11791 Design and Engineering of Intelligent Information Systems

Assignment 2 Report

Name: LIU Xi

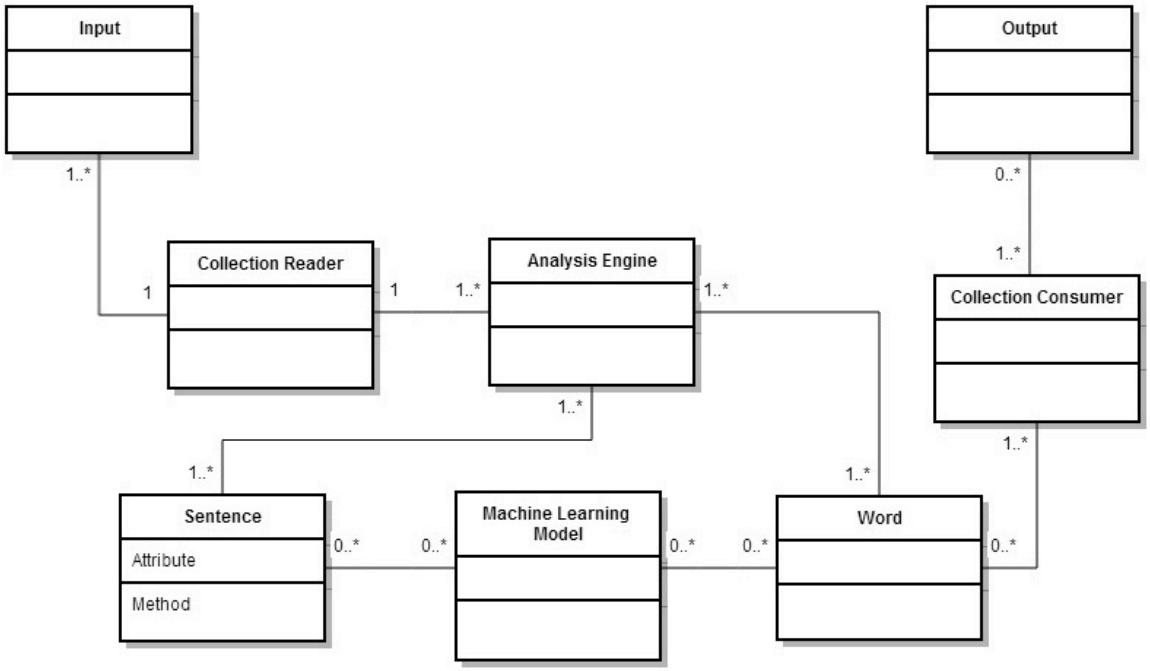
Andrew ID: xiliu1

Date: 10-08-2014

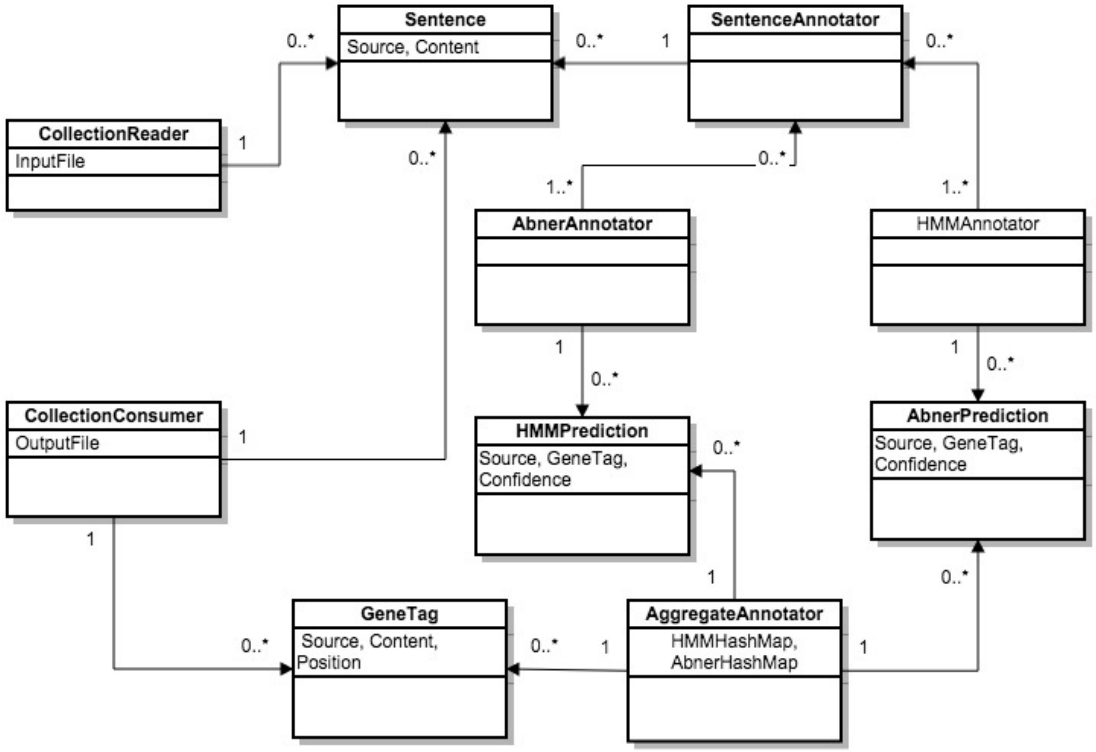
Contents

1. System Overview

Domain Model



UML Design



2. External Resources

Machine Learning Techniques

Hidden Markov Model

HMM is a simplest dynamic Bayesian network, where the state is not hidden while the output relying on the state is visible. For each possible output tokens, the states have probability distributions, so that the sequence of tokens shows the sequence of states in HMM. HMM is widely used for natural language recognition, and also solve the bio-informatics tagging problems in a linguistic approach.
Lingpipe trains the HMM used in this project with cross-validation, and achieved high precision. Different from the previous assignment, this model is retrained by additional 5000 instances. Larger dataset and cross-validation could reduce the level of over-fitting.

NLPBA and BioCreative Model

ABNER combined NLPBA and BioCreative model to perform prediction, and achieved relatively high precision on biological data, especially on DNA, RNA, and protein entity recognition.

Tools

Toolkit 1: LingPipe
LingPipe is toolkit for processing text using computational linguistics with Java.
LingPipe Source Code: GeneTag Corpus

Toolkit 2: ABNER
ABNER is toolkit for entity recognition on biological data, especially DNA, RNA, Protein tags. It provides Java API.

Dataset: NCBI GeneTag
Produced by the Unites States National Center for Biotechnology Information (NCBI),
Related paper: Tanabe, L., N. Xie L. H. Thom, W. Matten, And W. J. Wilbur. 2004. GENETAG: a tagged corpus for gene/protein named entity recognition. BMC Bioinformatics 2005, 6(Suppl 1):S3.
Download path: ftp://ftp.ncbi.nlm.nih.gov/pub/lsmith/MedTag/medtag.tar.gz

3. Strategies for Post-prediction

Post Process on ABNER Prediction

Regular Expression

ABNER does not achieve the high precision as described on its documentations, so I did a manual check on its output and found there are an obvious error. That is, it contains some irrelevant characters such as “(”, “ ”, “{”, and “&”, etc. So a regular expression pattern `"[0-9a-zA-Z-\\s]+"` should be applied to eliminate such predictions before combine the prediction results from two models.

Strategies to Combine Prediction Sets

Voting versus Weighting

There are two major approaches to combine the predictions from multiple models, voting and weighting. Weighting requires detailed analysis on the performance and strength on each model, then adjust the weighting of each model, to calculate the overall confidence.

Voting is chosen for this project for two reasons. Firstly, due to the limitation of time and resources, only two models were trained with limited datasets, and roughly evaluated. Secondly, ABNER does not provide an API for confidence on each prediction, so it is difficult to determine if an individual prediction is strongly supported. Therefore, the two model could not be fully evaluated or assigned an interpretable weighting.

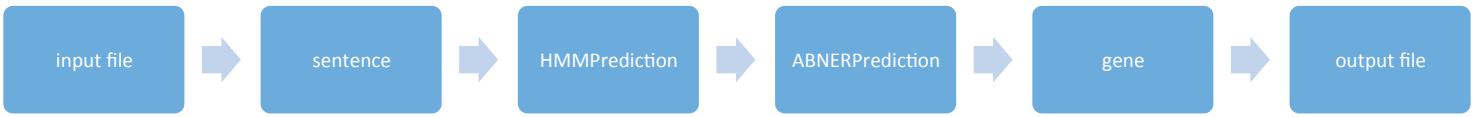
The voting strategy follows the rules below.

Rule 1. If both HMM and ABNER agree on the prediction, the prediction is selected by the final result.

Rule 2. If HMM predicts a tag with a high confidence (>=0.6) while ABNER disagree, the prediction is still selected.

Rule 3. If ABNER predicts a tag while HMM disagree, the prediction is neglected since the confidence is not provided.

4. General Data Flow



Scenarios:

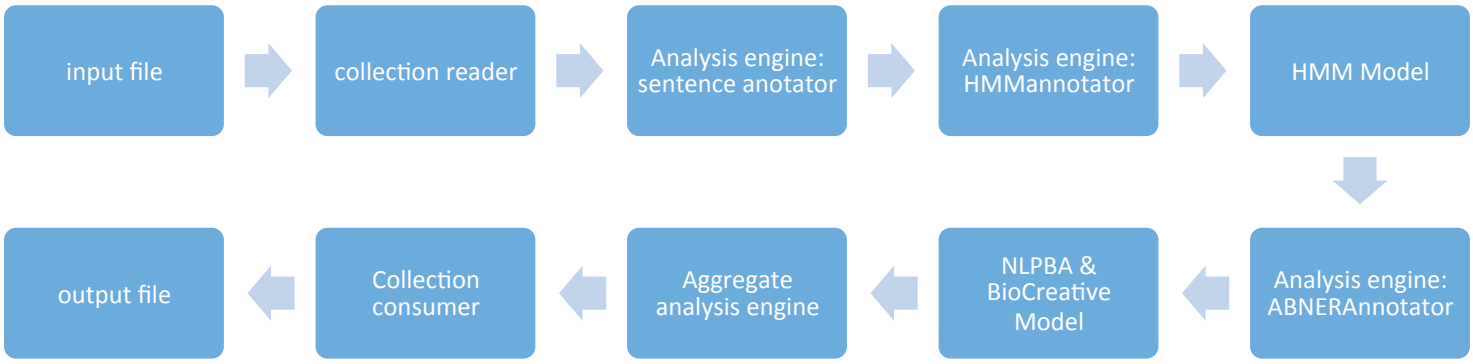
Step 1. Load the input and break the input file by sentences

Step 2. Break the sentences by words

Step 3. Predict if the word is a gene tag with HMM

Step 4. Predict if the word is a gene tag with ABNER
Step 5. HMM and ABNER vote for the final prediction as gene
Step 6. Calculate the position of gene tag and output

5. NER Pipeline Design



6. NER System Design

Type System

Sentence

Source: SentenceID
Content: Sentence content

GeneTag

Source: SentenceID
Tag: gene tag
Confidence: confidence for prediction

HMMPrediction

Source: SentenceID
Tag: gene tag

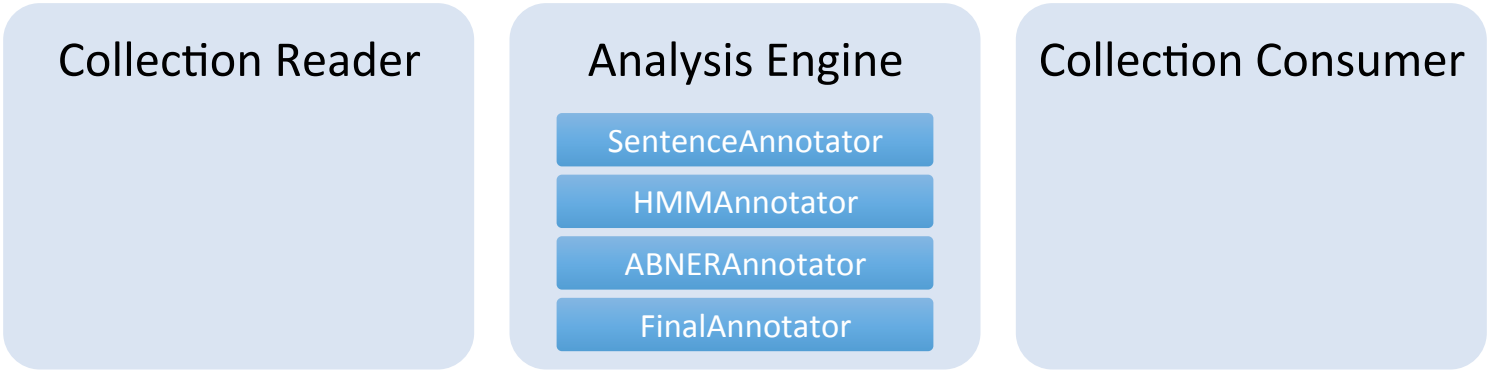
Confidence: confidence for prediction

ABNERPrediction

Source: SentenceID
Tag: gene tag
Confidence: confidence for prediction (ABNER does not provide the API for confidence, so all confidences are set to 0.5 by default)

Collection Processing Engine

Collection Processing Engine implements the pipeline design. The function of each parts are described later.



Collection Reader

Scenarios:
Step 1. Load the input file.
Step 2. Break the input file by sentences, and record the absolute position of each sentence, and feed the sentence annotator.

Analysis Engines

There are two annotators in total, and both read and produce JCAS data.

Sentence Annotator

Scenarios:
Step 1. Read the sentences, and break the sentence into words, and record the absolute position of each word,
Step 2. Feed the word to HMM model.

HMM Annotator

Scenarios:
Step 1. Read the Sentence
Step 2. Apply HMM model to find the gene tag.
Step 3. Store the predicted gene tag to HMMPrediction.

ABNER Annotator

Scenarios:
Step 1. Read the Sentence
Step 2. Apply NLPBA and BioCreative model to find the gene tag.
Step 3. Apply regular expression to eliminate obvious errors.
Step 3. Store the predicted gene tag to ABNERPrediction.

Final Annotator

Scenarios:
Step 1. Read the gene tags predicted by HMM and ABNER.
Step 2. Apply voting strategy to select the
Step 3. Feed the predicted gene tag to ABNERPrediction.

CAS Consumer

The CAS Consumer read JCAS data from Gene and output the results to the output file.

7. Reflections

Precision and Recall

Precision: 0.80605202619
Recall: 0.748152203668
F1 Score: 0.776023624283

The sample built by Stanford-NLP only recognize the nouns, so the precision, around 10% is very low.
The proposed LingPipe solution implements the HMM trained on the gene tags datasets, which is more capable for gene tag recognition and achieved the precision at around 80%.
The proposed ABNER implemented NLPBA and BioCreative, and achieved higher performance than Stanford-NLP. However, its precision and recall is much lower than HMM, and it extracts fewer gene tags than lingpipe.

The overall precision and recall are influenced by the post-prediction strategies. After eliminating the obviously incorrect predictions, ABNER achieved better precision. The voting strategy firstly combines the predictions made by HMM and ABNER and creates a larger set of predictions, meanwhile the predictions with high confidence is guaranteed to be selected. Because the total number of correct predictions increased, the recall is improved too.

Problems

There are three major problems.
Firstly, due to the limitation of resources, the dataset is relatively small so self-testing may happen.
Secondly, only two models were trained with limited datasets, and roughly evaluated.
Thirdly, ABNER does not provide an API for confidence on each prediction, so it is difficult to determine if an individual prediction is strongly supported.

Further Enhancements

The following aspects could further enhance the project.
Aspect 1. Try more machine learning models with more evidence to support decision.
Aspect 2. Train current models with larger dataset to avoid over-fitting and self-testing.
Aspect 3. Improve the performance of current HMM and ABNER.
Aspect 4. Introduce weighting strategy when there are more models with confidence.