REMI- Recipe Exploration and Modification Intelligence

For the Crawler/Scraper

-Take Care of TODO: Revisit preventer, set up method to allow crawling to be done not all at once

-change from returning lists to some sort of queueing behavior

-Foodwishes

-further separate into recipe pages and non-recipe pages [revisit regex]

-for recipe pages, just scrape text and don’t add links to queue

-examine raw soup output before style removal

-for non-recipe fw pages, limit domains to those on john’s site and try to find some sort of format commonality?

-honestly just change crawler behavior to toggle between only foodwishes domain and not, breadth of sites too vast

For the File Processor

-create recipe class

Properties

-Name

-cuisineDomain

-ingredientKeywords

-ingredientBody

-methodKeywords

-methodBody

[[[IF TIME]]]

-nutritionKeywords

-nutritionBody

-ingredientHash

-instructionsHash

-linkHash

-add some kind of tag to recipe files to indicate different processing

-identify features to extract preamble, ingredients, and instruction body- investigate tags???

-

DEFINITIONS

USE WORDNET TO GET TERM DEFINITION, ALLEN\_NLP TO CONSTRUCT A WAY TO USE A TERM IN A SENTENCE

Dependency parse to use instruction statement in a sentence?

Find allrecipes link within blog posts????

“[Follow this link to get the complete, printable written recipe](https://www.allrecipes.com/recipe/277509/italian-party-bread/)! “

See <https://github.com/onzie9/all_recipes_data/blob/master/main.py> for ban evasion on AllRecipies

To increase database regarding kismet, have an ‘I know what I’m doing’ mode and extract ingredients sections for all posted recipes

Or just fuck it and nab the youtube link

REMI Pseudocode

SCRAPER/CRAWLER:

Modify behavior to crawl first adding links to queue, then scraping links in queue????

-def checknovellink

Takes in a link,

IF link is not new

creates a hash based on the string passed in, stores as key-value (link as value)

returns updated dict and TRUE

ELSE

Returns dict and FALSE

Boolean return controls whether link is added to queue

-in Crawler

-need to make error handling more specific

IF httperror, change TOR proxy and recall function with same link

ELSE skip link and move on

-refine splitting of link categories

-if grailText found in link object explicitly, not .get call

-call AllRecipesScrape

-def AllRecipesScrape

In the LINK OBJECT [[NO GET]], if grailtext

Scrape site solely for printable link, get text from printable version of the site

-how will print dialog affect this behavior????

-def foodWishScrape

-nab youtube video link

-nab all text between “Ingredients” and pinterest tag

-in Main

From Class to Chatbot

To fix ingredient storage:

Create list of measurement keywords, store ingredient as non-measurment words

POS parse???

Collocations????

Quantity conversion????

Use this to help user scale down if not enough of somethign

1. Vector Space Model- Feature relation between cooking methods and their intended effects on food, cosine similarity to
   1. Methods- broiling, grilling, roasting/baking, sautéing, poaching, simmering, boiling, steaming, braising, stewing, sous vide, air fry/convection bake
   2. Features- minimum temperature, maximum temperature, direction of heat [categorical: unidirectional, hemidirectional, omnidirectional], moisture retention/loss, browning factor, technique difficulty
2. Ingredient Substitutions with WordNet- find synset for desired ingredient substitution, get list of lemmas, [some way of determining which is the best? Or just print em all out]
3. Dependency/POS parsing- ingredient lookup in database (also add to dict entry?)
   1. Dictionary lookup-
4. User model
   1. Name
   2. Ingredients on hand
      1. Quantulum3 allow quantity tracking
   3. Preferred Cooking Techniques
   4. Recipes gathered/made
   5. Nutritional Desires

TO EXTRACT INGREDIENTS FROM USER INPUTTED TEXT, KEEP ALL NOUNS, THEN CHECK TO SEE IF THEY HAVE SOME FOOD RELATED HYPERNYM IN WORDNET

Text, letter

Description automatically generated