Project 5: Passion Project
Cookit@Home - Recipe classification and recommendation system (tentative title)
Flora Xinru Cheng
November 20, 2019

Domain

How often have you thought "I could make that at home!" when eating out?
For me, the answer is *a lot*. I've set a rule for myself that when eating out, I don't order what I know how to cook at home. This helps me discover new recipes as well as improve cooking skills while on a budget.

My personal motivation for this data science project is to dive deeper into the domain for my Project 4. The business case is similar to that in my Project 4 proposal (https://github.com/floraxinru/metisproject04/blob/master/Project4_proposal_v2.pdf). I want to encourage people to expand their palate, as well as help people cook dishes similar to the ones they've had in restaurants (probably while travelling abroad). I would like to build a usable recipe recommendation system in the form of a web app, where the recommendation is generated based on a photo.

Design

Because this project is in the same domain of my Project 4, part of the challenge is to ensure there are still sufficient new data science problems in this project for the next few weeks. I would like to expand the scope of the project, to try my hand at new concepts and tools we learned in the past two weeks (such as CNN). But I don't want to merely using a fancy, complex model for the sake of it (especially when a simple model could give a better result).

The main new component of this project is image classification. For this task, I would likely compare a few CNN architectures, as well as use image segmentation tools such as U-net.
For the recipe recommender, I would like to learn to use models and libraries I didn't use in Project 4 (LDA, Gensim, SpaCy...), and better understand processes such as lemmatization.

The pipeline I am hoping to build will start from the photograph of a dish (which could be taken at a restaurant and uploaded to Instagram), to a classifier which outputs the type of food (for example: pasta, pizza, cake), to recipe recommendations of related dishes. I need to learn more about how to build an image classifier, and whether I need to label data in order to evaluate how good the predictions are.

I would also like to incorporate information about the cuisine type (i.e. Italian, Thai) or even the flavour profiles of the dishes into the recommendation system, which is what I didn't have time to explore in Project 4.

For the end product, I hope to make a (mobile-friendly) working Flask web app, where the user can click on (or upload?) an image, get the list of top classification results, as well as recipe recommendations in the form of links to recipes on the source websites.

I still haven't made a working Flask app yet. But because this is a data science project (not a web dev one), and I am also more interested in the machine learning and deep learning models, I don't want to spend too much time on the web app (emphasis on "working"), so I'll probably still tweak it in the final week, even after doing a few practice runs for my presentation.

If I have extra time (which I doubt), the next step would be to look into NLP libraries in other languages as well as scrape recipes from other languages, so the final product would be more helpful for foodie travellers. Also I know just enough French, German, and Spanish to understand names of food and ingredients. There's also a popular Chinese website for recipes (where I learned to cook) and I'd love to be able to use their data. But I realized this could be a separate project on its own.

Data

I found this data dump of 140K English recipes with images (https://archive.org/details/recipes-en-201706), and figured out how to extract the tar.xz files. Since recipe data is time-insensitive, I will likely use this data instead of scraping from the same websites on my own. The data was scraped from four websites in the Summer of 2017:

Contents

allrecipes.com: 91K recipes, with photos, HTML, and crawling code
epicurious.com: 34K recipes, with photos, JSON, and crawling code
bbc.co.uk: 10K recipes, with photos, HTML, and crawling code
cookstr.com: 8K recipes, with photos, HTML, and crawling code

I also have ~125000 recipes (text-only, in JSON) from recipe-box https://eightportions.com/datasets/Recipes/#fnref:1, but I'm having issues downloading its images (it's still an open issue in their repo). I would need to decide whether to use the recipe-box dataset without pictures, just for NLP. The source for the two collections

have overlaps also. But I might still decide to write or build upon existing web scraping code to get more data, as well as images that match the text I already have.
Either way, I would like to use a decent-sized (~100k recipes) dataset with both text and images, since I also need to set aside a test set for classification (as well as recommendation).

In addition, I might need to find or scrape labelled data in order to build the classifier (having a recipe and its corresponding image that is labelled with tags such as “pizza”, “Italian”). At this point I’m not sure how difficult this would be.

MVP
Because the MVP is due so soon (on Monday, Nov 25), and I expect to have the most questions with the image classification component of this project, my MVP would be a working image classifier (no recommender).

Timeline/To-do List for MVP (Week 1 of project)
Get data/Web scraping
Cleaning and Preprocessing data: image data for classification, text data for recommendation system
(Keras tutorials on image processing and CNN)
Reading popular recent papers on CNN and Computer Vision to decide which models to implement with transfer learning

Tools
Python libraries:
Python, Pandas, NumPy, SciPy, matplotlib, seaborn
Machine Learning:scikit-learn, pickle
NLP: nltk, Gensim, spaCy

Deep Learning, Computer Vision, and transfer learning:
Compare different image detection and segmentation algorithms (Mask R-CNN, RetinaNet, U-Net, etc. -- need to read more papers before diving in)

Web App: Flask, HTML, CSS, JavaScript
GitHub, cookiecutter (datasciencemvp template)
Scripting: Jupyter Notebook, Visual Studio Code
Web Scraping: BeautifulSoup, Selenium
May also include: AWS, MongoDB, SQL, Tableau