

Semi-Supervised Keyphrase Extraction on Scientific Article using Fact-based Sentiment

Felix Christian Jonathan*¹, Oscar Karnalim²

Maranatha Christian University, Indonesia

*Corresponding author, e-mail: oscar.karnalim@gmail.com

Abstract

Most scientific publishers encourage authors to provide keyphrases on their published article. Hence, the need to automatize keyphrase extraction is increased. However, it is not a trivial task considering keyphrase characteristics may overlap with the non-keyphrase's. To date, the accuracy of automatic keyphrase extraction approaches is still considerably low. In response to such gap, this paper proposes two contributions. First, a feature called fact-based sentiment is proposed. It is expected to strengthen keyphrase characteristics since, according to manual observation, most keyphrases are mentioned in neutral-to-positive sentiment. Second, a combination of supervised and unsupervised approach is proposed to take the benefits of both approaches. It will enable automatic hidden pattern detection while keeping candidate importance comparable to each other. According to evaluation, fact-based sentiment is quite effective for representing keyphraseness and semi-supervised approach is considerably effective to extract keyphrases from scientific articles.

Keywords: Fact-based sentiment; Semi-supervised approach; Keyphrase extraction; Scientific article; Deep belief network

Copyright © 2018 Universitas Ahmad Dahlan. All rights reserved.

1. Introduction

Keyphrases (or keywords) are natural language terms used to represent the context of a document [1]. It is frequently used to check whether given document matches user need without reading the whole content. Keyphrases are frequently found on scientific articles [2]; scientific publishers encourage authors to provide those phrases on published article so that prospective readers will not waste their time for reading irrelevant articles comprehensively. Considering this need, automatic keyphrase extraction approaches have been developed to mitigate human effort [2].

Automate keyphrase extraction is not a trivial task; keyphrase characteristics may overlap with the non-keyphrase's [3]. For instance, even though most keyphrases are found in abstract, not all abstract terms are keyphrases. As a result, additional unique characteristics are introduced to distinguish keyphrases from non-keyphrases.

It is true that the use of numerous characteristics enhances the accuracy of automatic keyphrase extraction for scientific articles. However, to date, its accuracy is still considerably low [4]. Hence, this paper introduces fact-based sentiment as a new keyphrase characteristic. Different with standard sentiment, it is purely resulted from facts (with an assumption that scientific articles contain facts). We would argue that such characteristic may enhance the accuracy of existing keyphrase extraction since, according to our manual observation; fact-based sentiment of keyphrases is patterned implicitly on scientific articles: most keyphrases are mentioned while discussing novelty and benefits of their work and these aspects are frequently written as fact-based sentences in neutral-to-positive sentiment.

To enhance the accuracy further, we also propose a combination of supervised and unsupervised approach for extracting keyphrases. It is inspired from [5] where each candidate is sorted in unsupervised manner (i.e., TF-IDF ranking [1] in our case) and fed into a classifier to determine its keyphraseness in supervised manner (i.e., Deep Belief Networks (DBN) [6] in our case).

2. Related Works

In general, keyphrase extraction can be roughly classified into two categories: unsupervised or supervised approach [2]. Unsupervised approach relies on ranking and similarity mechanism. It assigns each keyphrase candidate with a particular score and select the highest-scored candidates as its keyphrase. Some examples of such approach are works proposed in [7–11] where they use graph-based ranking, topic-based clustering, simultaneous learning, language modeling, and conditional random fields respectively. In contrast, supervised approach relies on learning algorithm and training dataset. Different with unsupervised approach, the pattern is not required to be defined manually. It is automatically extracted from training dataset. Two learning algorithms which have been used in this approach are naive bayes [12,13] and decision tree [14].

Most works about keyphrase extraction are focused on scientific articles considering keyphrases (or keywords) are required to represent each scientific article [15–19]. Some examples of them are: 1) a work proposed in [15] that combines maximal frequent sequences and PageRank; 2) a work proposed in [20] that extracts keyphrases based on sentence clustering & Latent Dirichlet Allocation; 3) a work proposed in [16] that utilizes skill set concept; and 4) a work proposed in [17] that incorporates Likely ratio.

It is important to note that keyphrase extraction is not the only emerging topic regarding scientific articles. Other topics such as title generation [21], scientific data retrieval [22], scientific article management [23], and publication repository [24] are also emerged. However, in this work, we will only focus on keyphrase extraction.

The accuracy of keyphrase extraction on scientific articles can be enhanced through three mechanisms: local extraction, structure utilization, and implicit behavior utilization. First, local extraction means that only candidates from particular sections will be considered; those sections are assumed to have all keyphrases. Some sections which have been used for local extraction are abstract [25,26], references [18], and the first 2000 characters [27]. Second, structure utilization means that article structure is converted to feature(s) for determining keyphraseness. It is frequently used with an assumption that the structure of scientific article can be generalized. This mechanism has been used in several works [28–30] where some features are derived from article structure. Third, implicit behavior utilization means that the behavior of scientific articles is mapped to feature(s) for determining keyphraseness. For instance, Treeratpituk et al [31] and Berend & Farkas [32] consider acronym as one of their learning features with an assumption that it is frequently used on scientific articles to enhance article readability. Another example is a work proposed in [33] that does not favor terms in bracket as a feature. They assume such terms are seldom used as keyphrases.

Nevertheless, despite several enhancement mechanisms exist, the accuracy of keyphrase extraction on scientific articles is still low [34]. We would argue that such phenomenon is caused by two rationales. First, some keyphrase characteristics are not exclusively owned by keyphrases; they are also owned by non-keyphrases. Second, both supervised and unsupervised approach have their own drawback [5]. Unsupervised approach disables automatic hidden keyphrase pattern detection while supervised approach disables comparable candidate importance.

3. Methodology

This paper aims to enhance the accuracy of keyphrase extraction on scientific articles by proposing two contributions. First, a feature called fact-based sentiment is proposed. It works in similar manner as standard sentiment except that it is purely based on fact. Such feature is expected to strengthen keyphrase characteristics since, according to manual observation, most keyphrases are mentioned while discussing novelty & benefits of their work and these aspects are frequently written as fact-based sentences in neutral-to-positive sentiment. It is true that some keyphrases are also written while discussing related works and drawbacks. However, its occurrence is typically low due to research scope and limited paper page. Second, a combination of supervised and unsupervised approach is proposed to take the benefits of both approaches [5]. It will enable automatic hidden pattern detection while keeping candidate importance comparable to each other.

In general, our proposed keyphrase extraction consists of three phases which are: 1) keyphrase candidate identification; 2) keyphrase ranking; and 3) keyphrase classification.

Further, our work also incorporates a module to train learning model for keyphrase classification.

3.1. Keyphrase Candidate Identification

This phase identifies keyphrase candidates from a scientific article based on several limitation heuristics. These heuristics are:

- a. Keyphrase candidate should be a noun phrase with phrase length lower or equal with 4 words. This heuristic is applied since most keyphrase candidates are noun phrase in 1-to-4 grams according to several works (2,3). Noun phrase is identified based on DFA proposed in [35] which regular expression can be seen in (1). This expression incorporates Penn Treebank Part-Of-Speech (POS) notation where POS of each token is obtained using Stamford log-linear part-of-speech tagger [36].

$$(\epsilon+DT)(JJ+JJR+JJS)^*(NN+NNP+NNS+NNPS)+ \quad (1)$$

- b. Keyphrase candidate should not contain stop words as its keyphrase member. This heuristic is inspired from [12]. Stop word list is taken from Snowball stop word list (<http://snowball.tartarus.org/algorithms/english/stop.txt>) with an assumption that such list represents natural language stop words.
- c. Keyphrase candidate should not be recognized as a named entity. This heuristic is applied based on our manual observation from a dataset proposed in [30]. We found that most keyphrases on scientific publications are not related to people, organization, and location name. Named entity is recognized using Stanford Named Entity Recognizer [37].

3.2. Keyphrase Ranking

This phase sorts all keyphrase candidates based on its importance in descending order. Keyphrase importance is defined using TF-IDF weighting (1) described in (2); $tf(t)$ represents term frequency of noun phrase t , $df(t)$ represents document frequency that contain noun phrase t , and N represents the number of documents in collection. TF-IDF is selected as our ranking mechanism due to its very robust performance across different dataset (3).

$$TFIDF(t,D) = tf(t)*(-^2 \log (df(t)/N)) \quad (2)$$

To handle affixes phenomena found on natural language, candidates with similar lemma are merged as one candidate where its score is summed and its lemma is considered as their candidate term. For example, suppose there are two candidates which are networker and networking where TF-IDF score for networker is 1 and TF-IDF score for networking is 2. Since both candidates yield similar lemma (i.e., network), both candidates will be replaced with a candidate called network where its TF-IDF score is $1 + 2 = 3$. Lemma for each candidate is obtained using Stanford CoreNLP [38].

3.3. Keyphrase Classification

After keyphrase candidates are stored on descending order based on their respective importance, each candidate will be popped out from the beginning of the list and fed to a classifier until N keyphrases are selected. Our approach incorporates Deep Belief Networks (6) (DBN) as our classifier since DBN is a deep learning algorithm and deep learning is proven to be more effective than standard learning on various learning task [39]. DBN is commonly used to extract deep hierarchical representation based on given dataset. As suggested by Bengio et al [40], our DBN is also pre-trained with Restricted Boltzmann Machine (RBM) so that its initial node weights are more synchronized with the data itself.

Our classifier incorporates 9 classification features which are listed on Table 1. The first three features represent TF and IDF. Even though most recent works only apply either TF & IDF or TF-IDF since they share considerably similar characteristic [41], we believe that utilizing them at once may yield more representative pattern. SDD, SDS, and PDS are calculated based on its average relative position toward a particular component where each number is pre-normalized based on their respective size to avoid misleading pattern.

Table 1. Classification Features

ID	Feature	Description
TF	Term Frequency	The number of candidate occurrence within given PDF file
IDF	Inverse Document Frequency	Inverse number of candidate occurrence within collection
TFIDF	Term Frequency – Inverse Document Frequency	A numeric value to represent candidate importance toward given document in a collection
SDD	Section Distance in Document	The average number of sections preceding its container section where each number is normalized by the total number of section
SDS	Sentence Distance in Section	The average number of sentence preceding its container sentences where each number is normalized by the total number of sentence in container section
PDS	Phrase Distance in Sentence	The average number of words preceding its phrase occurrence where each number is normalized by the total number of words in container sentence
WC	Word Count	Total words on keyphrase candidate
PL	Phrase Length	Total characters on keyphrase candidate
FSV	Fact-based Sentiment Value	The average fact-based sentiment value of sentence container

Fact-based sentiment is our unique feature which has not been incorporated on other keyphrase extraction works. Its value for each candidate is defined as the average sentiment value for each sentence where the candidate occurs. Sentiment value is obtained using Sentiment Analysis module on Stanford CoreNLP (38). This module returns an integer value ranged from 0 to 4: 0 represents extremely negative; 2 represents neutral; and 4 represents extremely positive. For example, suppose a keyphrase candidate Artificial Neural Network is occurred in two sentences from an article. The first sentence is Artificial Neural Network is easy to use and understand compared to statistical methods whereas the second one is It is hard to interpret the model of Artificial Neural Network since this approach is a black box once it is trained. Based on Sentiment Analysis module on Stanford CoreNLP, the first sentence is assigned as 3 (positive) whereas the second one is assigned as 1 (negative). Thus, fact-based sentiment value of Artificial Neural Network will be $(3 + 1)/2 = 2$ (neutral).

3.4. Learning Model Training

Since DBN requires training data to tune its weight, each article given as a part of training data is converted into 80 training instances; half of them are keyphrases while the rest of them are non-keyphrases. Keyphrases are selected from human-tagged keyphrases on given article. If the number of actual keyphrase is lower than 40, actual keyphrases will be oversampled till their number reaches 40. In contrast, non-keyphrases are selected based on TF-IDF ranking.

It is important to note that we do not include all non-keyphrases as instances based on two rationales. First, the number of non-keyphrases for each article is extremely large and processing all of them may be inefficient. Second, the proportion between keyphrases and non-keyphrases for each article is extremely imbalance. It may generate biased result if all of them are included as training instances.

4. Evaluation

4.1. Evaluating Learning Features

The effectiveness of classification features incorporated in our approach will be evaluated by calculating accuracy difference between default and feature-excluded scheme for each feature. The detail of how to calculate the difference can be seen in (3). It is calculated by subtracting the accuracy of default scheme with the accuracy of feature-excluded scheme. Default scheme represents learning scheme that incorporates all classification features whereas feature-excluded scheme is similar to default scheme except that it excludes target feature. The accuracy for each schema is calculated using 10-fold cross validation while DBN are set with 500 epochs, 0.1 learning rate, 0.9 momentum, and 5 layers with 9 nodes for each layer.

$$acc_diff(f) = default_acc - f_exc_acc(f) \quad (3)$$

For our evaluation dataset, we adapt dataset proposed in (30) which consists of 211 scientific articles. We convert keyphrases for each article to their respective lemma to overcome

affix issues and remove articles without proper keyphrases. As a result, we have 16.320 instances from 204 articles.

Accuracy difference for each feature can be seen on Figure 1. Horizontal axis represents classification features whereas its vertical axis represents accuracy difference values. Several findings can be deducted from such result. First, IDF is the most important feature; it yields the highest accuracy difference. It is natural since IDF represents the uniqueness of keyphrase candidate toward given article in collection. Second, TFIDF can be replaced with TF and IDF since it generates small accuracy difference (even though it is still a positive difference). Third, the variance of our keyphrase phrase length is quite high since PL yields the lowest accuracy difference. Fourth, PL should not be used as a feature since its accuracy difference yields a negative value.

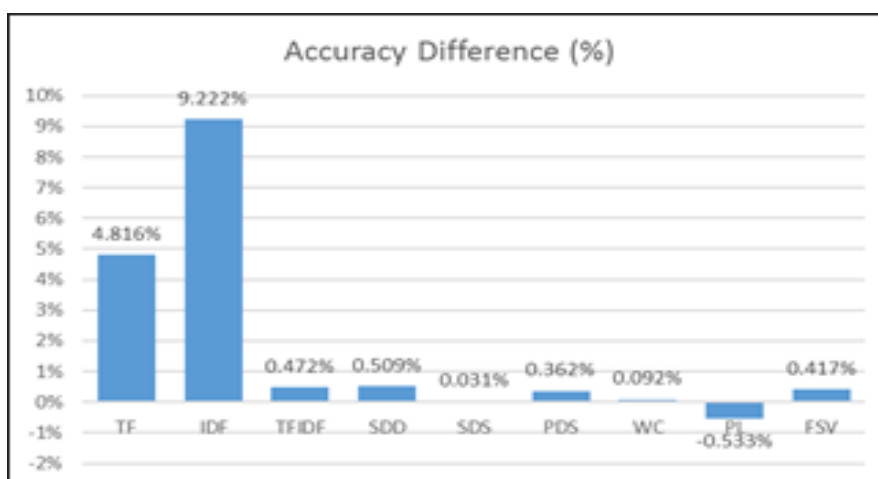


Figure 1. Accuracy difference of classification features

When compared to other classification features, Fact-based Sentiment Value (FSV) is considerably important since its accuracy difference outperforms half of our proposed features. It outperforms SDS, PDS, WC, and PL. In other words, it can be stated that fact-based sentiment is quite effective for differentiating keyphrases from non-keyphrases.

4.2. Evaluating Overall Effectiveness

The overall effectiveness of our proposed approach is measured using standard IR metrics namely precision, recall, and F-measure. Each IR-metric is generated by comparing generated keyphrases with human-tagged keyphrases under three retrieving schemes: Top-5, Top-10, and Top-15. In terms of evaluation dataset, we utilize dataset used for evaluating learning features by deriving it to three datasets: Default Dataset (DD), Occurrence Dataset (OD), and Candidate Dataset (CD).

First, DD is generated by replicating dataset used for evaluating learning features. It is conducted to measure overall effectiveness in general. Second, OD is generated by excluding human-tagged keyphrases that are not found on article content. It is conducted to measure overall effectiveness when selected keyphrase is found on article content. Third, CD is generated by excluding human-tagged keyphrases that are not found on article content or are not recognized as keyphrase candidate through proposed candidate selection heuristic. It is conducted to measure the effectiveness of TF-IDF + DBN for extracting keyphrases.

Evaluation results for overall effectiveness can be seen on Table 2. In all datasets, recall is proportional to the number of retrieved keyphrases yet inversely proportional to precision. Further, their respective harmonic mean (i.e., F-measure) is still lowered when the number of retrieved keyphrases increases. We would argue that both findings are natural considering the number of assigned keyphrases for each article is considerably small (a typical scientific article only has about 3 to 5 keyphrases).

Table 2. Evaluation Metrics for Overall Effectiveness

Dataset	Retrieving Scheme	Precision	Recall	F-Measure
Default Dataset (DD)	Top-5	12.25%	15.23%	13.22%
	Top-10	7.01%	17.57%	9.79%
	Top-15	5.02%	18.93%	7.81%
Occurrence Dataset (OD)	Top-5	12.31%	18.01%	14.01%
	Top-10	7.04%	20.85%	10.18%
	Top-15	5.06%	22.63%	8.05%
Candidate Dataset (CD)	Top-5	12.40%	20.02%	14.62%
	Top-10	7.10%	23%	10.47%
	Top-15	5.10%	24.85%	8.24%

Among three datasets, DD yields the lowest effectiveness, followed by OD and CD respectively. Hence, it can be stated that some keyphrases are not found on article content or excluded as the result of candidate selection heuristics. However, since the differences are considerably small, it can be stated that most keyphrases are still found on its article content and passed our candidate selection heuristics. Our proposed approach yields 13.22% F-measure in Top-5 default scheme when evaluated based on DD. Therefore, it can be stated that our approach is moderately effective considering most keyphrase extraction approaches generate similar F-measure [34].

5. Conclusion and Future Work

In this paper, we have proposed a keyphrase extraction approach that utilizes fact-based sentiment and semi-supervised approach. According to our evaluation, two findings can be deducted. First, fact-based sentiment is quite effective for representing keyphraseness; it ranked as the 5th in terms of accuracy difference. Second, semi-supervised approach is considerably effective to extract keyphrases from scientific articles. it generates moderate F-measure.

For future work, we plan to measure the effectiveness of our approach on different scientific article dataset. We want to know whether its impact is consistent toward various datasets. In addition, we also plan to compare the effectiveness of our approach when compared with other publicly available keyphrase extraction system such as KEA [12] and GenEx [14].

References

- [1] Croft WB, Metzler D, Strohman T. Search engines : information retrieval in practice. Addison-Wesley; 2010: 520.
- [2] Hasan KS, Hasan KS, Ng V. Automatic keyphrase extraction: A survey of the state of the art. In: The 52nd Annual Meeting of the Association for Computational Linguistics. 2014.
- [3] Hasan KS, Ng V. *Conundrums in unsupervised keyphrase extraction: making sense of the state-of-the-art*. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters. 2010: 365–73.
- [4] Kim SN, Medelyan O, Kan M-Y, Baldwin T. Automatic keyphrase extraction from scientific articles. *Language Resources and Evaluation*. 2013 Sep 18; 47(3):723–42.
- [5] Karnalim O. *Software Keyphrase Extraction with Domain-Specific Features*. In: 2016 International Conference on Advanced Computing and Applications (ACOMP). IEEE; 2016: 43–50.
- [6] Le Roux N, Bengio Y. Representational Power of Restricted Boltzmann Machines and Deep Belief Networks. *Neural Computation*. 2008 Jun 17; 20(6):1631–49.
- [7] Mihalcea R, Tarau P. TextRank: *Bringing order into text*. In: Proceedings of the 2004 conference on empirical methods in natural language processing. 2004.
- [8] Liu Z, Huang W, Zheng Y, Sun M. *Automatic keyphrase extraction via topic decomposition*. In: Proceedings of the 2010 conference on empirical methods in natural language processing. 2010: 366–76.
- [9] Wan X, Yang J, Xiao J. Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction. In: ACL. 2007: 552–9.
- [10] Tomokiyo T, Hurst M. *A language model approach to keyphrase extraction*. In: Proceedings of the ACL 2003 workshop on Multiword expressions analysis, acquisition and treatment-Morristown, NJ, USA: Association for Computational Linguistics; 2003: 33–40.

- [11] Zhang C. Automatic keyword extraction from documents using conditional random fields. *Journal of Computational Information Systems*. 2008; 4(3):1169–80.
- [12] Witten IH, Paynter GW, Frank E, Gutwin C, Nevill-Manning CG. *KEA: practical automatic keyphrase extraction*. In: Proceedings of the fourth ACM conference on Digital libraries - DL '99. New York, New York, USA: ACM Press; 1999: 254–5.
- [13] Frank E, Paynter GW, Witten IH, Gutwin C, Nevill-Manning CG. *Domain-specific keyphrase extraction*. In: 16th International Joint Conference on Artificial Intelligence (IJCAI 99). 1999: 668–73.
- [14] Turney PD. Learning Algorithms for Keyphrase Extraction. *Information Retrieval*. 2000;2(4):303–36.
- [15] Ortiz R, Pinto D, Tovar M, Jiménez-Salazar H. *BUAP: An unsupervised approach to automatic keyphrase extraction from scientific articles*. In: Proceedings of the 5th international workshop on semantic evaluation. 2010: 174–7.
- [16] Bordea G, Buitelaar P. *DERIUNLP: A context based approach to automatic keyphrase extraction*. In: Proceedings of the 5th international workshop on semantic evaluation. 2010: 146–9.
- [17] Paukkeri MS, Honkela T. *Likey: unsupervised language-independent keyphrase extraction*. In: Proceedings of the 5th international workshop on semantic evaluation. 2010: 162–5.
- [18] Lu Y, Li R, Wen K, Lu Z. *Automatic keyword extraction for scientific literatures using references*. In: Proceedings of the 2014 International Conference on Innovative Design and Manufacturing (ICIDM). IEEE; 2014: 78–81.
- [19] Nguyen TD, Kan M-Y. Keyphrase Extraction in Scientific Publications. In: Asian Digital Libraries Looking Back 10 Years and Forging New Frontiers. Berlin, Heidelberg: Springer Berlin Heidelberg; 2007: 317–26.
- [20] Pasquier C. *Task 5: Single document keyphrase extraction using sentence clustering and Latent Dirichlet Allocation*. In: Proceedings of the 5th international workshop on semantic evaluation. 2010: 154–7.
- [21] Putra JWG, Khodra ML. Rhetorical Sentence Classification for Automatic Title Generation in Scientific Article. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2017 Jun 1; 15(2): 656–64.
- [22] Li J, Cao Q. DSRM: An Ontology Driven Domain Scientific Data Retrieval Model. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2014 Feb 1; 12(2).
- [23] Subroto IMI, Sutikno T, Stiawan D. The Architecture of Indonesian Publication Index: A Major Indonesian Academic Database. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2014 Mar 1; 12(1): 1–5.
- [24] Hendra H, Jimmy J. Publications Repository Based on OAI-PMH 2.0 Using Google App Engine. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2014 Mar 1; 12(1): 251–62.
- [25] HaCohen-Kerner Y. *Automatic Extraction of Keywords from Abstracts*. In: International Conference on Knowledge-Based and Intelligent Information and Engineering Systems. Springer, Berlin, Heidelberg; 2003: 843–9.
- [26] Bhowmik R. *Keyword extraction from abstracts and titles*. In: IEEE SoutheastCon 2008. IEEE; 2008: 610–7.
- [27] Eichler K, Neumann G. *DFKI KeyWE: Ranking keyphrases extracted from scientific articles*. In: Proceedings of the 5th international workshop on semantic evaluation. 2010: 150–3.
- [28] Nguyen TD, Luong MT. *WINGNUS: Keyphrase extraction utilizing document logical structure*. In: Proceedings of the 5th international workshop on semantic evaluation. 2010: 166–9.
- [29] Ouyang Y, Li W, Zhang R. *273. Task 5. keyphrase extraction based on core word identification and word expansion*. In: Proceedings of the 5th international workshop on semantic evaluation. 2010: 142–5.
- [30] Nguyen TD, Kan M-Y. Keyphrase Extraction in Scientific Publications. In: Asian Digital Libraries Looking Back 10 Years and Forging New Frontiers. Berlin, Heidelberg: Springer Berlin Heidelberg; 2007: 317–26.
- [31] Treeratpituk P, Teregowda P, Huang J, Giles CL. *Seerlab: A system for extracting key phrases from scholarly documents*. In: Proceedings of the 5th international workshop on semantic evaluation. 2010: 182–5.
- [32] Berend G, Farkas R. *SZTERGAK: Feature engineering for keyphrase extraction*. In: Proceedings of the 5th international workshop on semantic evaluation. 2010: 186–9.
- [33] HaCohen Kerner Y, Gross Z, Masa A. *Automatic Extraction and Learning of Keyphrases from Scientific Articles*. In: International Conference on Intelligent Text Processing and Computational Linguistics. Springer, Berlin, Heidelberg; 2005: 657–69.
- [34] Kim SN, Medelyan O, Kan M-Y, Baldwin T. Automatic keyphrase extraction from scientific articles. *Language Resources and Evaluation*. 2013 Sep; 47(3): 723–42.
- [35] Sarkar K, Nasipuri M, Ghose S. A New Approach to Keyphrase Extraction Using Neural Networks. *International Journal of Computer Science*. 2010; 7(2).
- [36] Toutanova K, Klein D, Manning CD, Singer Y. *Feature-rich part-of-speech tagging with a cyclic dependency network*. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. 2003: 173–80.

-
- [37] Finkel JR, Grenager T, Manning C. *Incorporating non-local information into information extraction systems by gibbs sampling*. In: Proceedings of the 43rd annual meeting on association for computational linguistics. 2005: 363–70.
- [38] Manning CD, Surdeanu M, Bauer J, Finkel JR, Bethard S, McClosky D. The stanford corenlp natural language processing toolkit. In: ACL (System Demonstrations). 2014: 55–60.
- [39] LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*. 2015 May 28; 521(7553): 436–44.
- [40] Bengio Y, Lamblin P, Popovici D, Larochelle H. *Greedy layer-wise training of deep networks*. In: Advances in neural information processing systems. 2007: 153–60.
- [41] Hulth A. *Improved automatic keyword extraction given more linguistic knowledge*. In: Proceedings of the 2003 conference on Empirical methods in natural language processing. 2003: 216–23.