# Recommendation system

## General idea

using generated topic-keywords distribution by LDA model to locate user’s interested research field and recommending relevant papers to users by user input and user action. Models are trained incrementally by newly collected data

## description

* Training a LSTM model (long short-term memory neural network) to predict users’ major interests through their input text from search bar. Input will be publication abstract data with label among the topic/fields generated by LDA model, and output will be LSTM model, a multi-classifier. The prospective outcome of the model will be a series of probability representing user’s interest toward different fields
* According to their preference (“like” or “dislike” they chose is taken as feedback to our system), pretrained a LSTM model for each individual user. Getting a two-class classifier. Specifically, input of LSTM network will be paper abstract labelled by user’s “like” or “dislike” action and output will be a model that is able to distinguish user-favored papers from data
* Fetching all the papers belonging to interested field as input for personal LSTM model to determine whether papers should be recommended

## Implementation

### Training model

Generated topic keywords distribution among all papers

Input:

abstract

LSTM neural network

A multiclass classifier that can predict user’s interests through contents he/she types into search bar

Output:

model

“like” or “dislike” labelled data defined by user action

LSTM neural network

A 2-class classifier that can distinguish whether users are interested in input papers

To differentiate, we call the first gained model “general recommendation model”, and the other below “individual recommendation model”. The general commendation model is a classifier which takes plain text typed in search bar as input and produce prediction of user potential interested field/topic. After that, we fetched all the papers belonging to this category and transfer them to individual recommendation model to get recommended result.

There will be only one general recommendation model and it will be updated incrementally when database get larger and dozens of 2-class classifiers produced by for each user.

### Prediction

Retrieving corresponding papers belonging to such field

User input from search bar: plain text

Word2vec model

formalized

modelized

User input vector

Change the plain text into a vector

LSTM general Classifier

Feed the vector to the Long short-term memory neural network

Input:

vector

Predicted research field users most interested in

Output:

Fields of interests

LSTM individual preference classifier

Recommend those papers to users

Select the paper categorized as “like”

Input:

abstract

Output:

model

# Data retrieval and preprocessing

## Target

retrieving publication data (only abstract, title, author list and non-full paper information) of Mizzou faculty members periodically and incrementally from several public databases. Reconstructing retrieved data into our designed data structure stored in our project database.

## Technology

1. web crawler: python scrapy library (https://scrapy.org/)

2.public Api from IEEE, Scopus, ScienceDirect (https://dev.elsevier.com/scopus.html#/)

Faculty member list

Get faculty profile from mizzou official website

Author id and electronic id

Author search Api

Publication list information of each author

Document

Retrieval Api

Using Elsiver’s author search Api to obtain their detailed author information

Applying abstract retrieval Api to get publication abstract

Paper abstract

Upload data

Project database

General description

* Api mentioned above need to register an Api key to activate
* Mainly use http GET method to corporate with Api, and packaging several web crawler/data collectors into a scrapy project
* During author search, we applied string matching to eliminate irrelevant data record and keep tracking author id to exam whether our data are authentic
* Retrieving data -> reconstructing data -> upload data