1A1-OS-05a-1

クックパッドにおける Deep Learningを用いた料理画像判別の取り組み

Approaches to Food/Non-food image classification using Deep Learning on cookpad

菊田 遥平 *1　染谷 悠一郎 *1　レシェック リビツキ *1
Yohei Kikuta　Yuichiro Someya　Leszek Rybicki

*1クックパッド株式会社
Cookpad Inc.

In this paper we report our approach to image classification, in particular to the food/non-food image classification problem, as used by our 料理きろく (Cooking Log) product of Cookpad Inc. We augment our existing services with a computationally expensive image analysis architecture implementing this solution. One challenge is that the non-food class is very vast and varied and can only be defined in context. We find that having the non-food class consist of multiple subclasses effectively improves both precision and recall by capturing different types of features in the images of the non-food class.

1. はじめに

本稿では, クックパッド *1 における「料理きろく」(図 1) というプロダクトを題材として, Deep Learning を用いた料理画像判別問題に関する取り組みを報告する. 特に,Deep Learning を組み込んだプロダクトのアーキテクチャの提案と, 料理・非料理判別の精度向上を目的として実施したアプローチとその効果の考察を行う.

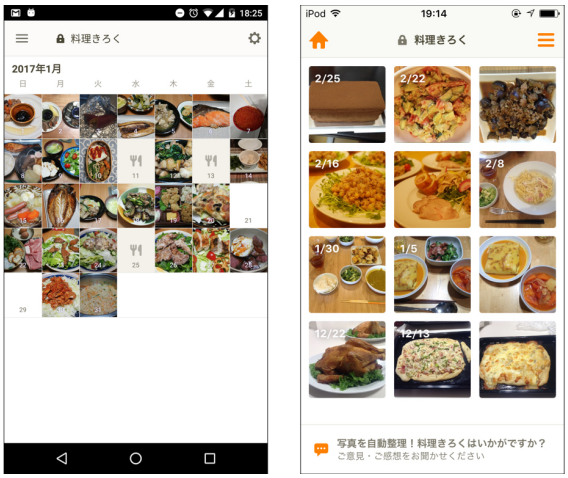

図 1: 料理きろくのキャプチャ画面. 左側が Android で右側が iOS の画面であり, 携帯端末中の料理画像を自動的に抽出して表示している. 画像をタップすることでフィードバックを送ることも可能になっている. 料理きろくはユーザ数が約 5 万人, 料理と判別された画像が約 130 万枚 (2017 年 3 月 7 日時点) のプロダクトである.

進展の著しい機械学習の分野の中でも特に Deep Learning の隆盛は驚嘆すべきものがあるが, Deep Learning を組み込んだプロダクトを一定度の規模でかつ定常的に運用している事例はそう多くないと思われる. その理由として以下のようなものが考えられる.

連絡先: 菊田遥平, クックパッド株式会社 研究開発部, yohei-kikuta@cookpad.com

*1 https://cookpad.com

- 十分な技術やデータが揃っていない
  高い技術力を持っていても MNIST のような実際のサービスとの関連が薄いデータでは活かせる場が限定的である. また、データが豊富でもそれを活用しきる技術がなければ機械学習の威力を享受することはできない.
- 既存サービスに組み込む際のコストやリスクが大きい
  Deep Learning を用いた分析は高負荷で分析環境も特殊なものを要求するため, 他サービスに悪影響を与えずに有効に扱うには分析以外の知見と技術が必要である.
- 費用対効果が見込めない
  多くの場合,Deep Learning を取り入れてもすぐには高い効果を望めない. 中長期的な視座とそれを実行できる組織体制が必要である.

クックパッドはこれらの課題を解決してサービスを提供できる稀有な企業の一つであり, 現在進行形で様々なサービスの提供と新しいプロダクトの開発を推進している.

以降の章では, 我々の問題設定とその解決のために実施した具体的な取り組みの技術的側面を説明し, その結果と今後の展望を述べる.

2. 問題設定

クックパッドは月次利用者数が 6,300 万人超で登録レシピ数が約 260 万品（2016 年 12 月末時点）という日本最大のレシピサービスである. 多くのユーザが使用している巨大なサービスであるが, 巨大であるがゆえに限定的な利用に留まるユーザも存在する. 例えばレシピを検索して料理を作り写真も撮るが, 物理的・心理的障壁により, それを投稿するというサービスにとって有用なアクションまでは到らない場合も少なくない. これはサービスにとって重要な課題の一つである.

このような課題を改善するための試みの一つとして, 画像分析技術に基づくプロダクトに注目する. 具体的には, ユーザの携帯端末中の画像から料理画像のみを抽出し, 料理の履歴が簡単に閲覧できることに加え, その画像を起点としてサービスにフィードバックも送ることができる「料理きろく」というプロダクトに注目する. 本稿では, このようなプロダクトを実現するためのアーキテクチャの構築と, 料理画像抽出部分における

Deep Learning を用いた画像の料理・非料理判別の性能に関して述べる．この二点に関して要求される観点としては次のようなものが挙げられる．

- アーキテクチャの構築
  Deep Learning を用いた画像判別は重い処理でかつモデルの環境依存性が高いため, 下記の観点が重要となる．
  - 非同期で処理を行う
  - 他サービスへの影響を可能な限り抑える
  - 判別モデル部分のコンポーネントを他コンポーネントと疎結合にする
- 料理・非料理判別
  単純な画像判別問題は Deep Learning により解けたと言われているが, 実際のプロダクトはそう単純ではなく, 下記の観点が重要となる．
  - 料理・非料理のような二値判別問題において, 限られたデータでは一方が他方の補集合とはならないため, 精度を高める工夫が必要となる
  - ユーザ心理を考慮して判別をコントロールする（precision と recall のバランスを取る）
  - ユーザのプライバシー保護のため真のデータ分布にはアクセスできないため, モデルの評価に注意する

3. 具体的な取り組み

ユーザの携帯端末中の画像から料理画像のみを自動的に抽出する機能を実現するための構成要素として, 本章では全体のアーキテクチャ構築と Deep Learning モデルによる画像の料理・非料理判別精度向上に関して述べる．

3.1 アーキテクチャ

Deep Learning により学習されたモデルによる画像判別は大量の演算処理を伴うため, 軽量な API コールに対して相対的に処理時間が長くならざるを得ない．一方で, クックパッドの API サーバーとして利用されている Unicorn[*2] は, リクエストを同時に処理するプロセスの数が限定されている．このような場合, 画像判別を API コールとして同期的に実現する事は, 画像判別のリクエストが大半のプロセスを専有し, その他の軽量なリクエストの処理が滞るなどの影響が考えられるという点で現実的でない．そのため, 画像判別はクライアント (iOS, Android) からのリクエストに対して非同期に行う必要がある．

料理きろくにおいては, この問題を解決するために, 非同期メッセージキューを利用したアーキテクチャによる処理フローを実現した．図 2 にアーキテクチャの全体図を示す．この時, クライアントから判別のためのリクエスト (図の A の部分) と, 実際の判別処理 (図の B の部分) が非同期に行われることが肝要となる．画像の判別結果や状態の管理等, 判別処理以外の軽量な処理は API サーバーを介して行い, 判別処理そのものはメッセージキューを介することで非同期性を確保した上で行い, 判別結果を API サーバーを介してアプリケーションから利用可能な状態にする, という手法を採用した．

また, 判別処理を行うモデルとの通信においてメッセージキューを経由することで, モデルとアプリケーションの関係を疎結合に保っている．Deep Learning によるモデルの学習は, 通常それぞれに特色を持った Deep Learning 専用のフレームワークにより行われ, 判別処理も基本的にフレームワークの枠組みの中で行われる．また, 全体の計算コストの削減とパフォーマンスの向上を目的として GPU を利用する事が多い．従って, このようなモデルは動作に適した環境が一般的な Web アプリケーション,API サーバー等とは異なる場合が多い．このような性質の異なるコンポーネントを疎結合に保つことで, それぞれのコンポーネントのアップデートが容易になる, 故障がシステムの他の部分に及ぼす影響を抑えることができる, などのプロダクトにおいて重要となる観点を考慮したアーキテクチャを実現した．

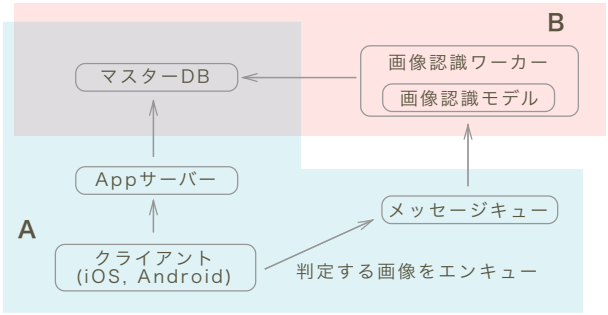

図 2: 料理きろくのアーキテクチャ概略図. 携帯端末中の料理画像を自動的に抽出して表示する．

3.2 料理・非料理判別問題

画像の料理・非料理判別器は,Deep Learning のモデルを学習することで構築する．学習用のデータとする料理と非料理の画像は, クックパッドのウェブサイトもしくはライセンスフリーのものをそれぞれ 80,000 枚程度収集した．

3.2.1 ベースラインのモデル

我々が最初にプロダクトにデプロイしたモデルである AlexNet [Krizhevsky 12] をベースラインのモデルとして考える．これはプロダクトとして迅速にデプロイするために採用したモデルである．事前学習を施したモデルに対して, 料理・非料理画像の二値判別問題として転移学習したモデルとなっている．非料理を料理と誤判別する（例えば人間を料理と判定して表示してしまう）リスクが大きいため, softmax の出力において料理と判定する際の閾値は 0.9 と設定した．

3.2.2 考慮するモデル

判別性能を向上させるため, 様々なモデルを実装し, その判別性能を検証した． 本稿では, 実施した検証の一部の例として,GoogleNet [Szegedy 15] と VGG19 [Simonyan 14] を取り上げる．これらのモデルの実装には Caffe [Jia 14] を用いた．

3.2.3 学習モデルの多クラス化

料理・非料理のような, あるクラスとその補集合という二値判別問題には本質的な困難がある．料理画像の全集合 X_1^{all} が把握できているならば $X_0^{all} := \overline{X_1^{all}}$ こそが非料理の定義となるが, 現実のデータは大量かつ動的に生成されていくものであり全集合を扱うことはほとんど原理的に不可能である [*3]．1 class classification のような枠組みも存在するが, 画像のような多様性のある対象に対しては, 筆者らの知る限り実用上十分な性能を達成できているものは存在しない．

*2 https://bogomips.org/unicorn/

*3 何を以って料理とするかという問題もあるが, ここではラベルは与えられているものと仮定し, この問題には立ち入らない．

上記のような状況に鑑み, 実際には料理の画像と非料理の画像の両方を収集して学習することになる. モデルにとっては, 非料理は料理でないものではなく, 収集したデータから学習された非料理らしいものを指す. しかしながら, 非料理画像とは幅広い概念であるため, 単一のクラスとして扱うには特徴量の分布は分散が大きすぎるものと考えられる. 実際に, 単純な二値判別モデルにおいては閾値を高く設定しても非料理を料理と誤判別する場合があり, その典型的な例が図 3 のような画像である.

図 3: 初期のモデルで誤判別されやすかった画像の例. 画像は http://www.publicdomainpictures.net/ から取得.

このような問題を回避するため, 我々は出力を複数クラスに設定することで精度を上げることを試みた. 料理・非料理共に単一クラスの場合と複数クラスの場合でモデルを構築し, その性能を比較した. 複数ラベルの場合の料理・非料理判別問題は以下のように定式化する.

まず, 二値判別の場合は予測対象となるラベルは次のように表現できる.

$$t \in \mathcal{T} = \{0, 1\} := \mathcal{T}_0 \oplus \mathcal{T}_1 \tag{1}$$

我々の問題設定においては, 複数クラスへの拡張は次のように定式化することができる.

$$t \in \mathcal{T} = \{(0, 0), \cdots, (0, k_0), (1, 0), \cdots, (1, k_1)\} \tag{2}$$

$$\mathcal{T} := (\mathcal{T}_{0,0} \oplus \cdots \oplus \mathcal{T}_{0,k_0}) \oplus (\mathcal{T}_{1,0} \oplus \cdots \oplus \mathcal{T}_{1,k_1}) \tag{3}$$

ここで,k_i where $i = \{0, 1\}$ は親クラス i に属する子クラスの数を表す. モデル学習時はこれらの子クラスは別のクラスとして複数クラス判別問題として扱うが, 予測の際には $\mathcal{T}_{i,a} \rightarrow \mathcal{T}_i$ where $a = \{0, \cdots, k_i\}$ と落とし込むことで二値判別を実施する.

本稿では, 料理クラスにおいてはパンやパスタなどの 7 クラス, 非料理クラスにおいては子供や植物などの 11 クラスを準備し, 料理・非料理それぞれにおいて単一クラスの場合と複数クラスの場合での性能を検証した. これらのクラスの選び方は heuristic なものである.

4. 結果

前章で述べた取り組みによる結果をアーキテクチャとモデルの観点から記述する.

4.1 アーキテクチャの安定性と柔軟性

定量的な評価が難しいところもあるが, 我々が構築したアーキテクチャによって得られた効果をいくつか述べる.

料理きろくのプロダクトはリリース以降安定的に動作し, 他サービスに影響を与えることなく, これまでに合計で約 780 万枚の画像を判別し, そのうち料理と判別されたものが約 130 万枚となっている（2017 年 3 月 7 日時点）.

また, 次節で述べるモデルの精度評価に従い, リリース後に一度モデルをアップデートを実施した. その際には Deep Learning のモデルを実装するためのライブラリ自体も変更したが, モデル部分の疎結合性により, 特筆すべき問題や工数が発生することなく移行が完了した.

本プロダクトとは別の場面で画像判別部分の機能が必要となった際も, モデル部分だけを切り出すことで容易に同様の機能を提供することが可能であった.

4.2 料理・非料理判別モデルの精度

まず, モデルの判別性能を評価するためのデータセットに関して述べる. 我々のプロダクトにおいては, ユーザのプライバシーを考慮して画像そのものを視認することができないため, 真のテストデータを扱うことはできない. そのため, 表 1 のようなテストデータセットを構築してモデルの性能を評価した. このテストデータセットは, 特定の画像の種類に偏らないように注意を払いつつ, モデルにとって判別が難しいと経験上明らかになったものを加えて構築したものである.

料理画像	非料理画像
3,028	10,086

表 1: モデルの性能評価で用いたテストデータセット. 数字は枚数を表す.

精度指標に関しては, 我々の問題設定において重要なのは料理画像に対する precision と recall であるため, これらの指標を用いてモデルの評価を実施した. ただし, 実際にデプロイされるモデルは単純に F 値が最も良いものではなく, recall を高い水準で保ちつつ precision が最も優れたものであることに注意されたい. これは, 非料理写真が料理として表出されることがユーザにとって不快感を与えやすいことを考慮している.

ベースラインの AlexNet の結果は表 2 のとおりである. 閾値の設定により precision が高い値を取るようになっているが, それでも図 3 のように誤判別をしてしまうケースも無視できない割合で存在している. また,recall が低く料理画像の取りこぼしが多くなってしまっている.

AlexNet P/R
0.973 / 0.758

表 2: ベースラインのモデルにおける結果. 料理クラスに対する precision(P) と recall(R) の値を記載している.

次に,GoogLeNet と VGG19 に関して, 多クラス化も含めて性能を評価した結果が表 3 である. この結果から, 多クラス化は性能向上に有用であるが, その効果には料理クラスと非料理クラスの間で顕著な違いが生じていることが伺える. 料理クラスというある種の定まったクラスに関しては, 多クラス化によって recall は上昇するが precision は減少するという振る舞いを見せている. これはモデルが抽出する料理らしいという特徴量分布に対して, 多クラス化によってその分布の周辺までも料理と認識することで, より多くの料理画像を判別できる代わりに少し似ている非料理の画像も拾ってしまうためだと考

{ 料理, 非料理 } クラス	GoogLeNet P/R	VGG19 P/R
{single, single}	0.911 / 0.900	0.829 / 0.861
{multi, single}	0.864 / **0.947**	0.891 / **0.918**
{single, multi}	**0.992** / 0.910	0.980 / 0.725
{multi, multi}	0.984 / 0.930	**0.990** / 0.883

表 3: 性能評価の実験結果. 一番左の列はモデル構築の際に単一クラスにしたか複数クラスにしたかを表している. モデルの列における数値は料理クラスにおける precision(P) と recall(R) の値を記載している.

えられる. 一方で, 非料理クラスという多様で大きく異なる特徴量分布の集合で構築されるクラスに関しては, 多クラス化によって precision と recall の両方が上昇する振る舞いを見せている. 単一クラスに押し込める場合では捉えきれない特徴量分布に関しては微妙な違いで料理・非料理双方の誤判別が生じ得るが, 適切なクラスを追加することでそれらの判別がつきやすくなるためだと考えられる.

F 値が最も高いモデルは料理・非料理ともに多クラス化したものだが, 提供するサービスの性質上, 我々は料理クラスは単一クラスで非料理クラスは多クラス化したものを採用している.

5. 結論

本稿では, 料理きろくというプロダクトに注目して, クックパッドにおける料理画像判別の取り組みに関して述べた. 処理が重く環境依存性も高い画像分析コンポーネントを疎結合で実現するアーキテクチャを構築し, Deep Learning を用いた料理・非料理の判別精度向上には非料理クラスを多クラス化することが特に有効であることを明らかにした.

今回の取り組みによってモデルの判別性能を向上させることに成功したが, 対象となるデータ分布は動的に変化し続けていくものであるため, 試行錯誤を繰り返し継続的にモデルの改善を行っていくことも必要であると考えている.

課題としては, 現状のテストデータは偏りが生じないように注意を払ってはいるが恣意的に収集したものであるという点が挙げられる. 適切な評価のためにはテストデータを真の分布に近づける必要があるため, 画像の収集や選別などの取り組みを継続して行っている.

本稿ではクックパッドにおける代表的な取り組みとして料理・非料理判別を取り上げたが, それ以外にもメニューや具材の判別や料理部分の領域検出など, 様々な画像分析の課題に取り組んでいる.

参考文献

[Jia 14] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T.: Caffe: Convolutional Architecture for Fast Feature Embedding, *arXiv preprint arXiv:1408.5093* (2014)

[Krizhevsky 12] Krizhevsky, A., Sutskever, I., and Hinton, G. E.: Imagenet classification with deep convolutional neural networks, in *Advances in neural information processing systems*, pp. 1097–1105 (2012)

[Simonyan 14] Simonyan, K. and Zisserman, A.: Very deep convolutional networks for large-scale image recognition, *arXiv preprint arXiv:1409.1556* (2014)

[Szegedy 15] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A.: Going deeper with convolutions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9 (2015)