

運用で泣かないアーキテクチャで動く原稿作成支援システム ～リクルートにおけるDeepLearning活用事例～

リクルートテクノロジーズ 西村隆宏

概論

リクルートは多種多様なアルバイト求人、レストラン予約、美容院予約等様々なマッチングプラットフォームを運営しており、日々多数の原稿を書き、カスタマーの皆様へ価値ある情報を届けている。一方原稿を書く負荷は非常に高く、現状1原稿あたりにかかる時間が長いという問題がある。リクルートテクノロジーズではその問題を解決するため、原稿作成の支援システムを開発している。内部ロジックにはRecurrent Neural Networkを用いており、入力中に次に出現する文章を推薦することで原稿の文章作成にかかる負担を減らす取り組みを紹介する。またユーザー企業では運用の問題も発生する。機械学習の仕組みの運用負荷を低減させる取り組みについても紹介する。

本講演で紹介する項目は下記の3点であり、それぞれについて説明する。

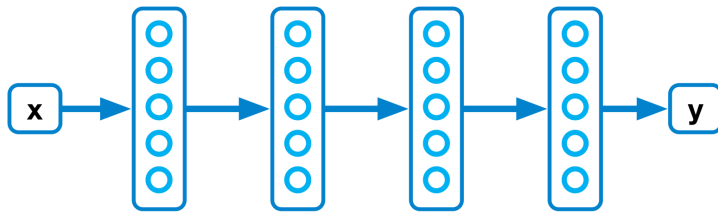
- リクルートにおける原稿作成の負荷
- Recurrent Neural Networks(RNNs)を用いた文章推薦システム
- 高速に検証サイクルを回す、運用負荷を下げるための取り組み

リクルートにおける原稿作成の負荷

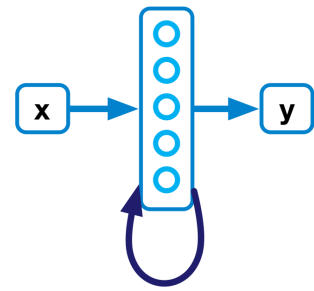
リクルートではカスタマーに届ける個別情報を一般に原稿と称しており、アルバイトや正社員の求人原稿、旅行における旅館の原稿、住宅における物件情報等が当てはまる。求人原稿や物件情報、結婚式場情報といったライフイベント領域、旅館原稿、レストランの詳細情報といった日常消費領域のあらゆる場面で原稿を記入しているが、原稿作成者の負荷が非常に高い。提供サービスや担当者によって差はあるが、担当クライアントに最も近い場所にいる営業マンが書く場合もある。ただ営業マンは営業や提案活動の領域におけるスペシャリストのため、その領域に注力できるような仕組み化が必要である。そこでリクルートテクノロジーズは営業マンの原稿作成負荷の低減を目指すため、原稿作成支援システムを開発することになった。

Recurrent Neural Networks(RNNs)を用いた文章推薦システム

原稿作成支援システムにおける文章推薦にRNNsを使った言語モデルを利用した。RNNとはNeural Networksの一種で、よく用いられるフィードフォワード(FF)のネットワークと違い、ネットワーク内に閉路を持つNeural Networkである。



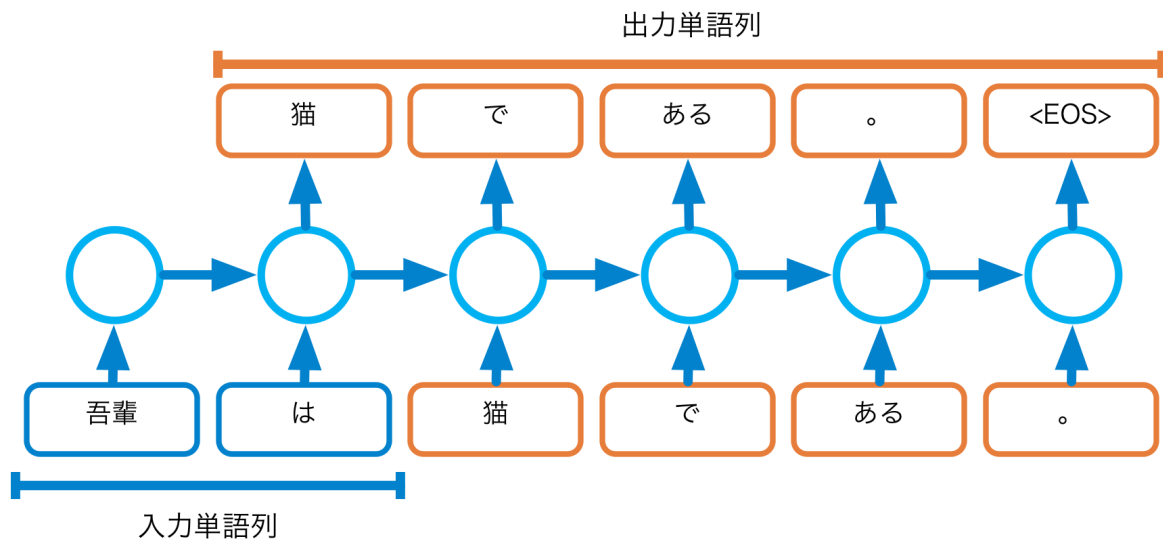
フィードフォワード型のNeural Networks



Recurrent Neural Networks

FF型Neural NetworksとRNNsのネットワークの違い

RNNsは、隠れ層に閉路を持つため、系列の概念が入る。そのため時系列データに対する識別で用いられる場合が多いが、言語モデルへの適用も可能である。言語モデルを単純に説明すると、与えられた単語列の次に来る単語を予測するモデルである。そのため、系列の概念がはいるRNNsが近年よく用いられるようになった。一方単語の出力を行う機械は文章推薦システムとはいえない。よって今回は出力された単語を入力に入れる事で再帰的に単語を出力し、出力単語を見て文末である、と判断された場合に処理を打ち切り画面にフィードバックすることで文章推薦システムを開発した。



RNNsを言語モデルに応用した例

Suggestボタンを押すと次に来る文章を予測するよ!

- 単語をサジェスト
- フレーズをサジェスト
- 文章をサジェスト

SUGGEST SENTENCE

未経験

サジェスト結果

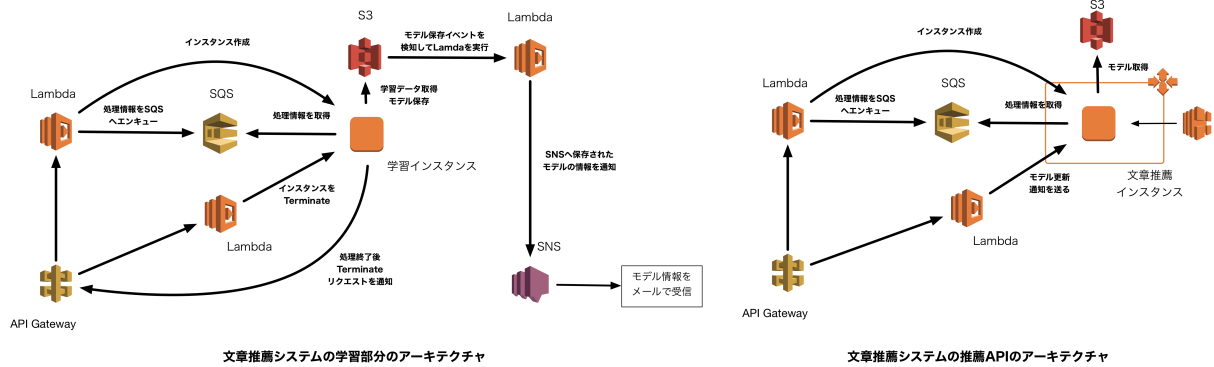
- > スタート也大歓迎です♪
- > の方も安心!
- > から始められるお仕事です!
- > からスタートしたスタッフも多く、未経験から始めたスタッフも多数在籍!
- > OK!
- > の方も大歓迎!
- > でもOKです!
- > 者の方も活躍中!
- > 者也大歓迎です♪

文章推薦システムのサンプル画面

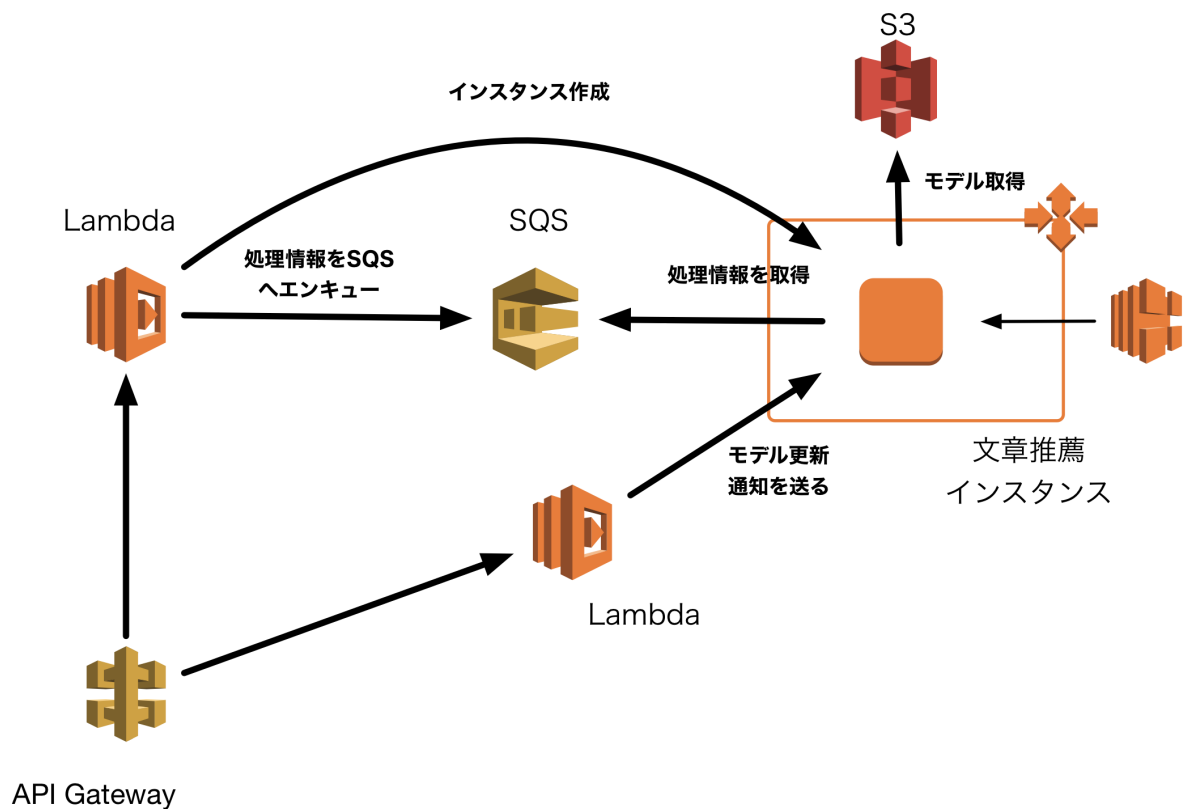
高速に検証サイクルを回す、運用負荷を下げるための取り組み

文章推薦システムはAPIとして提供されるシステムで入稿システムからXmlHttpRequestで呼ばれる。よって単に文章推薦システムを構築するにはシンプルなアーキテクチャになる。

一方RNNsはネットワークの構造の違い、ネットワークのパラメータ、最適な勾配降下法の選択および各勾配降下法のパラメータといった多数のチューニング要素が存在するが、一回の学習が非常に重いため全てを試すには時間が足りない。また、リクルートは様々なマッチングプラットフォームを運営し、データも異なるため、同じようなモデルを複数作る事も存在する。これを高速に検証するためにはなるべく管理する部分を減らす必要がある他、手作業で行う開発を極力少なくしなければ事業の利用ニーズに応える事は難しい。また、今後を想定すると機械学習について明るくない人でも実行できるようにしなければならない。よって、文章推薦システムを下記のアーキテクチャで実行することにより、簡単に文章推薦システムを試す事が可能になった。



文章推薦システムの学習部分のアーキテクチャ



文章推薦システムの推薦APIのアーキテクチャ

また、今回採用したアーキテクチャは機械学習のロジックをAPIとして利用するユースケースに広く対応可能であるため、環境構築においてもAnsible, Packer, Terraformを利用しコードで管理することを徹底した。その結果別プロジェクトにおいても環境構築の工数が大幅に削減された。そして利用したAWSのサービス群のうち、フルマネージドサービスを利用することで運用対象を減らし運用負荷を低減させた。API Gateway, Lambda, SQS, SNS, S3の部分である。AWSのマネージドサービスを活用するほど、AWSへのロックインが発生するが、今回はAWSから他のクラウドサービスへの移行が起きにくい事、運用負荷を低減することが目的だったため積極的に利用した。

おわりに

本稿は文章推薦システムへのDeepLearningの応用、および開発者への開発工数を削減する

ためのアーキテクチャ、開発環境の整備について論じた。今後は文章推薦において、推薦されるテキストが文章として体をなしているのか、推薦されるテキストの多様性についての検証の他、リクルートのサービスへの展開を考えている。