Extracting keyphrases and finding relations from scientific publications (research papers).

The akatsuki of Lolis

Team Number = 11

- Priyansh Gupta (2019101080)
- Harshit Sharma (2019101083)
- Bhavya Jain (2019101095)

Github repo link for cloning used datasets and saved models:
https://github.com/xLeviackermanX/intro-to-nlp-project.git

Given Scientific Documents How to extract keyphrases?

How to do this task automatically?

Tip

Sometimes things which don't look impressive may contain hidden information that may be helpful in beautifying itself after a little bit of efforts.

First approach/idea

We can first make candidate set of keyphrases by storing all 1-6 grams. Then we can try to come up with some score mechanism which will rank these phrases and we can then say that top 10 or 15 ranked phrases are my keyphrases.

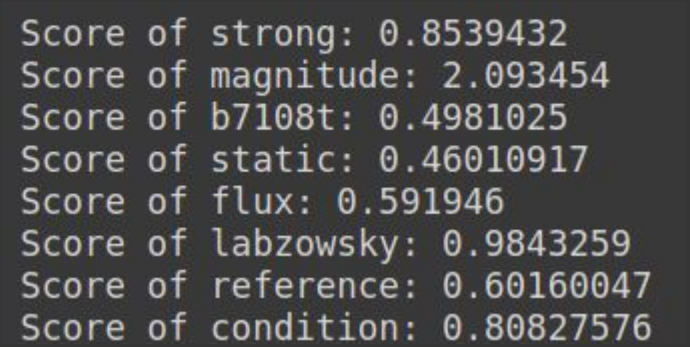
Implementation of this rank approach

Firstly preprocess the text, tokenize it by using lemmatization. Remove all the candidates with stopwords only. Create a word vocabulary for this document.

Graph construction

Now construct a graph containing each word of vocab as it's nodes and there are weighted undirected edges between these words. These edge weights depends upon how much the connected two words appear in document at what distance.

Find Scores



```
Score of strong: 0.8539432
Score of magnitude: 2.093454
Score of b7108t: 0.4981025
Score of static: 0.46010917
Score of flux: 0.591946
Score of labzowsky: 0.9843259
Score of reference: 0.60160047
Score of condition: 0.80827576
```

Now we can find score for each node(word) using a method similar to finding page rank. So we will iteratively find scores of each word in doc.

Remember to make scores of stopwords zero after finding convergence.

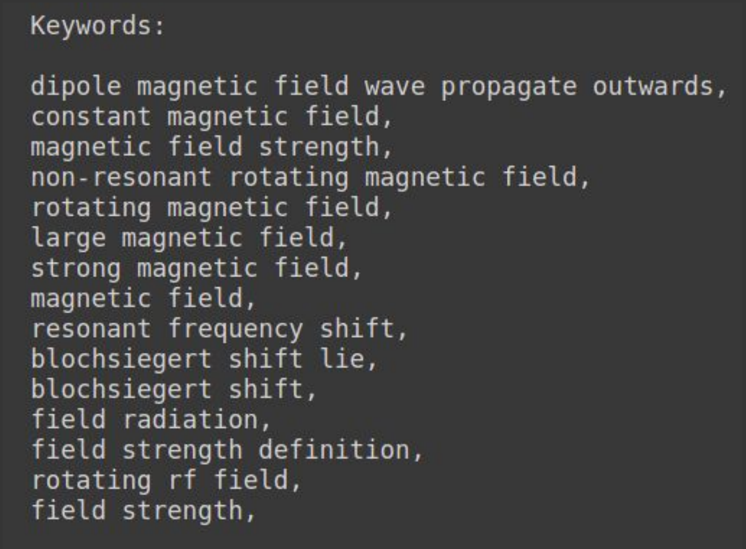
Now rank them

Now for each phrase in candidate set the score of phrase is defined as sum of score of all tokens(words) that appear in the phrase. Now sort all the phrases based on their score and take top 10 or 15 phrases.

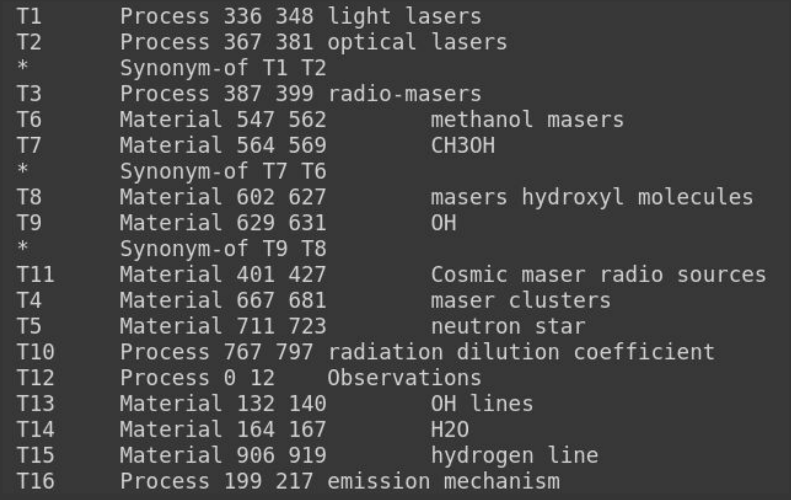
Example output

Predicted:

Actual:



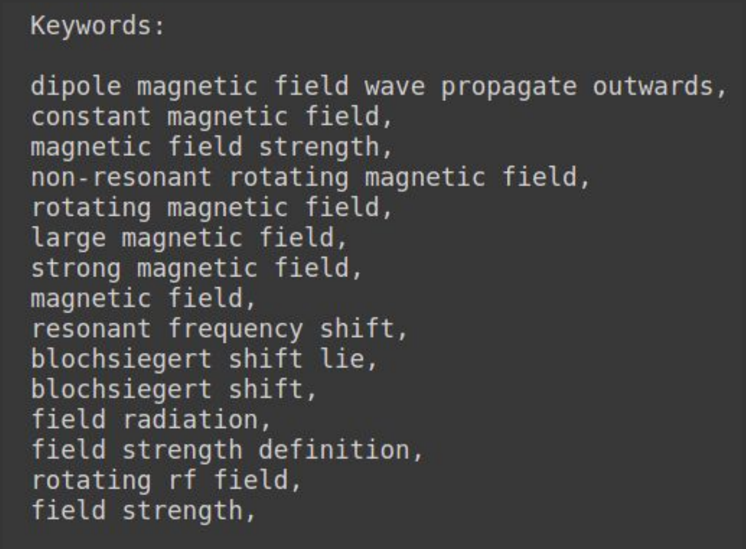
```
Keywords:

dipole magnetic field wave propagate outwards,
constant magnetic field,
magnetic field strength,
non-resonant rotating magnetic field,
rotating magnetic field,
large magnetic field,
strong magnetic field,
magnetic field,
resonant frequency shift,
blochsiegert shift lie,
blochsiegert shift,
field radiation,
field strength definition,
rotating rf field,
field strength,
```



```
T1      Process 336 348 light lasers
T2      Process 367 381 optical lasers
*       Synonym-of T1 T2
T3      Process 387 399 radio-masers
T6      Material 547 562        methanol masers
T7      Material 564 569        CH3OH
*       Synonym-of T7 T6
T8      Material 602 627        masers hydroxyl molecules
T9      Material 629 631        OH
*       Synonym-of T9 T8
T11     Material 401 427        Cosmic maser radio sources
T4      Material 667 681        maser clusters
T5      Material 711 723        neutron star
T10     Process 767 797 radiation dilution coefficient
T12     Process 0 12    Observations
T13     Material 132 140        OH lines
T14     Material 164 167        H2O
T15     Material 906 919        hydrogen line
T16     Process 199 217 emission mechanism
```

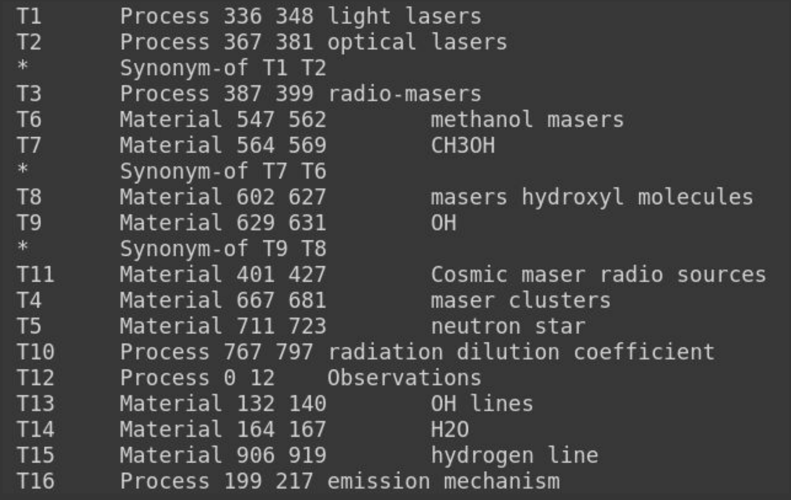
Example output

Predicted:



```
Keywords:

dipole magnetic field wave propagate outwards,
constant magnetic field,
magnetic field strength,
non-resonant rotating magnetic field,
rotating magnetic field,
large magnetic field,
strong magnetic field,
magnetic field,
resonant frequency shift,
blochsiegert shift lie,
blochsiegert shift,
field radiation,
field strength definition,
rotating rf field,
field strength,
```

Actual:



```
T1      Process 336 348 light lasers
T2      Process 367 381 optical lasers
*       Synonym-of T1 T2
T3      Process 387 399 radio-masers
T6      Material 547 562        methanol masers
T7      Material 564 569        CH3OH
*       Synonym-of T7 T6
T8      Material 602 627        masers hydroxyl molecules
T9      Material 629 631        OH
*       Synonym-of T9 T8
T11     Material 401 427        Cosmic maser radio sources
T4      Material 667 681        maser clusters
T5      Material 711 723        neutron star
T10     Process 767 797 radiation dilution coefficient
T12     Process 0 12    Observations
T13     Material 132 140        OH lines
T14     Material 164 167        H2O
T15     Material 906 919        hydrogen line
T16     Process 199 217 emission mechanism
```

Second Approach(model training for binary classification)

Extract features out of candidate keyphrases. These features are:

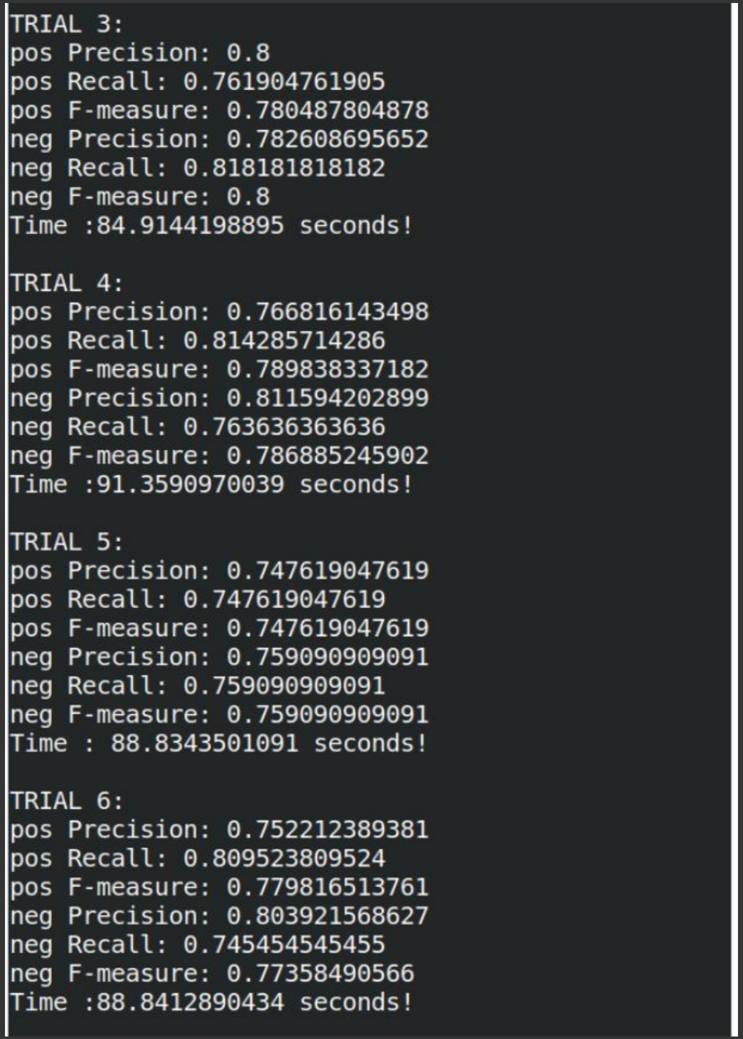
- Standard deviation
- frequency
- length
- line_position
- parabolic_position
- part_of_speech
- nth position_list

Classification

Now we trained a model for binary classification using SVM and logistic regression.

It was important to take positive and negative samples in equal ratio in the training set.

Results (wiki20 dataset)



```
TRIAL 3:
pos Precision: 0.8
pos Recall: 0.761904761905
pos F-measure: 0.780487804878
neg Precision: 0.782608695652
neg Recall: 0.818181818182
neg F-measure: 0.8
Time :84.9144198895 seconds!

TRIAL 4:
pos Precision: 0.766816143498
pos Recall: 0.814285714286
pos F-measure: 0.789838337182
neg Precision: 0.811594202899
neg Recall: 0.763636363636
neg F-measure: 0.786885245902
Time :91.3590970039 seconds!

TRIAL 5:
pos Precision: 0.747619047619
pos Recall: 0.747619047619
pos F-measure: 0.747619047619
neg Precision: 0.759090909091
neg Recall: 0.759090909091
neg F-measure: 0.759090909091
Time : 88.8343501091 seconds!

TRIAL 6:
pos Precision: 0.752212389381
pos Recall: 0.809523809524
pos F-measure: 0.779816513761
neg Precision: 0.803921568627
neg Recall: 0.745454545455
neg F-measure: 0.77358490566
Time :88.8412890434 seconds!
```

Score Metrics that we used:

- Accuracy
- Negative and positive precision
- Positive and negative Recall
- Positive and negative F1 score

Classification of Keyphrases into 3 classes: {Task, Material, Process}

Tip

Don't wait till the end of the presentation to give the bottom line.

Reveal your product or idea (in this case a translation app) up front.

Dataset of Semeval:

The dataset contains .xml and .ann files. Two scripts were provided along with the dataset to read from xml files and convert them into text passages.

.ann files contained keyphrases and their type.

Link to dataset: - github repo

Input:

We took 4 tokens of keyphrase, 4 tokens from the left context and 4 tokens from right context.

So the input sequence was of length 12 tokens.

Link to dataset: - github repo

Embedding:

We initialized word embeddings with glove embeddings and then later trained our own model to fine tune them.

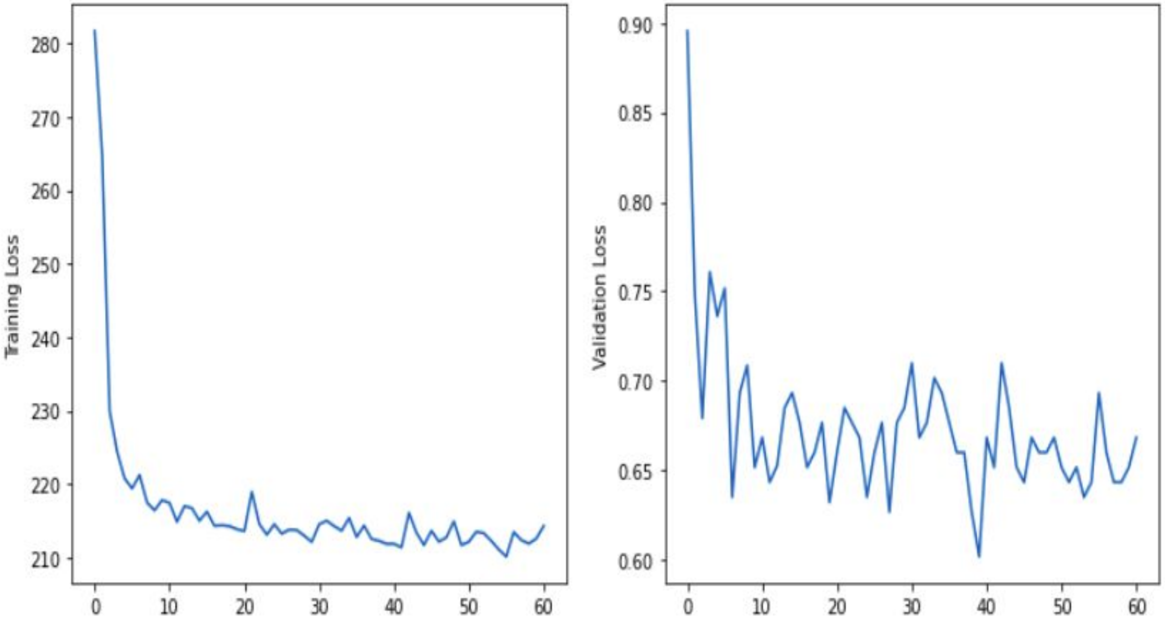
Embedding size = 100

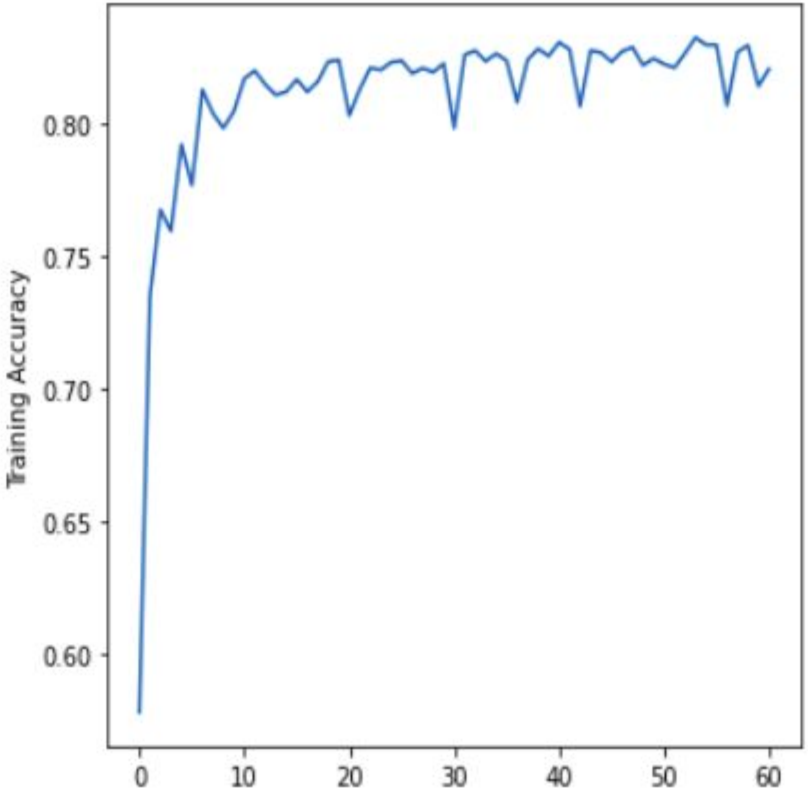
Model:

First had embedding layer, followed by use of 2 CNN layers of 64 filters and different kernel size. The output of two layers were concatenated and then sent to LSTM layer followed by 2 FCN layers performing classification into

three classes.

Training Loss Curves

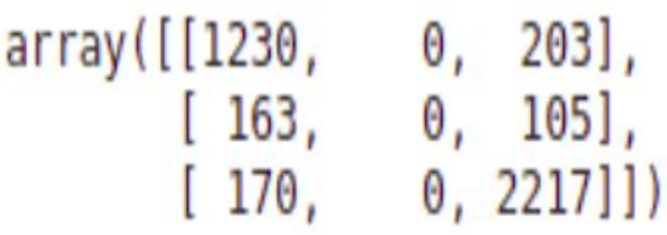




Accuracy curve

F1 scores and Confusion Matrix

F1-score on test data for keyphrase classification is 0.8431996086105675.



```
array([[1230,    0,  203],
       [ 163,    0,  105],
       [ 170,    0, 2217]])
```

Why classification for class Task fails?

we observed that in the training data, distribution of Material, Process and Task

keyphrases was 40, 44 and 16 percent respectively, showing that class Task is underrepresented in data.

Thank You!