Document Manage System with Auto Classification, Tagging

Date: 04.12.2019

Course Name: Cognitive Computing and Deep Neural Networks (INFO7374)

Professor Name: SRI KRISHNAMURTHY

Team members: Wenjun Song, Haimin Zhang, Yanjun Liu

1. Overview

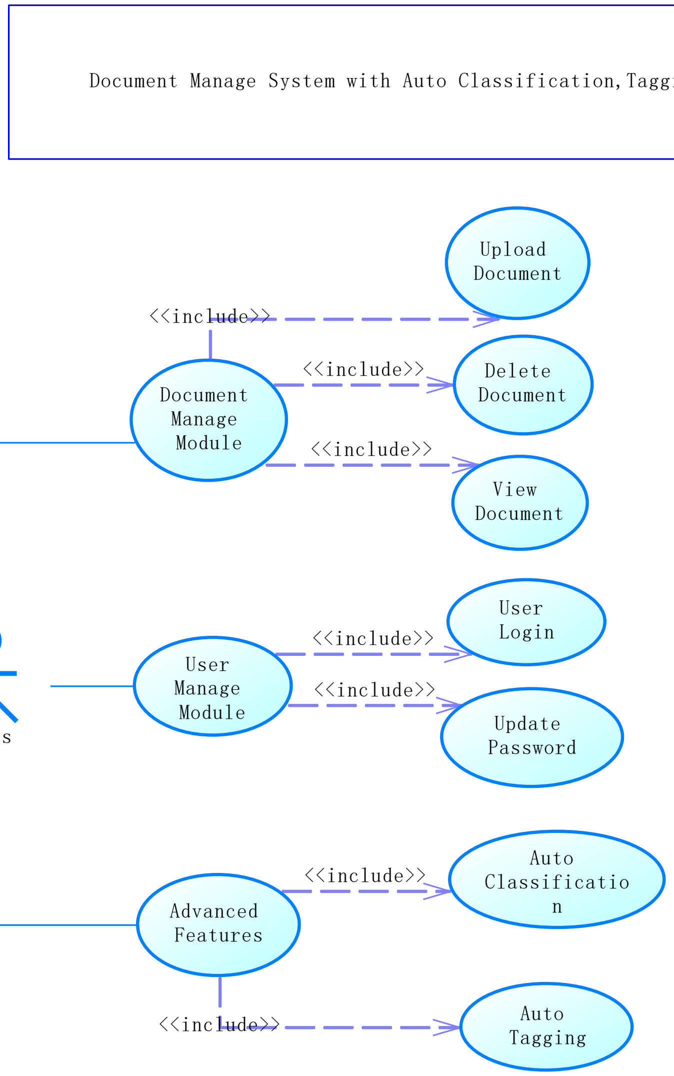
Nowadays, a lot of documents are created every day. By managing these large documents, people have to spend a lot of time on organizing and acting on document every day. Document Classification and Tagging reduces the burden of manual decision making that is done by employees by accurately and automatically organizing information. Document Classification and tagging helps organize unstructured content by analyzing the full text of documents and applying rules that automate classification decisions.

2. Goals

In order to solve problems above and enhance the efficiency of employee, we make these following goals.

1. To offer good algorithms to make good classification and tagging on documents.
2. To offer a simple web application to let people upload documents and manage documents manually
3. To offer advanced features to help people organize documents with auto classification and tagging.

3. Use Cases



3.1 Document Manage Module

Upload document: Users could upload file into system

Delete document: Users could delete file from system

View document: Users could view document online

3.2 User Manage

User Login: Users could login into CRMS with user name and password

Update Password: Users could update their password

3.3 Advanced Features

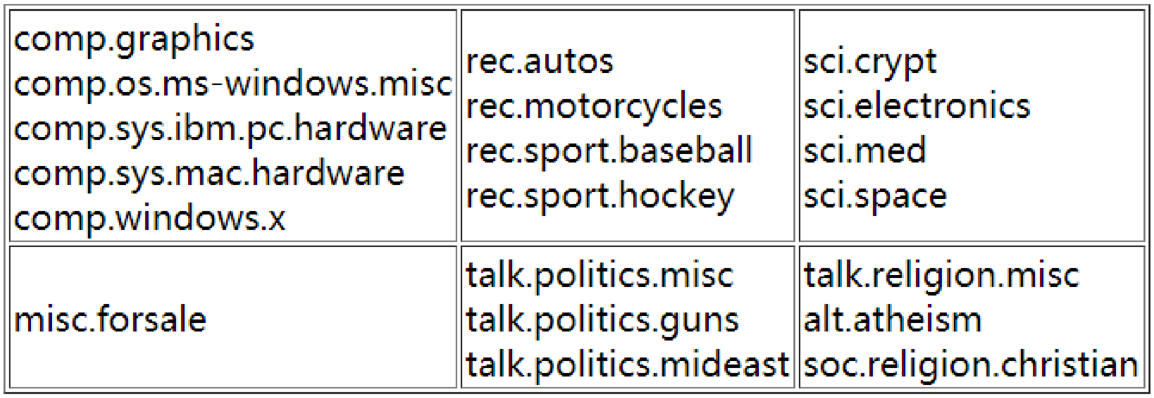
Auto Classification: Users could click classify button and documents would be separated into different types of folders.

Auto Tagging: Users could click tagging button and documents would be tagged with related key words.

4. Data

4.1 The 20 Newsgroups data set

The 20 Newsgroups data set is a collection of approximately 20,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups. The 20 newsgroups collection has become a popular data set for experiments in text applications of machine learning techniques, such as text classification and text clustering. The data is organized into 20 different newsgroups, each corresponding to a different topic. Here is a list of the 20 newsgroups, partitioned (more or less) according to subject matter



comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

4.2 AG's corpus of news articles

AG is a collection of more than 1 million news articles. News articles have been gathered from more than 2000 news sources by ComeToMyHead in more than 1 year of activity. The dataset has four categories: World, Business, Sports, SciTech.

4.3 Reuters-21578 Text Categorization Collection

This is a collection of documents that appeared on Reuters newswire in 1987. The documents were assembled and indexed with categories.

4.4 BBC Datasets

Two news article datasets, originating from BBC News, provided for use as benchmarks for machine learning research.

It Consists of 2225 documents from the BBC news website corresponding to stories in five topical areas from 2004-2005.

Its class labels are business, entertainment, politics, sport, tech

5. Process Outline

1. Collect related papers and make a research on auto classification and tagging

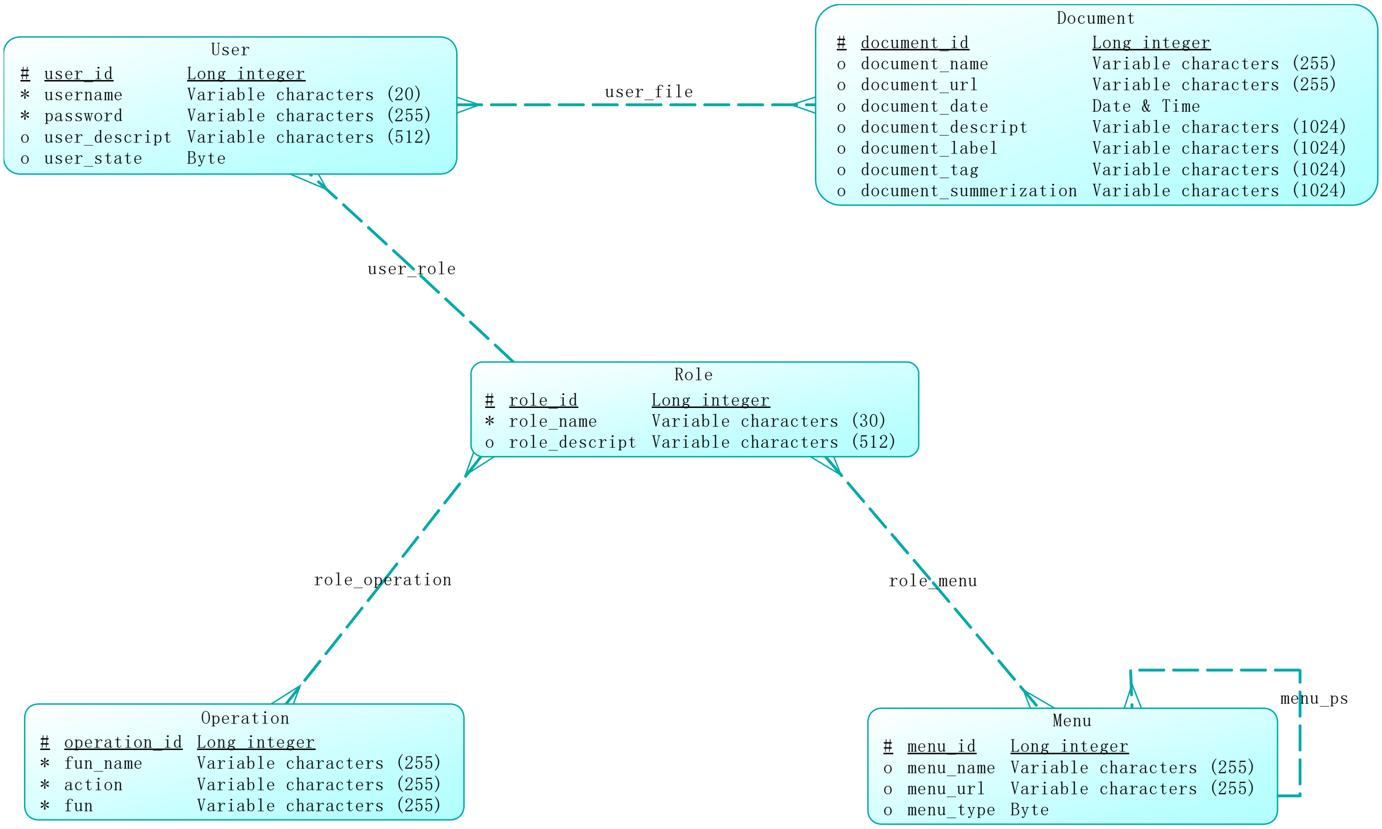
2. Data Preprocessing

3. Study of Supervised approaches and select the best model for prediction

4. Design of a pipeline and system to implement this approach and discussion on the system's capabilities

5. Build a web application to manually and automatically manage documents

6. Database Design



User

#	user_id	Long integer
*	username	Variable characters (20)
*	password	Variable characters (255)
o	user_descript	Variable characters (512)
o	user_state	Byte

Document

#	document_id	Long integer
o	document_name	Variable characters (255)
o	document_url	Variable characters (255)
o	document_date	Date & Time
o	document_descript	Variable characters (1024)
o	document_label	Variable characters (1024)
o	document_tag	Variable characters (1024)
o	document_summerization	Variable characters (1024)

Role

#	role_id	Long integer
*	role_name	Variable characters (30)
o	role_descript	Variable characters (512)

Operation

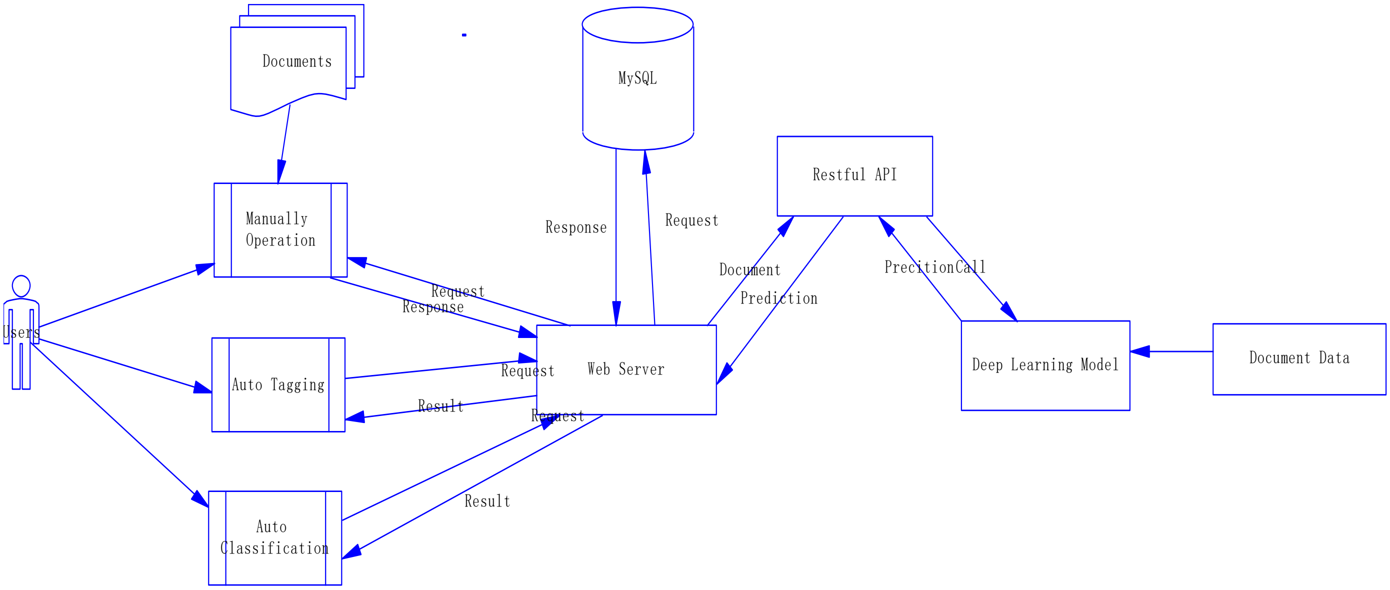
#	operation_id	Long integer
*	fun_name	Variable characters (255)
*	action	Variable characters (255)
*	fun	Variable characters (255)

Menu

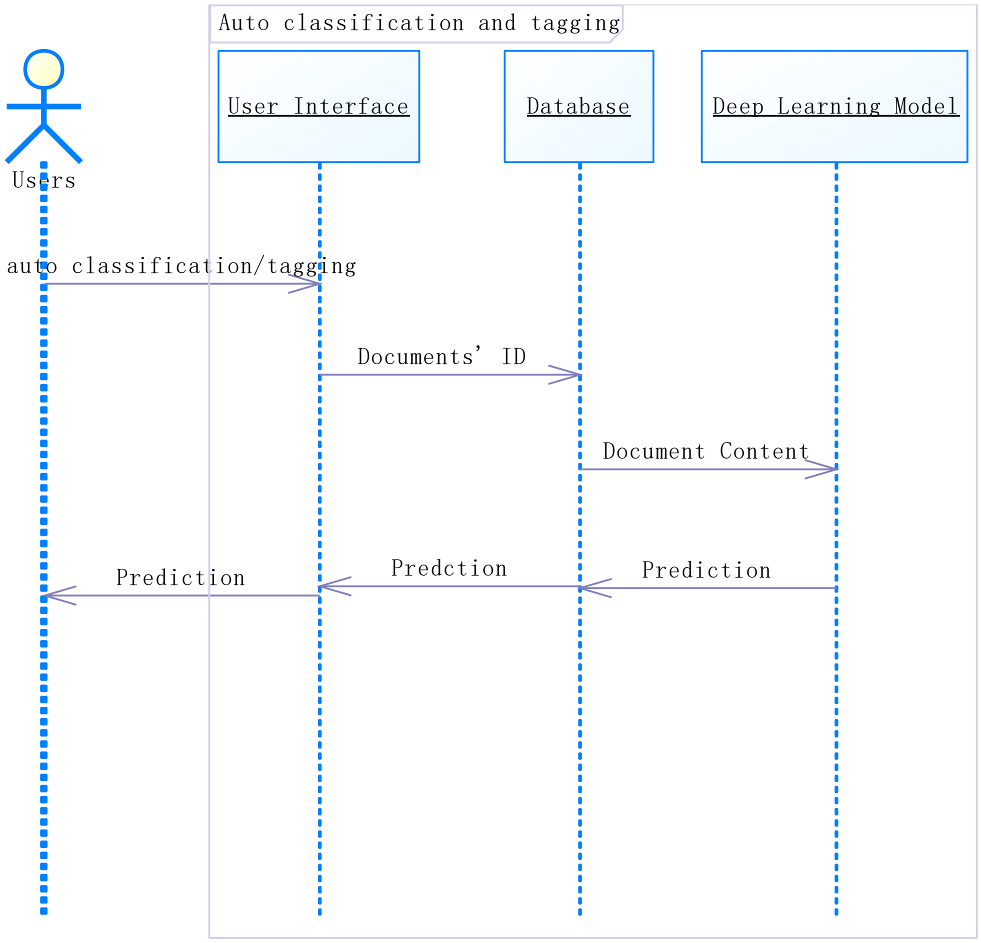
#	menu_id	Long integer
o	menu_name	Variable characters (255)
o	menu_url	Variable characters (255)
o	menu_type	Byte

7. System Design

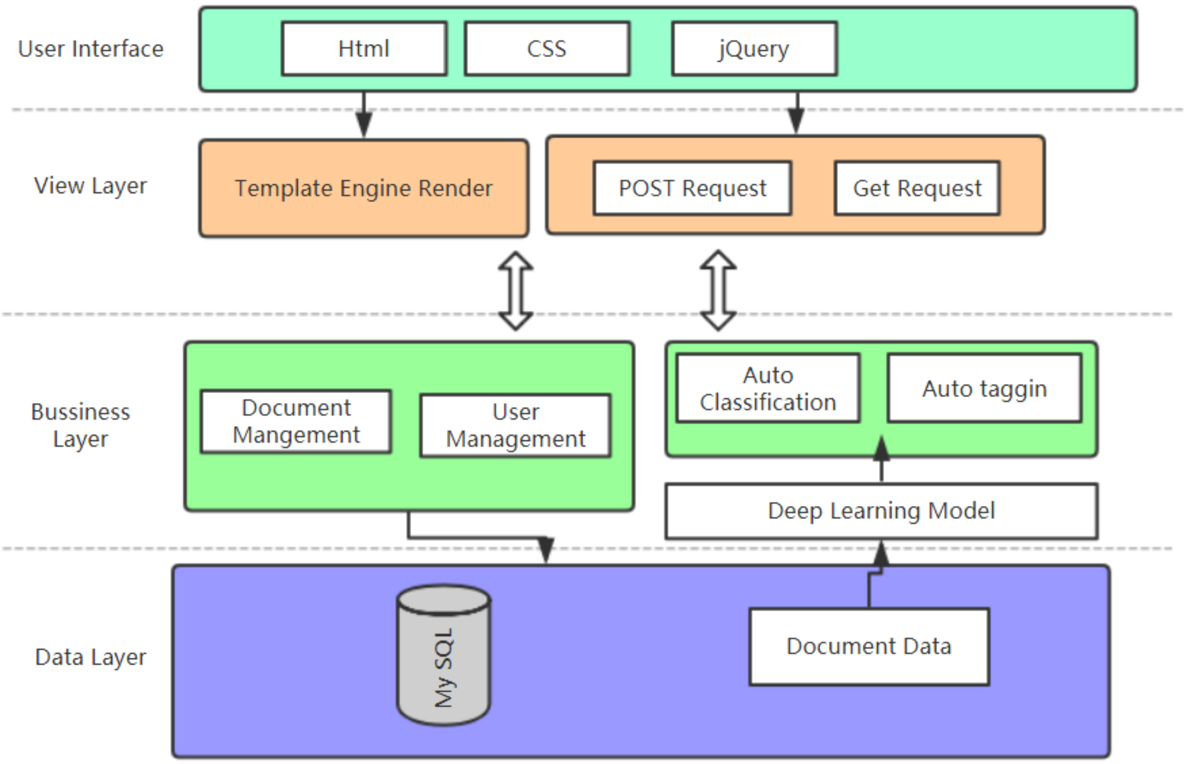
7.1 Workflow



7.2 Sequence Diagram


Auto classification and tagging
Users
User Interface
Database
Deep Learning Model
auto classification/tagging
Documents' ID
Document Content
Prediction
Predction
Prediction

7.3 System Architecture



8. Models

8.1 Auto-Classification model

- Using Word2Vec model as Embeding.
- Attached with Dense Layer as Classifier
- Accuracy at 85.3%
- Parameter setting

  The Auto-Classification model base on the assumption that the distribution of new incoming Docs should be same as Our training set.

  The more training data, the more true accuracy of the prediction (merged 20 News data and BBC data)

8.2 Auto-Tagging model

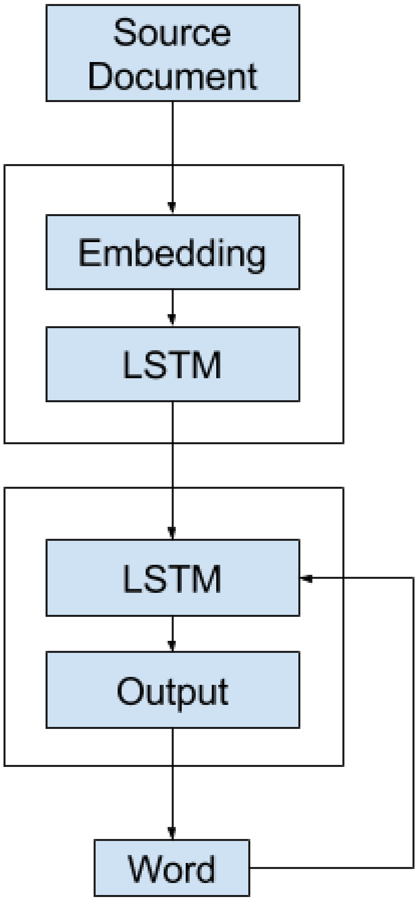
- Normalized Word Freq as Weight of words:

Function:

The Weight = Freq(words) / Number (doc with such words)

The first 5 words with high weight are chosen to be Tags.

8.3 Doc Summarization model



There are two broad approaches to summarization: extractive and abstractive. Extractive methods assemble is taken directly from the source text; abstractive methods generates out new meanings from words and phrases– as human being abstract usually does. The extractive approach is easier, however, some high-quality summarization, such as paraphrasing, generalization, or the incorporation of real-world knowledge, are possible only in an abstractive method. Our model refer to the latter one – abstractive method. (Later I will present a interesting output to show the concept of abstractive method, which need a robust dataset to train the model, or else the output will be weird.)

We use the *CNN/Daily Mail* dataset (Nallapati et al., 2016), which contains online news articles (781 tokens on average) paired with multi-sentence summaries (3.75 sentences or 56 tokens on average). We used scripts supplied by Nallapati et al. (2016) to obtain the same version of the the data, which has 287,226 training pairs, 13,368 validation pairs and 11,490 test pairs.

We cited the model from *Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In Neural Information Processing Systems*. And the output of Assignment 4 from Team #3. We use RNN to train our summarization models, which can both read and freely generate text to abstract summarizations from documents or news.

After loading data, we firstly encode the texts(fit_text), Then we did the pad_sequence steps to make them at the same length.

9. Milestones

Timeframe	Delivery
Day 1-2 (4.13~4.14)	Data Preprocessing and Exploratory Data Analysis, Read research paper
Day 3-11(4.15~4.24)	A. Model Building, Training, Selection(Wenjun Song Yanjun Liu) B. Deployment of models on cloud and build web application(Haimin Zhang)
Day 12-13 (4.24~4.25)	System integration and documentation

10. Reference and Sources

Datasets:

http://qwone.com/~jason/20Newsgroups/

http://mlg.ucd.ie/datasets/bbc.html

http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html

Reference:

Distributed Representations of Sentences and Documents Q Le, T Mikolov - International conference on machine learning, 2014 - jmlr.org