ISSN 1870-4069

Towards Topic Detection in Plain Documents Using Key Terminology Extraction Supported by Institute of Technology Tallaght

Juan Huetle-Figueroa[1], Fernando Perez-Tellez[1], David Pinto[2]

[1] Institute of Technology Tallaght,
Ireland

[2] Benemérita Universidad Autónoma de Puebla,
Mexico

{juan.huetle, fernandopt, juan.huetle, fernandopt}@gmail.com,
dpinto@cs.buap.mx

Abstract. This paper presents an approach for helping in the topic detection tasks. The idea is to use collocation measures to extract key terminology from plain text. We use three measures for ranking N-grams (sequence of terms), Point mutual information, Likelihood-ratio and Chi-square. With this measures we built three different groups: bigrams, trigrams and quadrigrams. Each of the measures were implemented with the purpose of comparing and helping in the detection of good key terminology in plain text. In order to obtain the best N-grams, we have implemented two filters: the first one is to get common N-grams with the highest values in the three measures (intersection). For this, we use the most significant percentages of N-grams to create subsets and then select the key terminology with high value in the three measures. The second filter is to detect the occurrence of important collocations based on part-of-speech patterns. The corpora used in this research work was obtained from the website jobs, i.e. related to job descriptions. In the results we show the key terminology extracted by this approach to demonstrate its effectiveness.

Keywords: collocations, n-grams, POS, key term extraction.

1 Introduction

Collocations refer to words relationships, there are useful in the natural language processing area (NLP). They are expressions formed with two or more consecutive terms that correspond to a way of saying concrete ideas or concepts. They usually include noun phases such as *deep learning* or phrasal verbs such as *to look for*. The use of collocations in the NLP area makes the text to sound natural and makes more sense to people. The importance of the experiments presented in this research work is to obtain a list of relevant topics discussed in plain text

pp. 27–39; rec. 2018-03-24; acc. 2018-05-16

through the detection of key terminology. In order to achieve this goal we have defined a range of measures to compare them with each other and detect the best key terminology in a given text.

The measures used are Pointwise mutual information (PMI), Likelihood-radio and Chi-square. They were chosen for simplicity, low compute capacity required and they showed acceptable results in the experiments. Also in this research work, we used N-grams which are sequence of n terms, in particular we used bigrams (two terms), trigrams (three terms) and quadrigrams (four terms). We created two different experiments one with stop words and another without stop words. Stop words are words with very little or no lexical meaning such as *and, a, to,* and *in*). Therefore, we defined two different ways of analysing the N-grams. We considered that the phrase “state of the art” does not have the same meaning as “state art”. It can be noticed that these phrases have two different meanings. The first phrase refers to ”the latest and most sophisticated or advanced stage of a technology, art, or science.”[12] and the second has no understandable meaning. Ranking N-grams with measures mentioned before we could understand and identify different key terminology. To do this, we extracted the best terms by selecting collocations with high value in the measures mentioned previously.

In this research work, we used intersection between the sets of N-grams ranked by collocation measures and filtered by the highest values. The intersection task is when a set of N-grams filtered by the highest value of a collocation measure appears in another set of N-grams filtered by a different collocation measure. In Table 6 the phrase ”dublin city centre” appears with high value in PMI, Likelihood-ratio and Chi-square.

Table 1. Universal POS.

Universal POS	
ADJ: adjective	PART: particle
ADP: adposition	PRON: pronoun
ADV: adverb	PROPN: proper noun
AUX: auxiliary	PUNCT: punctuation
CCONJ: coordinating conjunction	SCONJ: subordinating conjunction
DET: determiner	SYM: symbol
INTJ: interjection	VERB: verb
NOUN: noun	X: other
NUM: numeral	

We carried out experiments where parts of speech (POS) are used. In the Table 1 and 2, we present the different POS used in this research work: *noun, pronoun, adjective, determiner, verb, adverb, preposition, conjunction, and interjection.* They were used to identify different lexical patterns in the N-grams. We briefly explain the experiments carried out in this research work.

- **Experiment 1:** N-grams are created with stop words such as (*a, at, of, the, etc.*) to created the specific correct key terminology.
  – Experiments with universal and normal POS.
  – We refer as 'universal POS' to the tokenization in the text with 17 possible tags shown in Table 1.
  – We refer to normal POS to the tokenization in the text with 45 possible tags shown in Table 2.
- **Experiment 2:** N-grams are created without stop words to be able to use them in queries or phrases, it is not the same meaning ”deep learning” and ”deep” or ”learning” as you can see they have two different meanings, the first pair of terms together refer to the specific area of computing and the other two terms individually are only single words with different meanings.
  – Experiments with normal and universal POS

Table 1 and Table 2 show the different POS tags and their respective meanings. Table 1 shows 17 different tags. Having few tags made us limited ourselves in our experiments. In this research work, we needed use specific tags such as the adjective(JJ) and superlative(JJS) to apply a filter. Besides that we need to use the comma, explained in Section 3.1 Prepossessing.

Table 2 shows the tags used in this research work. There are 45 possible tags, giving us a closer approximation about the text and it provides with the necessary tools to apply the filter used in this research work.

Table 2. Normal POS.

Normal POS		
Punctuation Marks: “““”, “:”	NNP: noun, proper, singular	VBD: verb, past tense
“;”, “””, “(”, “)”, “,”, “–”, “.”	NNS: noun, common, plural	VBG: verb, present
CC: conjunction, coordinating	PDT: pre-determiner	participle or gerund
CD: numeral, cardinal	POS: genitive marker	VBN: verb, past participle
DT: determiner	PRP: pronoun, personal	VBP: verb, present tense,
EX: existential there	PRP$: pronoun, possessive	not 3rd person singular
FW: foreign word	RB: adverb	VBZ: verb, present tense,
IN: preposition or conjunction	RBR: adverb, comparative	3rd person singular
JJ: adjective or numeral	RBS: adverb, superlative	WDT: WH-determiner
JJR: adjective, comparative	RP: particle	WP: WH-pronoun
JJS: adjective, superlative	SYM: symbol	WRB: Wh-adverb
MD: modal auxiliary	TO: ”to” as preposition	
NN: noun, common	VB: verb, base form	

The rest of this paper is organized as follows. The next section provides a review of related work to obtain key terminology, methods and application. In Section 3, we describe the measures used and intersection of the terms with their measures. Section 4 describes the corpus (dataset) used for the experiments. Section 5 contains a description about the preprocessing of the data, and the two experiments carried out in this research work. Finally, in the last Section, we conclude the paper and outlines future work directions.

2 Related Work

Our research goal is to obtain key terminology from plain documents, we have studied previous research works focused on keyword extraction. Researchers in [13] have reported the use of statistical methods and approaches such as simple statistics, linguistics, machine learning approaches. They extracted small set of units, composed of one or more terms, from a single document. They discussed about the extraction of small units sets, composed of one or more terms, from a single document. It is an important problem in Text Mining (TM), Information Retrieval (IR) and Natural Language Processing (NLP). Authors focused on the graph-based methods. They have compared methods with existing supervised and unsupervised methods. On the other hand, [8] used statistical methods with TFIDF (term frequency, inverse document frequency) they described the use of TFIDF in different parts of the plain document. For example, if a word appears sporadically in more than half of the document it is also considered as a keyword without taking into account the stop words. As well as multiple times in a single paragraph but not in the overall document TFIDF will not consider the word as keyword considering its low frequency.

In [7] authors used unsupervised approaches to automate the keyword extraction process from meeting transcript documents and they incorporated the use part-of-speech (POS) information in similar manner that we did. Then, they identified key-words using F-measure and a weighted score relative, giving them good results with TFIDF. The data that they used was *meeting recordings* converted into text.

The authors of [11] automatically generated a headline for a single document. They mixed sentence extraction and machine learning, their corpus were scientific articles. Another interesting approach is [1] they combine resources for lexical analysis such as electronic dictionary, tree tagger, WordNet, N-grams, POS pattern, resulting in a survey, they used different dataset the most relevant for us is the web pages, encyclopedia article, newspaper articles, journal articles and technical report. In [14] used *salience* rank in 500 news articles, the result was to improve the quality of extracted keyphrases and balance topic in corpus.

There is also some research in the field of real-time automatic speech recognition. In [4] authors applied keywords to formulate implicit queries to a just-in-time-retrieval system for use in meeting rooms.

3 Measures

We used three types of collocation measures to define the best filter in the N-grams. These measures were chosen for the easy implementation, good results and the low computing power needed with large volume of information, the following measures have been reported in [10].

- **PMI** Pointwise mutual information is a measure of association:

$$pmi(x;y) \equiv log\frac{p(x,y)}{p(x)p(y)}, \tag{1}$$

$pmi(x;y)$ means the association between two terms (bigram), the first word is represented with x and the second word with y. It's a popular measure for the simply implementation and the good results.

- **Likelihood-ratio** We used *"maximum Likelihood-estimation"* to decide if there is a important contrast between the expected and the observed frequencies in bigrams, trigrams and quadrigrams. This measure expected two hypothesis $L(H_1)$ and $L(H_2)$ shown in the formula (2).The following formula describe the occurrence frequency of a bigram w^1w^2.

  Hypothesis 1. The occurrence of w^2 is independent of the previous occurrence of w^1:
  $$P(w^2|w^1) = p = P(w^2|\neg w^1).$$
  Hypothesis 2. It is a formalization of dependence which is good evidence for an interesting collocation:
  $$P(w^2|w^1) = p_1 \neq p_2 = P(w^2|\neg w^1).$$
  For p, p_1 and p_2 and write c_1, c_2, and c_{12} for the number of occurrences of $w1, w2$ and w^1w^2 in the corpus[10].
  $$\begin{aligned} log\lambda &= log\frac{L(H_1)}{L(H_2)}, && (2)\\ &= log\frac{b(c_{12}, c1, p)b(c_2 - c_{12}, N - c_1, p)}{b(c_{12}, c1, p_1)b(c_2 - c_{12}, N - c_1, p_2)}, && (3)\\ &= logL(c_{12}, c_1, p) + logL(c_2 - c_{12}, N - c_1, p), && (4)\\ &\quad -logL(c_{12}, c_1, p_1) - logL(c_2 - c_{12}, N - c_1, p_2). && (5)\end{aligned}$$

- **Chi-square** We used Chi-square with the same purpose that Likelihood ratio search important contrasts between the frequencies in bigrams, trigrams and quadrigrams, the formula (6) shown how work:
  $$X^2 = \sum_{i,j} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}, \qquad (6)$$
  where i ranges over rows of the table, j ranges over, O_{ij} is the observed value for cell (i,j) and E_{ij} is the expected value.

3.1 Intersection

We implemented Likelihood-ratio positive, because we are only interested in positive results. A positive result means an estimate of the occurrence of an N-gram in the corpus and a negative result is the estimate that an N-gram does not occur in the corpus. We create a filter derived from the aforementioned measurements, we take the results of each one and we intersect them giving a subset. That is to say each one has its own range, so only took the best results of each one. We represent the set PMI as set A, Likelihood-radio as set B and Chi-square as set C. Thus we get the following intersections.

- In Table 3, we can observe $A \cap B$ (see Fig. **??**). This intersection between two sets of values PMI and Likelihood-ratio, where both have hight values and we see the 10 first trigrams with the highest value. For do the intersection only 50% was taken that is to say one subset from A and another B. You can see the difference in the N-gram "competitive salary earn" has in PMI 331.944 higher than Likelihood-ratio with 21.049 Table 3.

Table 3. Sample trigrams filtered by the intersection $A \cap B$.

Trigram	Freq.	PMI	Likelihood-ratio
dublin city centre	53	2019.562	17.231
telecoms tech support	13	501.210	18.167
successful candidate joining	3	496.637	18.667
third level qualification	7	349.176	18.474
benefits competitive salary	3	341.853	18.110
competitive salary earn	2	331.944	21.049
fast paced environment	6	328.250	17.544
equal opportunities employer	6	316.0285	21.500
competitive salary gym	2	314.754	18.242
proven track record	6	306.108	21.363

- In Table 4, we can observe $A \cap C$ (see Fig. **??**). This intersection between two measures PMI and Chi-square, where both have hight values and we see the 10 first trigrams with the highest value. In special the term "equal opportunities employer" start to obtain key terminology. If you compare Table 3 with Table 4 you will start to see deleted terms.

Table 4. Sample trigrams filtered by the intersection $A \cap C$.

Trigram	Freq.	PMI	Chi-square
dublin city centre	53	8154393.488	17.231
telecoms tech support	13	3826386.926	18.167
successful candidate joining	3	1258923.953	18.667
benefits competitive salary	3	856803.543	18.110
competitive salary earn	2	4349623.070	21.049
fast paced environment	6	1149592.709	17.544
equal opportunities employer	6	17803759.837	21.500
competitive salary gym	2	628854.620	18.242

- In Table 5, we can observe $B \cap C$ (see Fig. **??**). This intersection between two measures Likelihood-ratio and Chi-square, where both have highest values and we see the 10 first trigrams with the highest value. We can see that measure Chi-square delete terms because they do not exist in their subset.
- In Table 6, we can observe $A \cap B \cap C$ (see Fig. **??**). This intersection between three measures PMI, Likelihood-ratio and Chi-square. It is one of the main

Table 5. Sample trigrams filtered by the intersection $B \cap C$.

Trigram	Freq.	Likelihood-ratio	Chi-square
related locations dublin	61	2371.108	1662990.691
centre job description	24	2202.153	994671.932
south job description	20	2198.519	1503450.729
cork job description	14	2067.289	539817.166
limerick job description	9	2021.597	559561.417
dublin city centre	53	2019.562	8154393.488
waterford job description	6	1987.107	501852.430
laois job description	4	1967.965	502968.807
locations dublin city	31	1787.154	1764364.726

objectives of this research work, because we can observe how begin to filter the information. You can see the respective measure of each one. When comparing the Tables 3, 4 and 5, we see that the measure with the most delete terms was Chi-Square.

Table 6. Sample trigrams filtered by the intersection $A \cap B \cap C$.

Trigram	Freq.	PMI	Likelihood-ratio	Chi-square
dublin city centre	53	2019.562	8154393.488	17.231
telecoms tech support	13	501.210	3826386.926	18.167
successful candidate joining	3	496.637	1258923.953	18.667
third level qualification	7	349.176	2550273.661	18.474
benefits competitive salary	3	341.853	856803.543	18.110
competitive salary earn	2	331.944	4349623.070	21.049
fast paced environment	6	328.250	1149592.709	17.544
equal opportunities employer	6	316.028	17803759.837	21.500
competitive salary gym	2	314.754	628854.620	18.242
proven track record	6	306.108	16186542.685	21.363

4 Data

In this research work, we were working with jobs descriptions, all the data was taken from jobs.ie[3] a website in Ireland. The website has 46 different sectors and a number of jobs description on each sector. They are shown in the Table 7. Each job description file contains information as skills needed, payments and area of work. All the documents were in HTML and JSON format, we had to clean the documents from HTML tags, and download the updated information for each week. For this research work, we used in specific the IT (information technology) list count with 153 jobs descriptions, the average of clean files is 3

[3] https://www.jobs.ie/

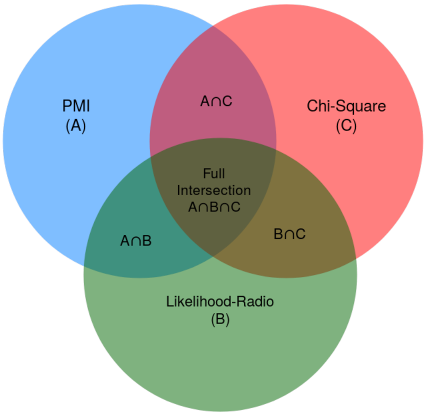

Fig. 1. Set intersection.

Kilobytes per file. To collect these data we used a web crawler (HTTrack)[4] to automatically download all the jobs descriptions every week.

The reasons to chose these data are:

- The potential to use the key terminology to match job seeker and companies.
- The functionality of using different work sectors in the corpus.
- Use the N-grams in open questions for the companies.
- The volume of real information retrieved.
- The diversity of information content.
- To use the information obtained in the future in conjunction with the CV to make a semantic matches.

5 System Overview

We carrying out two different experiments: the first is using the stop words and part of speech and the second one was without stop words.

5.1 Preprocessing

The following list shows the preprocessing for this research work.

- We explained in section 4 that whole data was downloaded in HTML and JSON files.
- We clean all unnecessary lines such as HTML and JavaScript tags in the corpus.
- The information was storaged in different files such as *job1, job2, ... jobn.*
- We created a string with all this information.

[4] https://www.httrack.com

Table 7. Categories of job descriptions.

Sector	num.	Sector	num.	Sector	num.
Academic	21	Pubs, Bars and Clubs	199	HR/Recruitment	102
Architecture/Design	20	Retail	293	Legal	52
Big Data/Business A.	17	Sales - Up to 35k	297	Manufact./Engineering	140
Chef Jobs	374	Security	34	Miscellaneous	54
Construction/Eng.	103	Telec./Tech Support	45	Multi-lingual	143
Education/Training	77	Travel/Tourism	92	Pharma./Sci./Agricul.	116
Financial Services	101	Warehouse/Logis./Ship.	153	Proper./Facilities Manag.	59
Franchise/Business	5	Accountancy/Finance	304	Restaurant/Catering	669
Hair and Beauty	100	Banking/Insurance	110	Sales - 35k+	270
Hotels	1021	Call-Centre/Cust. Serv.	340	Secre./Admin/PA	257
IT	153	Childcare	54	Senior Appointments	23
Manager/Supervisor	267	Drivers	71	Trades/Operative/Man.	144
Marketing	99	Renewable Energy	9	Charity Work	31
Motors	120	Fitness and Leisure	55	Work Exp./Internship	6
Online/Digital M.	47	Graduate	63		
Merchandising	43	Health/Med./Nursing	156		

- We removed all symbols such as *@, ”, ’, *, ?, , etc.* because the job description is written by the companies and they usually use symbols.
- We convert all the letters in lowercase, because it is the same say *”computer science”* that *”Computer science”*, only change the first letter and we had two different bigrams (in this case).
- We used NLTK[5] to tokenize the whole corpus with POS[6] functions, because NLTK works by context that is to say use the words before and after of each word, one example is ”Support” could be a noun or verb.
- We discard possibles combinations with *”.”*, *”,”* and *”;”*, for example we had a lot of incomplete ideas such as *”customers, and providing”* and *”innovation happens. And”*. For this the program we developed uses a classification pattern when put a conditional.

For the second experiment we used a stop words list, to not discard combinations. In Table 8. we can see how was building the N-grams used in this research work.

5.2 Experiment 1

Experiment 1 presents the set intersection between the measures use to rank terms but without POS filter and order by Likelihood-ratio. We can see the different results in the Table 9 and 10.

[5] Natural Language Toolkit https://www.nltk.org/

[6] Part of speech

Table 8. How the N-grams were created.

N-gram	Freq.	N-gram	Freq.	N-gram	Freq.
hardware software	12	skills experience	16	software development	21
centre dublin	12	excellent communication	17	dublin city centre	21
dublin south job description	12	related job description	18	dublin city	24
city centre dublin	12	locations job description	19	city centre job description	24
part of team	13	related job	19	project management	24
team player	13	locations job	19	years experience	25
tech support	13	south job description	20	successful candidate	25
customer satisfaction	13	skills ability	20	related city	26
strong knowledge	13	south job	20	related city centre	26
work environment	14	locations city centre job	21	centre job	27

In Table 9, we can observe that the measures Chi-square and PMI are not congruent in a descending or ascending form. This is due to the fact that many terms were discarded by the intersection. The Likelihood-radio results are ordered in a descending form but between each value there are a big difference, this is also due to the fact that N-grams were discarded.

To explain better why N-grams are discarded when the intersection of the three measurements is done. It is necessary to know that an intersection is a subset of other sets, in this case of three sets (measures). We call full intersection to this subset (see Fig. 1).

Table 9. Sample trigrams filtered by the intersection process.

Trigram	Freq.	Likelihood-ratio	Chi-square	PMI
related locations dublin	61	2371.108	1662990.691	14.732
centre job description	24	2202.153	994671.932	15.312
south job description	20	2198.519	1503450.729	16.178
cork job description	14	2067.289	539817.166	15.172
limerick job description	9	2021.597	559561.4171	15.856
dublin city centre	53	2019.562	8154393.488	17.231
waterford job description	6	1987.107	501852.430	16.271
laois job description	4	1967.965	502968.807	16.856
job description summary	3	1943.123	295844.586	16.441

5.3 Experiment 2

Experiment 2 is defined by the intersection of sets generated by the three collocation measures defined and a POS filter. We also used tokenization with POS tags. The POS filter consists in verify if the first word is tagged by a *JJ* or *NN* followed by any other tag or couple of tags and ending with a tag *NNS* or *NN*. For instance, in Table 10 we can see N-grams filtered by discarding mainly verbs.

In Table 11, we can observe the N-grams that did not followed the POS pattern defined. We can see a pattern at the beginning of the N-grams that start with the following tags: *IN*, *VB*, *VBG* or *RB*. Taking into account this pattern, the filter was created discarding all the N-grams that had that pattern. We called this discarding as POS filter.

It is important to note that we only defined the POS pattern at beginning and at the end of the N-grams that means that in the middle of the N-grams could be any other N-grams with any POS tag.

Table 10. Trigram with set intersection and filter with POS.

Trigram	Freq.	Likelihood-ratio	Chi-square	PMI
related/JJ locations/NNS dublin/NN	61	2371.108	1662990.691	14.732
centre/NN job/NN description/NN	24	2202.153	994671.932	15.312
south/NN job/NN description/NN	20	2198.519	1503450.729	16.178
cork/NN job/NN description/NN	14	2067.289	539817.166	15.172
limerick/NN job/NN description/NN	9	2021.597	559561.417	15.856
dublin/NN city/NN centre/NN	53	2019.562	8154393.488	17.231
waterford/NN job/NN description/NN	6	1987.107	501852.430	16.271
laois/NN job/NN description/NN	4	1967.965	502968.807	16.856
job/NN description/NN summary/NN	3	1943.123	295844.586	16.441
wicklow/NN job/NN description/NN	3	1939.786	242438.628	16.119

Table 11. Trigram with set intersection and tokenized but without filter POS.

Trigram	Freq.	Likelihood-ratio	Chi-square	PMI
ensure/VB customer/NN satisfaction/NN	2	223.515	41012.619	14.247
across/IN multiple/NN projects/NNS	2	208.225	69104.106	15.030
establish/VB best/JJS practice/NN	2	177.990	3266621.424	20.638
across/IN multiple/NN time/NN	2	176.725	92012.22154	15.458
rewarding/VBG work/NN environment/NN	3	170.576	191825.611	15.962
privately/RB owned/VBN media/NNS	3	156.291	67784782.307	24.429

6 Conclusion

We start out by choosing three measures: PMI, Chi-square and Likelihood-ratio to rank N-grams (bigrams, trigrams and quadrigrams) and obtain key terminology from different plain documents. We have shown that intersecting the highly ranked N-grams (with collocation measures) can help in filtering out irrelevant terms and identify useful key terminology. In this research work, we also have used specific POS tags to rule out the unnecessary N-grams. The POS pattern used to detect important key terminology consist of having the first word tagged

as *JJ* or *NN* followed by any word or couple of word with any POS tag(s) and ending with a word tagged as *NNS* or *NN*. This pattern contributed to obtain good results. This idea can be applied in other corpus to obtain key terminology by defining a POS pattern to filter relevant N-grams.
In these experiments, we show that using the POS patterns can help in better detection of key terminology. We also used the intersection of highly ranked N-grams by collocation measures and we got better key terminology when we applied both. Future work includes corpus evaluation with precision and recall to obtain the relevant subsets. We are also planning to use a thesaurus to enrich the key terminology obtained in this work then to use machine learning algorithms fed by the enriched key terminology.

References

1. Bharti, S.K., Babu, K.S.: Automatic keyword extraction for text summarization: A survey (2017)
2. Bird, S.: NLTK: The natural language toolkit. In: Proceedings of the COLING/ACL on interactive presentation sessions, pp. 69–72, Association for Computational Linguistics (2006)
3. Brezina, V., McEnery, T., Wattam, S.: Collocations in context. International Journal of Corpus Linguistics, 20(2), pp. 139–73 (2015)
4. Habibi, M., Popescu-Belis, A.: Keyword extraction and clustering for document recommendation in conversations. IEEE Trans Audio Speech Lang Process 23(4), pp. 746–759 (2015)
5. Jurafsky, D., Martin, J.H.: Speech and language processing. Pearson (2014)
6. Kantor, P.: Foundations of statistical natural language processing. Information Retrieval, 4(1), pp. 80–81 (2001)
7. Liu, F., Pennell, D., Liu, F., Liu, Y.: Unsupervised approaches for automatic keyword extraction using meeting transcripts. In: Proceedings of human language technologies: The Annual conference of the North American chapter of the association for computational linguistics, pp. 620–628, Association for Computational Linguistics (2009)
8. Luthra, S., Arora, D., Mittal, K., Chhabra, A.: A Statistical Approach of Keyword Extraction for Efficient Retrieval. International Journal of Computer Applications, 168(7) (2017)
9. Maldonado-Guerra, A., Emms, M.: Measuring the compositionality of collocations via word co-occurrence vectors: Shared task system description. In: Proceedings of the Workshop on Distributional Semantics and Compositionality (ACL), pp. 48–53, Portland (2011)
10. Manning, C.D., Schütze, H.: Foundations of statistical natural language processing. MIT press (1999)
11. Mondal, A.K., Maji, D.K.: Improved algorithms for keyword extraction and headline generation from unstructured text. First Journal publication from SIMPLE groups, CLEAR Journal (2013)
12. Oxford dictionaries: http://www.dictionary.com/browse/state–of–the–art?s=t (2018)
13. Beliga, S.: Keyword extraction: A review of methods and approaches (2014)

14. Teneva, N., Cheng W., Salience, R.: Efficient Keyphrase Extraction with Topic Modeling. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (2), pp. 530–535 (2017)