Associated Press: Text-To-Image Recommendation System
Scope of Work (SoW) and Progress Report

Background

The Associated Press (AP) is a cooperative, not-for-profit, that serves news in over 100 countries[1]. Members of this cooperative grant AP permission to distribute to distribute their content. Non-member newspapers subscribe to AP and pay a fee to get their material published. Another source of revenue for AP is the sale of licensed AP content, such as images, videos and audio files, to independent editors and publishers through AP Images[2].

Problem Statement

AP Images contains numerous genres of royalty-free and rights managed images, music tracks and videos. Currently, editors and publishers search through the AP Images website manually in order to find images that are ideally suited to their article. AP's goal is to sell as many of these images to editors and publishers as possible and to make the process of finding relevant, high-quality images as smooth as possible.

This motivates a text-to-image recommendation system, where users can input their article text and view a set of matching images from the wide range of images at AP's disposal. Our task is to come up with a model that maps input text to relevant AP images using the text as well as the metadata of the images and articles, such as captions or tags.

Motivation

From AP's perspective, the creation of an efficient, automated interface for both internal and external clients allows for generation of quicker and more fitting image

---

[1] Source: https://www.ap.org/about/annual-report/2016/ap-by-the-numbers.html
[2] Source: http://www.apimages.com/

recommendations, promoting the AP Images service. The challenge from a natural language processing perspective is modeling the way a client would think about an image through a creative combination of NLP techniques with a flavor of computer vision.

Some potential issues in modeling image recommendations include access. We do not have access to the entire AP Image corpus, only those associated with a subset of existing articles. This means we need to develop creative ways to deal with out-of-sample articles. Additionally, we need to look at ways to explore article-image relationships other than simple tag-to-tag matching. This is where neural network approaches may come in handy to generate more information.

Resources Available

Data available includes:
- AP articles
  - 30,098 articles
    - Published / created from Feb - Sep, 2019
    - 99.9% in English
    - 11 missing full text
    - 7,348 without associated media files
    - Metadata includes tags about article subject, person, organization, and place information
- AP images
  - 81,126 media files
    - 96.8% images
      - Other file types: audio and video

API Resources:
- Tagging Service API
 - Rule-based tagging service for input text

High-Level Project Stages
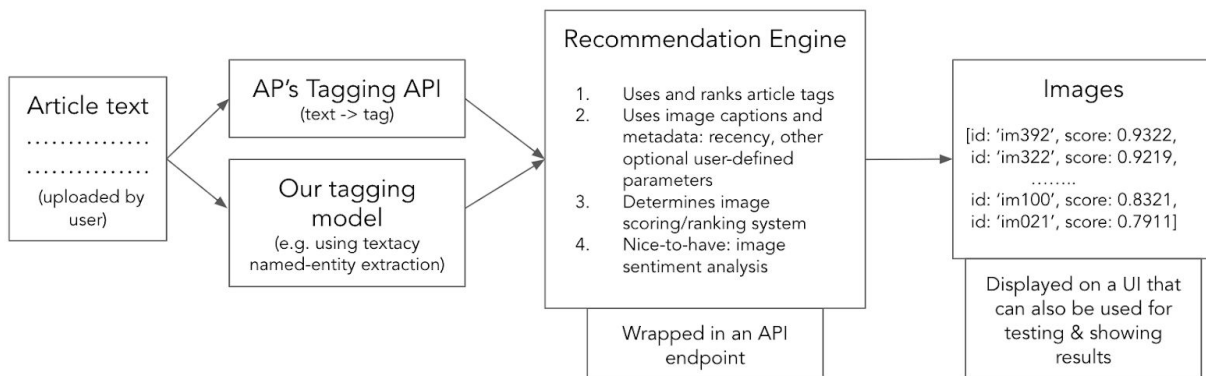
- Tag candidates optimization
  - Optimize candidate tags to recommend
- Modeling
  - Once we have the good tag candidates, we will explore different machine learning techniques that leverage data provided by AP of the linkage between article and images to recommend images to input text
- UI/API
  - Once we have a working model, we can wrap it under a user-interface so that a user can input article and get image recommendation

Progress Report

- Data Exploration and Analysis Setup
  - We have successfully retrieved all the data granted by our partner and set up the workflow to process the given json files into csv files
  - We have formatted the data of the 30,098 articles in a tabular format, that each row is an article. Similarly for images, we have organized the data of the 81126 media in a tabular format, that each row is a media
  - For exploratory data visualizations, we focused on tag-to-tag matching between article and image tags. The figure below displays the number of unique tags for subject, person, organization, and place tags. We see that overall, the articles come with more unique tags than images

Unique Tag Count by Content Type
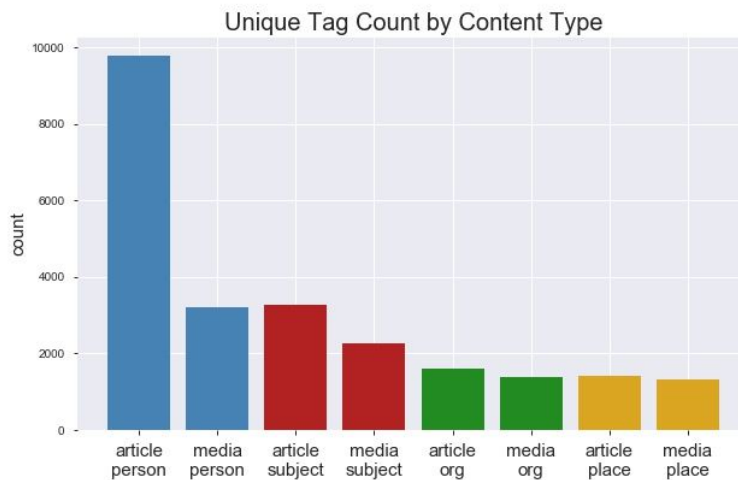
- ○ Based on the visualizations of tag-to-tag matching (details included in the attached slides), we identified potential challenges in setting up tag-to-tag matching models. Overall, without optimizing tags based on their relevance to the articles, we would need a large amount of images to capture all the article tags, as images come with fewer, and relatively unique tags
  - ○ Additionally, it appeared that images may be selected based on a subset of tags. As the current selection process is through human judgement, we are reviewing deep learning models that help capture such subjective selection processes by learning the latent relationships between articles and their associated images, without explicit preprocessing procedures (e.g. separating articles by type)
- ● Model
  - ○ We have set up our preliminary baseline model using TF-IDF
  - ○ Workflow to complete Named Entity Recognition and TextRank will be finished in a week

Project Timeline

Oct 15 - Oct 22:

- ● TextRank and named entities on full text
- ● Data Cleaning:
  - ○ Identify final versions of articles / images and drop duplicate values
  - ○ Separate article data by whether media files are attached
- ● Matching tokens from article with image tags
- ● Check if same images have different ids, in order to create a naive Bayes model
  - ○ P(image): empirical distribution of an image being used

- - ○ P(image|text): variable of interest
    - ○ P(text|image): text information about the article
    - ○ P(text): treat as constant in optimization
    - ○ P(image|text) = P(text|image) * P(image)/P(text)
  - Lit review on deep learning models
  - Taxonomy data

Oct 22 - Oct 29:

  - kNN model + tf-idf score of tags (model 1)
  - Start building the API and a pipeline between our models and the UI
  - Review lit review & start first iteration of modeling
  - Explore tagging API

Oct 29 - Nov 5:

  - Model training (model 2)
  - Include first model into the UI and see if it works
  - Build an API wrapper for our models

Nov 5 - Nov 12:

  - Model training (model 2)
  - Evaluation metrics (tags/what we use in recommending from text to captions of image)
    - ○ Recall, precision and F1-score
  - Improve API wrapper
  - Prepare milestone 3 presentation

Nov 12 - Nov 19:

  - Model training (model 2 / 3)
  - Iterate UI and API wrapper

Nov 19 - Nov 26:

  - Training complete
  - Nice to haves: sentiment matching & UI/UX
  - Meet with AP on next steps
  - Blog draft

Nov 26 - Dec 3 (Thanksgiving):

- Poster draft
- Blog intermediate draft
- Final presentation draft
- Model optimization
- Finalize UI/API endpoint
- Get code base ready for shipping

Dec 3 - Dec 12:

- Prepare IACS Showcase (Dec 9)
- Prepare final presentations (Dec 11)
- Finalize blog post (Dec 12)
- Finalize deliverables to AP (Dec 12)

Deliverable

- Recommendation model
  - Trained on data from AP text and AP images
  - It accepts input of article text and recommends images to the text
- API endpoint
  - This is a wrapper for our models and can be called by our UI and potentially from AP's other client-facing UIs
- UI
  - It takes article text as input and outputs recommended images
  - It supplements the recommendation model by providing a nice interface to show images recommended as well as tags/weights
- Poster
  - We are going to present a poster at the IACS Showcase early Dec
- Blog Post
  - A medium blog post to illustrate our workflow, modeling and results