Machine Learning Approaches for Catchphrase Extraction in Legal Documents

Tshepho Koboyatshwene
University of Botswana
Gaborone, Botswana
tshepho.koboyatshwene@mopipi.ub.bw

Moemedi Lefoane
University of Botswana
Gaborone, Botswana
moemedi.lefoane@mopipi.ub.bw

Lakshmi Narasimhan
University of Botswana
Gaborone, Botswana
lakshmi.narasimhan@mopipi.ub.bw

ABSTRACT

The purpose of this research was to automatically extract catchphrases given a set of Legal documents. For this task, our focus was mainly on the Machine learning approaches: a comparative approach was used between the unsupervised and supervised approaches. The idea was to compare the different approaches to see which one of the two was comparatively better for automatic catchphrase extraction given a dataset of Legal documents. To perform this, two open source text mining software were used; one for the unsupervised approach while another one was used for the supervised approach. We then fine tuned some parameters for each tool before extracting catchphrases. The training dataset was used when fine tuning parameters in order to find optimal parameters that were then used for generating the final catchphrases. Different metrics were used to evaluate the results. We used the most common measures in Information Extraction which include Precision and Recall and the results from the two Machine learning approaches were compared. In general our results showed that the supervised approach performed far much better than the unsupervised approach.

KEYWORDS

Catchphrase extraction, Legal domain, IRLeD

1 INTRODUCTION

Automatic keyword or catch phrase extraction is an area of research that seems like it has not been exploited much. Determining catchphrases manually can be time consuming, expensive and usually require expertise to perform the work [1], this therefore has motivated research towards automatic keyword extraction. There are different terminologies used to define terms that represent the most relevant or useful information contained in a document such as: key phrases, key segments, key terms and keywords [1]. In the FIRE2017 Information Retrieval from Legal Documents (IRLeD) task [2], the word "catchphrase" is used instead of a keyword or key phrase in the Legal domain.

Keyword Extraction involves automatically searching and identifying keywords within a document that best describes the subject of the document [1, 6]. Methods used for automatic keyword extraction can be classified into different approaches. According to Beliga et al and Lima et al [1, 6], the methods can use Simple statistical approaches, Linguistics approaches, Machine Learning approaches among others.

As the name suggests, Simple statistical approaches are very simple, they do not need any training and are language and domain independent. Keywords can be identified by using statistics of the word such as word frequency, word co-occurrences, term frequency-inverse document frequency (TF-IDF), N-gram statistics. The disadvantage with using this approach is that in some domains such as health and medical, the most important keyword may appear only once [1, 6]. Linguistic approaches looks at linguistic features of words, sentence and document such as lexical, syntactic structure and semantic analysis [1, 6]. Machine Learning approaches consists of both unsupervised and supervised. See Section 2. Other approaches consist of a combination of the methods described above and could also incorporate heuristic knowledge such as the position, the length, the layout feature of terms [1, 6].

This paper is organized as follows: We first presented related work for the supervised and unsupervised Machine learning approaches mainly focusing on Rapid Automatic Keyword Extraction, RAKE [5] and Multi-purpose automatic topic indexing, MAUI [4], followed by the approach we suggested which included all the experimental setups performed. Thirdly we outlined a brief overview of measures used for evaluating the results. We then presented and discussed the results. Lastly we concluded and briefly talked about possible future work.

2 RELATED WORK

According to Lima et al and Rose et al [5, 6], RAKE is an unsupervised Machine Learning approach which does not require any training and works by first selecting candidates keywords. Lima et al and Rose et al [5, 6] outlined RAKE's input parameters consisting of a stop list, a set of phrase delimiters, and a set of word delimiters. Firstly, the document is partitioned into candidate keywords using the phrase and word delimiters. After the selection of candidate keywords a graph of word co-occurrences is then created. Each candidate keywords is then assigned a score. Several metrics were used to calculate the score namely: word frequency, $freq(w)$, word degree, $deg(w)$ and the ratio of word degree to word frequency defined as $[ratio = \frac{deg(w)}{freq(w)}]$. Candidate keywords are then ranked starting with the highest.

According to Medelyan [4] MAUI was build based on four open-source software components: the Keyphrase extraction algorithm (Kea) used for phrase filtering and computing n-gram extractions, Weka used for creating topic indexing models and applying them to new documents, Jena used for incorporating controlled vocabularies coming from external sources and Wikipedia Miner used for accessing Wikipedia data. The four open-source software are used together with other classes to form a single topic indexing algorithm used to generate candidate topics, to compute their features, to build the topic indexing model and to apply the model to new document [4]. To create a model, a training dataset with known keyphrases is required. The only keyphrases that would then be classified will be the ones that have already been incorporated in the training data. Candidate phrases are selected in three steps namely: cleaning of input, phrase identification and lastly case-folding and stemming [7]. MAUI has a parameter that can be varied in order to control the size of the training set. Some candidates catchphrases are discarded based on their frequency of occurrence before creating a model. This will therefore reduce the size of the model [4].

3 PROPOSED APPROACH

A keyword extraction library called RAKE [5] was used for the unsupervised approach while MAUI [4] was used for the supervised approach. RAKE [5] and MAUI [4] consisted of parameters that were fine tuned before generating catchphrases. The approach used in this research was to set RAKE and MAUI parameters to different values. Then use part of the training dataset with known catchphrases for evaluation. The results of each approach were evaluated individually in order to determine optimal parameters that would be used for extracting catchphrases on the testing data. We then generated the final catchphrases using the testing data provided and the optimal parameters that yielded better results on each approach.

3.1 Experimental Setup

3.2 Dataset

For the IRLeD task, the dataset provided contained the following:

(1) Train docs - consisted of 100 case statements.
(2) Train catches - contained the gold standard catchwords for each of the 100 case statements provided in the Train docs.
(3) Test docs - contained 300 test case statements. For each of these 300 statements, a set of catchphrases was generated.

The training dataset was randomly divided into two groups consisting of 90 documents and 10 documents from the dataset. The 90 documents dataset was only used for training the supervised machine learning approach while the remaining 10 documents dataset were used for testing both the unsupervised and supervised methodologies.

3.3 Experiment 1 - RAKE parameter tuning on training dataset

RAKE consisted of the following parameters which were fine tuned for different experiments in order to find the optimal parameter values that yielded the best performance on the training set provided. Table 1 provides more details on parameters experimented with as well as performance results.

(1) The number of character can be varied in order to select keywords with a certain number of characters represented as No of Char/word in Table 1.
(2) The number of phrases for each keyword can be tuned to varies words represented as No of word/phrase in Table 1
(3) The number of times a keyword appears in a given text can be limited to a certain number represented as keyword frequency in Table 1.

3.4 Experiment 2 - MAUI parameter tuning on training dataset

As it was done in Section 3.3, parameter turning experiments were performed in order to find the optimal parameters for MAUI. The only parameter tuned for MAUI was to vary the frequency of occurrence of each keyword and discard some keywords based on that. The default MAUI parameter discards any candidate phrase(s) that appeared less than two times. See Table 2.

3.5 Final Run 1: Using RAKE

RAKE was used to generate catchphrases for the Test documents provided with parameters tuned to 3 3 1: meaning each word had atleast 3 characters, each phrase had at most 3 words and each keyword appeared in the text at least once.
UBIRLeD_1 - Catchphrases were generated for each document together with the corresponding scores for each catchphrase.

3.6 Final Run 2: Using MAUI

The supervised machine learning approach (MAUI) was used where a classifier was trained by using all the training documents provided with known training catchphrases in the training set. No candidates were discarded prior to training the model. We then used the trained model to generate catchphrases for the test documents.
UBIRLeD_2: 150 catchphrases were generated for each test document. The highest ranked catchphrases appeared first for each test document.

4 EVALUATION

Several measures were used to evaluate the results of the two approaches. In this experiments we looked at Recall, Precision and Mean Average Precision among others.

4.1 Recall Measure

According to Manning et al [3] *Recall* is defined as the fraction of relevant documents that are retrieved. In this task, we were interested in the fraction of relevant catchphrases retrieved in each document. The formula for Recall is given in Figure 1, where tp represents true positive; these are relevant retrieved catchphrases and fn represents false negative; these are relevant but not retrieved catchphrases.

$$Recall = \frac{tp}{tp + fn}$$

Figure 1: Recall equation as described by Manning et al [3]

Recall@K would be the proportion of relevant catchphrases that have been retrieved in the top-K.

Table 1: Results for RAKE parameter tuning

RAKE Experiments For Parameter Tuning				
Test Number	No of Char/word	No of words/phrase	keyword frequency	Recall
1	5	3	4	5.65
2	**3**	**3**	**1**	**25.78**
3	3	3	2	19.64
4	3	3	3	13.04
5	3	3	4	8.62

Table 2: Results for MAUI parameter tuning

MAUI Experiments For Parameter Tuning		
Test Number	Frequency of phrases to keep	Recall
1	**1**	**68.27**
2	2	48.62
3	3	31.24
4	10	6.03

4.2 Precision Measure

Precision is described as the fraction retrieved documents which are relevant according to Manning et al [3]. In this time precision will be the fraction of retrieved catchphrases that are relevant. The formula for Precision is given in Figure 2, where tp represents true positive and fp represents false positives; a situation in which non-relevant catchphrases have been retrieved as relevant.

$$Precision = \frac{tp}{tp + fp}$$

Figure 2: Precision equation as described by Manning et al [3]

Precision@K, would be the proportion of top-K catchphrases that are relevant. Mean Precision@K, will then cover the Mean of the Precision@K of each test document in the whole collection.

We used Manning et al [3]'s ideas when finding the Mean R precision. Computing Mean R precision required knowledge of all catchphrases that were relevant on each test document where R represented the total number of expected relevant catchphrases for a particular test document. R was then used as the cutoff for calculating precision. Precision would be equal to recall at the R-th position. Suppose that R relevant catchphrases were expected for test document Td1, and only r relevant catchphrases were retrieved at position R. We would only calculate precision of the top R catchphrases retrieved using the formula given in Figure 3. The Mean R precision would be the mean of R precision of all the test documents (queries)

$$RPrecision = \frac{r}{R}$$

Figure 3: R Precision equation as described by Manning et al [3]

4.3 Mean Average Precision Measure

Mean Average Precision (*MAP*) value is defined as "the arithmetic mean of average precision values for individual information needs" Manning et al [3]. The formula for Mean Average Precision (*MAP*) is given in Figure 4, where $MAP(Q)$ is mean of average precision across the whole collection list of queries being the test documents in this task. $Precision(R_{jk})$ is the precision score of ranked retrieved catchphrases from the top results until position k for test document j. For each of the test documents, a set of ranked catchphrases was produced, which was then used to compute precision and average precision (AP). Average precision is the mean of the precision scores after each relevant catchphrase is retrieved.

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=i}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk})$$

Figure 4: MAP equation as given by Manning et al [3]

5 RESULTS

Consider the results displayed in Table 3 UBIRLed_1 and UBIRLed_2 rows contain performance measures obtained after using the generated catchphrases from RAKE and MAUI respectively as mentioned in . Using the performance measures stated in Section 4, we observed that MAUI; the supervised approach, performed far much better than RAKE; the unsupervised approach. Comparing the results based on Mean Precision@10, we discovered that the proportion of top 10 catchphrases which were relevant was more effective using MAUI, MAUI result was 0.254 while RAKE result was 0.013. We also looked at the Mean Recall@100, MAUI still outperformed RAKE by retrieving more relevant catchphrases in the top 100. When finding MAP, the assumption was that we were interested in finding more relevant catchphrases for each test documents and hence we computed the Mean of average precision values of each test documents. The value of MAP obtained for MAUI was higher than the value computed using RAKE results. The Mean R precision value for MAUI had far much better proportion of retrieved catchphrases which were relevant considering the cutoff point which was equals the number of relevant catchphrases expected for each and every document provided in the testing dataset. Overall recall, RAKE was better although that was the only measure good compared to MAUI's performance.

Table 3: Final Results for RAKE and MAUI using Test documents

RAKE and MAUI Results					
Evaluation Metrics	Mean R precision	Mean Precision@10	Mean Recall@100	MAP	Overall Recall
UBIRLed_1	0.02316392684	0.01366666667	0.1723154757	0.04634794783	0.4992190452
UBIRLed_2	0.1901020309	0.2543333333	0.3050612978	0.3703664676	0.3259790763

6 CONCLUSION AND FUTURE WORK

In this paper we had proposed and compared two Machine Learning approaches namely: RAKE and MAUI for the unsupervised and supervised approaches respectively. In the proposed approach, fine tuning parameters before generating candidate catchphrases resulted in obtaining the optimal parameters for each method used. Based on the optimal parameters used for generating the final catchphrases, overall MAUI had high performance compared to RAKE. The differences in the performance was observed in most areas. RAKE achieved the highest recall but the precision was very low compared to MAUI. We strongly believe that Legal domain is an area which still requires a lot of work on Information Extraction. For the future work, we plan to experiment with different techniques used on the supervised approach in Machine learning and evaluate the performance after applying the different techniques.

REFERENCES

[1] Slobodan Beliga, Ana Metrovi, and Sanda Martini-Ipi. 2015. An Overview of Graph-Based Keyword Extraction Methods and Approaches. *Journal of information and organizational sciences* 39, 1 (June 2015), 1–20. http://hrcak.srce.hr/140857

[2] Arpan Mandal, Kripabandhu Ghosh, Arnab Bhattacharya, Arindam Pal, and Saptarshi Ghosh. 2017. Overview of the FIRE 2017 track: Information Retrieval from Legal Documents (IRLeD). In *Working notes of FIRE 2017 - Forum for Information Retrieval Evaluation (CEUR Workshop Proceedings)*. CEUR-WS.org.

[3] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval.* Cambridge University Press, Cambridge, UK. http://nlp.stanford.edu/IR-book/information-retrieval-book.html

[4] Olena Medelyan. 2009. Human-competitive automatic topic indexing. (2009). http://cds.cern.ch/record/1198029 Presented on July 2009.

[5] Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley. 2010. Automatic Keyword Extraction from Individual Documents. (03 2010), 1 – 20.

[6] Lima Subramanian and R.S Karthik. 2017. KEYWORD EXTRACTION: A COMPARATIVE STUDY USING GRAPH BASED MODEL AND RAKE. *Int. J. of Adv. Res. 5 (3)* (2017), 1133–1137. https://doi.org/10.21474/IJAR01/3616

[7] Ian H. Witten, Gordon W. Paynter, Eibe Frank, Carl Gutwin, and Craig G. Nevill-Manning. 1999. KEA: Practical Automatic Keyphrase Extraction. (5 Feb. 1999). arXiv:cs/9902007 http://arxiv.org/abs/cs/9902007