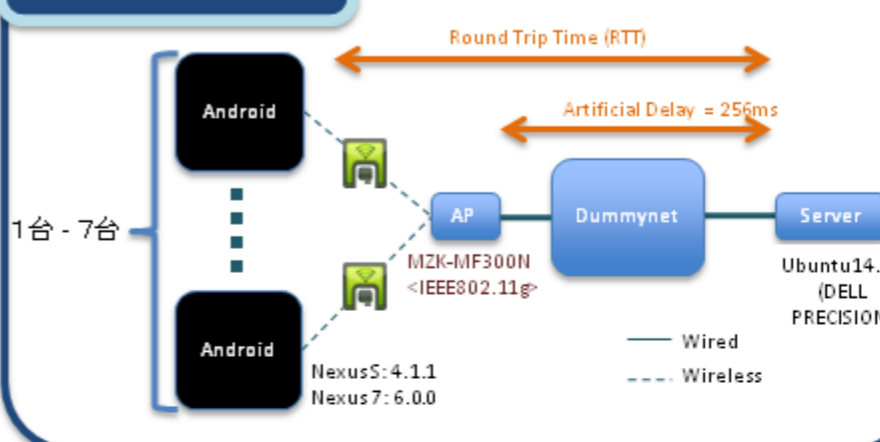
研究目的

- ◆通信制御ミドルウェアのタブレット端末への導入
 - 既存研究: スマートフォン端末(NexusS)のみでの実験
 - 本研究: 既存研究によって開発されたミドルウェアをタブレット端末に導入
 - ◆タブレット端末を用いた場合の通信性能の測定と検証
- タブレット端末
- メモリやCPUの性能が高い
 - 複数のアプリケーションを同時に動かすことができる
 - 大容量通信

既存研究

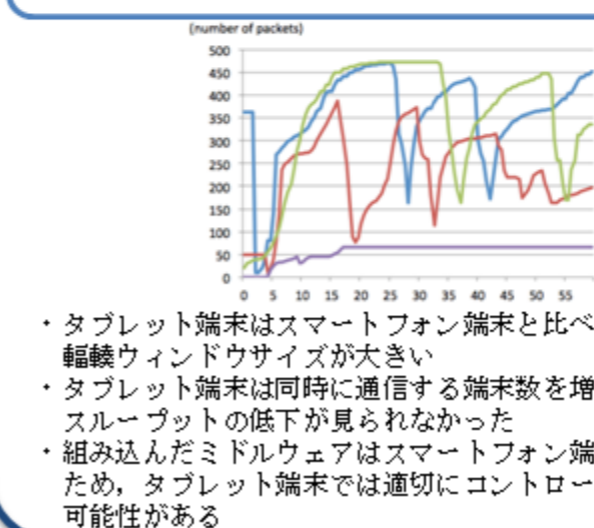
- ◆カーネルモニタ
 - 通信時におけるLinuxシステムのカーネル内部の処理を解析するシステムツール
 - カーネル内部のパラメータ値の変化を記録可能
 - Androidに組み込み、リアルタイムに解析
 - ◆輻輳制御ミドルウェア
 - 周辺端末の情報を利用し、連携した輻輳制御
 - APを共有する端末間で、通信状況に関する情報を通知
 - 同時通信端末数とRTTの増減によりトラフィックの混み具合を把握
 - 適切な輻輳制御の上限値を自動で算出し補正
- Real time visualize
- Monitoring TCP source
- Application
- TCP
- IP
- Ethernet
- Memory space in Kernel /proc file system
- Real time visualize
- ◆輻輳ウィンドウ
 - ◆ソケットバッファのキュー長
 - ◆各種エラーイベントの発生タイミング
 - ◆RTT

実験環境



性能評価1

iperfによって測定されたスループットやカーネルモニタによってモニタされたパラメータ値から、スマートフォン端末、タブレット端末それぞれの振舞いを検証、評価



性能評価2

通信を行うプロセスとしてiperfを使用し、1台のタブレット端末内で同時に実行する数を変更

- ◆プロセス数が1のときと比べ、複数のプロセスが同時に動作しているときは、CWNDの値は小さくなり、RTTの値が大きくなっている
- CWNDやRTTを制御することで、通信性能が向上する可能性

まとめと今後の課題

- まとめ
- ◆先行研究によって開発された通信制御ミドルウェアをAndroidが搭載されたタブレット端末に組み込むことに成功
 - ◆スマートフォン端末とタブレット端末の性能を比較
 - ◆タブレット端末の性能の高さに注目した性能評価
- 今後の課題
- ◆タブレット端末においても適切なコントロールが行われるよう、ミドルウェアを改善
 - ◆さらに遅延の大きい環境下におけるタブレット端末の性能評価
 - ◆スマートフォン端末やタブレット端末に加えノートPCも加えた実験