*Alkalmazott Nyelvészeti Közlemények, Miskolc, XVI. évfolyam, 2. szám (2023), pp. 102–115.*

# KEYWORDS AT THE BORDER OF TERMINOLOGY AND LANGUAGE TECHNOLOGY

# KULCSSZAVAK A TERMINOLÓGIA ÉS NYELVTECHNOLÓGIA HATÁRÁN

RÉKA DODÉ – NOÉMI VADÁSZ – KINGA JELENCSIK-MÁTYUS[1]

**Abstract:** In the present study we compared manually given keywords of scientific articles in Hungarian with keywords extracted using the Sketch Engine tool. This is a pilot study for a larger work, the output of which is a training material that will be suitable for fine-tuning a Hungarian language model for the task of keyword and term extraction. In the current phase of the research, we used approximately 1,000 academic articles to reveal how different a person's keyword input strategy is from frequency-based keyword extraction methods.

***Keywords:*** *keyword, term, keyword extraction, Sketch Engine*

**Absztrakt:** Jelen tanulmányban a tudományos cikkek magyar nyelvű, manuálisan megadott kulcsszavait hasonlítottuk össze a Sketch Engine eszközzel kinyert kulcsszavakkal. Ez egy nagyobb munka pilottanulmánya, melynek eredménye egy olyan oktatóanyag, amely alkalmas lesz egy magyar nyelvi modell finomhangolására kulcsszó- és terminuskinyerési feladatra. A kutatás jelenlegi szakaszában körülbelül 1000 tudományos cikket használtunk fel annak feltárására, hogy egy személy kulcsszóbeviteli stratégiája mennyiben különbözik a gyakoriságon alapuló kulcsszókivonási módszerektől.

***Kulcsszavak:*** *kulcsszó, terminus, kulcsszókinyerés, Sketch Engine*

## 1. Introduction

Keyword extraction is basically a text analysis technique for applications of Natural Language Processing, Information Retrieval, and Text Mining (Firoozeh et al. 2020). However, language technologists use the very same methods in term extraction as in keyword extraction, although – despite basic similarities – terms and keywords are different concepts. Terminologists, then, try to work with these term candidates extracted using statistical or hybrid methods (to put them in terminology databases, in a concept system, to find a matching definition, or to find an equivalent in another language), thus, no wonder they need to do a lot of post-editing. The concept system

[1] RÉKA DODÉ – NOÉMI VADÁSZ – KINGA JELENCSIK-MÁTYUS
Nyelvtudományi Kutatóközpont
dode.reka@nytud.hu
vadasz.noemi@nytud.hu
jelencsik-matyus.kinga@nytud.hu

is a "set of concepts structured in one or more related domains according to the concept relations among its concepts" (ISO 1087). The problem is that certain terms are very rarely, or not at all mentioned in texts, so they cannot be extracted using methods based on occurrence statistics. Terminologists strive to be able to include the entire conceptual set and terminology of a (sub)domain in their database, so it is important to obtain all terms from a text, even those that occur very rarely.

Keywords and their function are known to everyone who works in science and publishes. When writing scientific publications, it is common practice in several fields that authors provide keywords to help the reader. The functions of keywords are: to give insight into the content of the text, to represent it, and to facilitate easier navigation between academic texts, and, at the same time, keywords are also used to enhance visibility (through search engines).

The author of a study is an expert in that particular field. And as a specialist, he or she has solid background knowledge and provides the keywords based on this conceptual background knowledge. The author has a strategy, complex knowledge and an assumption that the reader also has background knowledge in the given subject area. The keywords given by the author will thus definitely be part of the conceptual structure of a given academic field and will not only be included among the keywords based on their occurrence. In this sense, we can examine these keywords using a concept-oriented approach.

The aim of the present research is to compare the manually given keywords with the keywords extracted from scientific papers using the relevant functions of Sketch Engine (Kilgarriff et al. 2004; Kilgarriff et al. 2014). The results may help to find the best training and evaluation data for neural automatic keyword and term extraction. A neural model trained on such data could show better results in finding not only the statistically frequent keywords, but also further elements of the concept system – which, until now, could be only given by the author, as an expert of the academic field.

The Library and Information Centre of the Hungarian Academy of Sciences has set up a repository at the request of OTKA (National Scientific Research Basic Programs) to store Hungarian scientific articles and research reports prepared with OTKA support: this is the REAL repository (REpository of the Academy's Library) (W9). In the pilot research we compile a corpus using 1,000 scientific papers of the REAL repository. With the help of this corpus, we fine-tune the HuBERT model (Nemeskey 2021) to support keyword extraction for Hungarian.

Manually given keywords have already been used in the Hungarian language for fine-tuning models built from media texts (Yang et al. 2020). Our aim is to use similar methods for academic texts.

### *1.1. Motivation*

Within the framework of a project of the Hungarian Academy of Sciences, the Language Technology Research Group of the Hungarian Linguistics Research Centre makes the material of the REAL repository searchable in a more efficient

way through the application of language technology. The project also includes terminological and scientific field classification tasks, where terms are extracted from the scientific texts of the repository.

Dodé (2023) examined the manually entered keywords of 20 papers, randomly selected from the REAL repository studying the following questions: 1. number of keywords 2. their (internal) structure, 3. additional semantic characteristics (e.g. proper nouns) 4. number of occurrences, 5. place of occurrence [in title, (sub)heading, beginning of sentence, before parenthesis, etc.], 6. collocations of all forms of keywords found in the text, and thus the mapping of generic-specific relationships, 7. if they can be considered as terms (examined applying several term definitions). The definition of term according to ISO standard is a "designation that represents a general concept by linguistic means" (ISO 1087). While terminology is a "set of designations and concepts belonging to one domain or subject" (ISO 1087). However, there are many other definitions according to the purposes for which the terminology is used (e.g. for translation) (Kis 2005; Foo 2009).

The research of Dodé (2023) revealed that the examined (manually given) keywords include all the three components from Cabré (2003): the cognitive (concept system of an academic field), the linguistic (lexical unit), and the socio-communicative component (professional communication), as well as other elements, which leads to the conclusion that we can really consider keywords as terms (e.g. the appearance of a definition or the appearance of the concept in another language and as a synonym). The study pointed out several important things. Firstly, each keyword can be considered a term based on at least one term definition. The texts are scientific and academic texts, therefore the socio-communicative component (Cabré 2003) is given. This means, among other things, that providing keywords presupposes conceptual (academic) knowledge not only for the writer, but also for the reader. When examining the keywords, conceptual relationships also appeared e.g. *galamb* (pigeon), *postagalamb* (homing pigeon), *versenygalamb* (racing pigeon) (for concept relations, see ISO 704), which, according to Cabré (2003), is the cognitive component. The third component is the linguistic component: terms are lexical units and behave like common words. At the same time, 37% of the examined keywords occur less than twice in the texts and 23% do not appear in the texts at all, as they are only listed among the keywords. Dodé (2023) counted the occurrences by searching for the keywords as provided (all forms with suffixes were counted, but derived forms were excluded).

Our hypothesis – from the observation mentioned above – was that the overlap between the keywords extracted by Sketch Engine and the ones given manually is going to be small.

### *1.2. Motivation from the point of terminology work*

Term extraction is an integral part of terminology work, and is an important and time-consuming task. When creating a term extraction application trained with

supervised machine learning methods, or in the case of fine-tuning deep neural networks, training material is also needed.

The keywords provided to the text by the author represent the content of the text the most effectively, since their keyword strategy is not based on word frequency in the text, or because it does not stick to the vocabulary of the text. Furthermore, the author of the text gives these expressions having strong professional background knowledge (Hulth 2003; Dodé 2023). Based on these terms (provided by the author), we can map the conceptual structure (or concept system) of the given text in the most comprehensive way. The concept system makes it easier to understand and define a concept and is also used for conceptual harmonization, which is an important subtask of terminology work. Concept harmonization is an "activity leading to the establishment of a correspondence between two or more closely related or overlapping concepts having professional, technical, scientific, social, economic, linguistic, cultural or other differences, in order to eliminate or reduce minor differences between them" (ISO 860: 1).

It is therefore worth dealing with manually entered keywords. With this motivation, in this research we consider the corpus of manually entered keywords as the gold standard.

## 2. The pilot corpus

The corpus of this pilot study was compiled using the articles of the REAL repository, a part of which is available for everyone to browse and download. In the present research, for the sake of simplicity, the materials of the corpus were selected from publicly available materials. The advanced search function was used to narrow our findings. From the possible 17 fields we only used two:

- **date:** only papers published after 2010 were selected, with the premise that since around that time it has become common practice/requirement to provide keywords.
- **type:** only articles were selected, because our research is limited to this genre.

The result was downloaded in JSON format in order to apply some further filtering. The file contained a field defining the language of the paper However, after checking the texts, it turned out that the language code was not set correctly: the value of the language field was set to Hungarian whereas the text was English. Therefore, we did not find the language element in the JSON file reliable, so we used a language detector (langdetect library in Python). Our aim was also to find the Hungarian texts which begin with a longer English section (like an English abstract). To reach this goal, the first 1 million characters of each paper were entered into the language detector. This way we got 29,502 files, which were articles in Hungarian published after 2010 on various scientific topics. We filtered these texts with the *grep* command to see which ones contain the typical pattern of keywords. In the end, we

were left with 9,226 articles. In the current phase of the pilot study, 1,146 random texts were selected.

It is important to note that in its present state the REAL repository contains OCR-ed documents. OCR clean up is also part of the project mentioned in 1.1, but, in the current research only dirty OCR files were available to us. We only applied minimal correction and cleaning, so it should be kept in mind that in order to improve performance, more thorough OCR correction must be carried out on the material.

## 3. Keywords in academic papers

There are basic conceptual differences between extracting keywords using statistical methods and when keywords are given by the author of an article. Let us have a look at the strategies and know-hows of giving keywords manually.

When publishing in professional journals, the name of the general academic field and its terms are not given as keywords, although they are inevitably included in the article. For example, language technology, terminology, etc. were not included among the keywords for this article. Although the name of the general academic field and its terms would help the reader/searcher to find out what the topic is, there are a number of reasons why they are not listed as keywords:

- the writer assumes that the reader has some knowledge of the subject
- more specific keywords automatically "bring in" the general concepts
- academic papers are usually published in the discipline's own publications

We have checked the guidelines of some prominent international academic journals, focusing on what they expect from the authors regarding keywords.

All the e-journals of John Benjamins (W1) expect authors to give up to 10 keywords when submitting an article. The guideline suggests topics for keywords: languages, methods, frameworks. It also mentions that the abstract should also contain the most important keywords of the study.

Taylor and Francis (W2) online also publishes numerous academic journals. When talking about keywords, it emphasises that they are used for indexing purposes both on Taylor and Francis online, and on general search engines. It also highlights that keywords play an important role in making the articles visible and easier to find for other scientists.

One of the most prominent groups of Hungarian academic journals is the AK Journals, publishing almost 50 periodicals. All of the journals have their separate submission guidelines. However, they are rather similar on the rules of providing keywords: for example the tutorial of Across Languages and Cultures (W3), like most of the journals, mentions only the number of keywords (4-6) expected from the authors, their place in the article, and nothing more.

We also checked the keyword giving practices of academic publishers in book chapters. The guidelines of Springer (W4) highlight that providing keywords is a tool to help indexers and search engines find relevant papers, and the keywords

represent the content and specify the field of expertise. The guideline shows examples of good and bad keywords typically emphasizing that a keyword is good if it is specific enough (e.g. *climate change* and *erosion* vs. *quaternary climate change* and *soil erosion*).

Oxford University Press (W5) also gives a detailed description on providing keywords. Apart from the basic ideas mentioned above, it also points out that properly selected keywords "help generate links to and from relevant content".

The guideline of MIT Press (W6) suggests that keywords should describe the content, themes, concepts, and are also used to enhance visibility – firstly because keywords have a strong impact on search ranking, and secondly, because end users might search for keywords – assuming common knowledge between the author and the reader. It is quite unique that the limit of keywords is given in characters: no more than 500-600 characters, preferably single-word or 2-3-word long expressions should be given.

### *3.1. Manually given keywords in REAL papers*

To learn more about manually entered keywords, we used texts from the REAL repository explained in Section 2. A simple processing script performed the following tasks. It selected the lines containing the string ***Kulcsszavak:*** (Keywords:) and extracted the keywords separated by a comma or semicolon. If the line ended with a separator character or a hyphen, the next line was also included, and so on.

Texts containing the string *Kulcsszavak:* more than once were excluded. These texts are typically publications in some sort of a collection, e.g. abstract volumes, therefore, they are not suitable to be used in our keyword extraction experiments, nor to be used as training data. Further files were excluded if our script could not parse the keywords properly due to an OCR error. In the end, a total of 1,046 articles remained.

The processing script has two outputs. Firstly, it produces a TSV file that contains the keywords that were entered manually in each article (gold standard set). Secondly, it also produces a version of the articles without the manually entered keywords. The latter one needed some further processing: we tried to mitigate the errors of dirty OCR files, so we fixed sentence segmentation with quntoken (W8). These outputs are suitable to be used as training data. The average number of the manually entered keywords was 4.7; minimum 1 and maximum 40 keywords were given for each text.

## 4. Keyword extraction methods

The issue of keywords is not simple, it is very diverse and depends on many things. It is no coincidence that many different methods have appeared over the years. Nomoto (2023) collected these methods accurately in his study and projected new possibilities. Based on this, we also briefly present the keyword extraction methods. It is important to note that the following list does not necessarily represent improvement in quality, but they are rather different perspectives.

1. Examples of statistical methods (frequency and co-occurrence): TFIDF (Salton–Yang 1973, occurrence in focus corpus compared to occurrence in reference corpus) is still relevant in the field of information retrieval. In this case, we are working with index terms that are present in a document, appearing frequently across multiple documents, and demonstrating a recognizable distributional pattern. Another example is discrimination value analysis (Salton et al. 1974) according to which a good term is able to separate documents. Furthermore, there is an extraction method using graphs: the keygraph (Ohsawa et al. 1998), where nodes and edges (as used in graphs) show the degree of association between a pair of words, determined by their semantic proximity.
2. Rule-based extraction (used in computational linguistics) looks for syntactic patterns. One version is head-driven keyword extraction (Barker–Cornacchia 2000), which searches for groups of nouns (noun phrase – NP), and identifies keywords as NPs containing the most frequent nouns as heads.
3. A new perspective, called TextRank (Mihalcea–Tarau 2004) is very similar to keygraph, but also takes into account the weight of contextual words. However, it does not have access to word frequency, which is an important factor.
4. External knowledge: In MAUI (Medelyan 2009) the process involves normalizing a term using external knowledge, coming from Wikipedia (a contextually relevant wiki page). It then has an option to produce keywords. From this point on, these methods use deep learning.
5. Classificatory keyword extraction is used by methods which, by sequentially examining consecutive segments of a text for specific keywords, analyze each word and decide whether to add it to a pool of potential keywords. It treats the text as a network of words, with the power of an association represented by the frequency of co-occurrence and uses the latent representations (Florescu–Jin 2018).
6. Deep learning (DL) turns our focus to the generation of keywords. It has two parts: the Encoder encodes the source text and the Decoder generates keywords. It is able to reuse parts of the input as it generates a keyword and it is also capable of building out-of-document keywords (Nomoto 2023).
7. Text classification: by expanding the range of vocabulary it encompasses, we can transform this method from being limited to a fixed set of topics into a keyword extractor.
8. Working with textual cues: RAKE (Rose et al. 2010), to extract keywords, using stop words the text is divided into contiguous word sequences, and the ones that occur most frequently between the stop words are selected.
9. Unsupervised deep learning: YAKE identifies keywords not by how important they are (funcion word). LDA (Blei et al. 2003) builds a language model based on the premise that there is an implicit set of topics that defines the distribution of words observed in a document. TopicalPageRank (Liu et al. 2010) aims to combine PageRank and LDA.

### *4.1. Keyword and term extraction with Sketch Engine*

Sketch Engine has a keyword and term extraction function. These functions of Sketch Engine are used by many lexicologists, lexicographers, terminologists, translators, linguists etc. Sketch Engine defines *term* for its own application and the statistical method it uses. This corresponds to what Jacquemin–Bourigault (2003) say, that in corpus-based terminology, the term is the output of terminological analysis. Jacquemin and Bourigault (2003) think that the classic term definition cannot be applied from a term extraction perspective. Sketch Engine defines term as follows (W7):

> A term is a multi-word expression ... which appears more frequently in focus corpus compared to another (reference) corpus and, at the same time, the expression has a format of a term in the language. The format is defined in a term grammar which is specific for each language... A term grammar is a set of rules written in CQL which define the lexical structures...typically noun phrases…

The keyword in Sketch Engine, however, is a single-token item. The first part of the definition is the same as the term definition, i.e. what appears more often in the focus corpus than in the reference corpus. However, there are no additional grammatical (morphosyntactic) restrictions.

Sketch Engine uses statistical methods for extraction. It determines the *keyness score* with normalized (per million) frequencies. The methods are explained in detail in (Kilgarriff 2009). To extract terms, Sketch Engine uses a term grammar, based on predefined (lexical) rules. Sketch Engine allows us to define our own term grammar for valid structures such as adjective + (optional) adjective + noun. Terms can be extracted only from tagged and lemmatized corpora. If there is no specific term grammar, Universal Word Sketch grammar is used. Sketch Engine provides 5 values for keywords and terms as results: frequency in focus corpus, relative frequency in focus corpus, frequency in reference corpus, relative frequency in reference corpus and keyness score. The results are sorted by the score.

In our experiment we used the output of the processing script introduced in 3.1: articles from the REAL repository, from which we cut out the manually entered keywords and which were further improved with quntoken.

With the help of the Sketch Engine API, the keywords and terms were extracted from the texts. The keywords and terms were sorted along the keyness score and the two lists were merged into one, because we did not want to treat single-word keywords and multi-word terms as separate entities.

## 5. Results and discussion

Our hypothesis was that the overlap between the keywords extracted by Sketch Engine and the ones given manually is going to be rather small. The motivation of the hypothesis is the observation mentioned in Section 1, namely that the keywords

given manually occur less frequently in the articles than expected, so frequency-based metrics, like tf-idf, do not seem reliable.

Evaluating term and keyword extraction is not an easy task and can be done from several perspectives. In addition, Sketch Engine, like other statistical methods, only provides one keyness score, but not the threshold above which an expression is considered a term or a word is considered a keyword. We also have to deal with the order of the resulting terms and keywords. In the case of manually entered keywords, we did not calculate with order, because although there is probably a strategy (guide) behind listing the keywords, we cannot declare that the first one is the most important, the second one is the second most important, etc. Manually entered keywords are therefore considered as a set in which all elements are equally important.

The number of the manually given keywords varies from article to article, while Sketch Engine outputs a predefined number of keywords for an article, so we had to make several comparisons. First, we counted the minimum, maximum and average number of manually entered keywords. As it was mentioned in 3.1, we found that minimum 1 and maximum 40 keywords were given manually, and the average number of the given keywords is 4.6. Accordingly, we performed four experiments: firstly, we used the keyword or term with the highest score, secondly, we used the top five ones (based on the average number of the given keywords), thirdly, we used the top 40 ones and finally we used a dynamic-sized subpart of the list produced by Sketch Engine for each file, adjusted to the size of the list of manually given keywords. These cuts were used with two types of evaluation metrics that we present below.

In the first round, we checked how similar the resulting word list was to the manually entered list. This study, therefore, does not take into account the fact that Sketch Engine sorts the keywords and terms based on the keyness score. In this study, we treated the keywords issued by the Sketch Engine not as a list, but as a set, and examined how big their intersection was with the set of manually entered keywords. We examined the size of the intersection of the two sets (manually given, and issued by Sketch Engine) and compared it with the size of the list of manually entered keywords. The results calculated for Sketch Engine lists with different numbers of elements described above can be seen in *Table 1*. The average column contains the average calculated for all documents, and the maximum is the best result achieved with the Sketch Engine list of the given size.

***Table 1.*** *The size of the intersection of the keyword sets entered manually and issued by Sketch Engine concerning sets of different sizes compared to the size of the list of the manually given keywords*

| | average | best |
|---|---|---|
| top 5 | 0.042 | 0.200 |
| top 1 | 0.096 | 1.000 |
| top 40 | 0.011 | 0.025 |
| dynamic | 0.047 | 0.500 |

The results show that the section size is on average very small, compared to the size of the gold set. The best result could be achieved, not surprisingly, with the single-element Sketch Engine list, but it is clear that the average result calculated for all documents is very low even in this test. However, it is important to note that examining the section size alone is not a sufficiently revealing metric when evaluating the task of keyword extraction.

Precision, recall, and F1-measure are often given, but we did not consider keyword extraction as a classification task, and in the case of multi-word keywords, it is difficult to determine false positives, so we discarded these metrics. Instead, we used a conceptually simpler metric. The metric we chose expresses how many items in the list produced by the sketch engine were included in the gold standard set (the manually entered keyword list). The ranking of the examined keyword or term in the list issued by Sketch Engine is factored into the metric. In the case of hits, the keywords that appeared higher in the list received a higher score (closer to one) if they were also included in the gold set, while those that appeared further back received a lower score (closer to zero). For example, if we look at the first five items in the Sketch Engine list, the first item gets 5/5 points, i.e. one if it is in the manual list, the second item gets 4/5, the third item gets 3/5, etc. And if the keyword was not included in the gold set, it did not receive any points. The points were added up and divided by the size of the gold set, so each file received a number between 0 and 1. The results are shown in *Table 2*.

***Table 2.*** *Rank weighted accuracy on the Sketch Engine lists of different sizes*

| | average | best |
|---|---|---|
| top 5 | 0.034 | 0.400 |
| top 1 | 0.096 | 1.000 |
| top 40 | 0.009 | 0.050 |
| dynamic | 0.038 | 0.500 |

The low numbers show that there are only a few gold keywords that got high keyness scores in Sketch Engine. There may be several reasons behind the low agreement.
A closer look at the extracted words reveals that the Sketch Engine often marks English terms and keywords. This may be because the articles often contain abstracts in English, and English words in the reference corpus probably occur with a low frequency. In the Sketch Engine list, you often come across unknown, non-existent word forms that can be traced back to the OCR files. In these cases too, Sketch Engine received a low frequency in the reference corpus. In addition, you also encounter cases where the manually extracted keyword list presumably contains forms with an OCR error.

In these cases, even though the Sketch Engine would find the same keyword, there is no match due to the error. In addition to all this, of course, the reason for the low match may be caused by the keyword-giving strategy of the article writers. Frequency-based keyword extraction techniques are probably not able to model

human keyword entry strategies, and we have not even taken into account that the keyword entry strategy may differ from person to person, nor that the purpose of entering keywords can be multiple.

And if we can capture human strategy more effectively with this training material (for fine-tuning huBERT), we hope that we can develop an application that supports the work of terminologists more efficiently.

The results presented above show that our hypothesis was confirmed: the overlap between the gold set and the Sketch Engine keywords is small, due to the differences between the manual and the statistical methods and the reasons mentioned in the above paragraphs.

## 6. Future work

In order to obtain more accurate results, our experiment must be repeated after the OCR clean up of the texts. We also plan to expand our experiments to additional texts. We will also try to see what happens if we remove the abstracts (both English and Hungarian) from the training corpus.

With the cleaned texts and manually extracted keywords, our goal is to compile a training corpus with which we can fine-tune the huBERT (Nemeskey 2020, 2021) language model for the task of keyword and term extraction.

## References

Barker, Ken – Cornacchia, Nadia 2000. Using noun phrase heads to extract document keyphrases. In: Hamilton, Howard J. (ed.): *Advances in artificial intelligence. Canadian AI 2000. Lecture notes in computer science*, Vol. 1822. Berlin–Heidelberg: Springer. 40–52.

Blei, David M. – Ng, Andrew Y. – Jordan, Michael I. 2003. Latent dirichlet allocation. *The Journal of Machine Learning Research* 3, 993–1022.

Cabré Castellví, Maria 2003. Theories of terminology: Their description, prescription and explanation. *Terminology* 9, 163–199.

Dodé Réka 2023. Kulcsszavak és terminusok vizsgálata a REAL repozitóriumának anyagán – pilot kutatás. *Knowledge sharing, information management, applicability*. 29th Congress of Hungarian Applied Linguistics Budapest, 17–18. March 2023. Department of Languages for Specific Purposes, Semmelweis University. (oral presentation)

Firoozeh, Nazanin – Nazarenko, Adeline – Alizon, Fabrice – Daille, Béatrice 2020. Keyword extraction: Issues and methods. *Natural Language Engineering* 26/3, 259–291.

Florescu, Corina – Jin, Wei 2018. Learning feature representations for keyphrase extraction. *Proceedings of the AAAI Conference on Artificial Intelligence.* 32. https://arxiv.org/abs/1801.01768 (Downloaded: 27. 6. 2023).

Foo, Jody 2009. *Term extraction using machine learning.* Linkoping: Linkoping University. https://www.researchgate.net/profile/Jody-Foo/publication/255638371_Term_extraction_using_machine_learning/links/5557937e08ae6943a874b19f/Term-extraction-using-machine-learning.pdf (Downloaded: 30. 5. 2023).

Hulth, Anette 2003. Improved automatic keyword extraction given more linguistic knowledge. In: Collins, Michael – Steedman, Mark (eds.): *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing.* Barcelona, Spain: Association for Computational Linguistics. 216–223. https://www.aclweb.org/anthology/W03-1028 (Downloaded: 15. 6. 2023).

Jacquemin, Christian – Bourigault, Didier 2003. Term extraction and automatic indexing. In: Ruslan, Mitkov (ed.): *The Oxford handbook of computational linguistics.* Oxford: Oxford University Press. 599–615.

Kilgarriff, Adam 2009. Simple maths for keywords. In: Mahlberg, Michaela – González-Díaz, Victorina – Smith, Catherine (eds.): *Proceedings of the Corpus Linguistics Conference.* Liverpool, UK: University of Liverpool. https://ucrel.lancs.ac.uk/publications/cl2009/ (Downloaded: 20. 2. 2023).

Kilgarriff, Adam – Baisa, Vit – Bušta, Jan – Jakubíček, Miloš – Kovář, Vojtěch – Michelfeit, Jan – Rychlý, Pavel – Suchomel, Vít 2014. The Sketch Engine: Ten years on. *Lexicography* 1, 7–36.

Kilgarriff, Adam – Rychlý, Pavel – Smrž, Paveé – Tugwell, David 2004. The Sketch Engine. In: Williams, Geoffrey – Vessier, Sandra (eds.): *Proceedings of the 11th EURALEX International Congress*. Bretagne: Université de Bretagne-Sud, Faculté des lettres et des sciences humaines. 105–115.

Kis Balázs 2005. Automatikus terminológiakeresés számítógéppel – kísérlet. *Fordítástudomány* 7/1, 84–97. https://www.epa.hu/04100/04125/00001/pdf/EPA04125_forditastudomany_2005_1.pdf (Downloaded: 30. 5. 2023).

Liu, Zhiuyan – Huang, Wenyi – Zheng, Yabin – Sun, Maosong 2010. Automatic keyphrase extraction via topic decomposition. In: *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing.* Cambridge: UK: Association for Computational Linguistics. 366–376.

Medelyan, Olena 2009. *Human-competitive automatic topic indexing.* PhD thesis, Department of Computer Science, The University of Waikato.

Mihalcea, Rada – Tarau, Paul 2004. Textrank: Bringing order into texts. In: *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing.* Barcelona, Spain: Association for Computational Linguistics. 404–411.

Nemeskey, Dávid Márk 2020. *Natural language processing methods for language modeling.* PhD thesis. Budapest: Eötvös Loránd Tudományegyetem.

Nemeskey, Dávid Márk 2021. Introducing huBERT. In: Berend Gábor (szerk.): *XVII. Magyar Számítógépes Nyelvészeti Konferencia*. 3–14.

Nomoto, Tadashi 2023. Keyword extraction: A Modern Perspective. *SN Computer Science* 4, article 92. https://link.springer.com/article/10.1007/s42979-022-01481-7 (Downloaded: 20. 6. 2023).

Ohsawa, Yukio – Benson, Nels E. – Yachida, Masahiko 1998. Keygraph: Automatic indexing by co-occurrence graph based on building construction metaphor. In: *Proceedings of the Advances in Digital Libraries Conference.* 12–18. https://ieeexplore.ieee.org/xpl/conhome/5492/proceeding.

Rose, Stuart – Engel, David – Cramer, Nicholas – Cowley, Wendy E. 2010. Automatic keyword extraction from individual documents. In: Berry, Michael. W. – Kogan, Jacob (eds.): *Text mining: Applications and theory.* Hoboken: John Wily & Sons Ltd. 1–20.

Salton, Gerard – Yang, Chung-Shu 1973. On the specification of term values in automatic indexing. *Journal of Documentation* 29/4, 351–372.

Salton, Gerard – Yang, Chung-Shu – Yu, Clement T. 1974. A theory of term importance in automatic text analysis. *Journal of American Society for Information Science* 26/1, 33–44.

Yang, Zijian Győző – Novák, Attila – Laki, László János 2020. Automatic tag recommendation for news articles. In: Kovásznai Gergely – Fazekas István – Tómács Tibor (eds.): *Proceedings of the 11th International Conference on Applied Informatics.* Eger: Eszterházy Károly Katolikus Egyetem. 442–451.

W1. *John Benjamins.* https://benjamins.com/downloads/guidelines/jb-guidelines-manuscript-submission-apa.pdf (Downloaded: 30. 5. 2023).

W2. *Taylor and Francis.* https://authorservices.taylorandfrancis.com/publishing-your-research/writing-your-paper/using-keywords-to-write-title-and-abstract/ (Downloaded: 30. 5. 2023).

W3. *AK Journals.* https://akjournals.com/fileasset/author-guidelines/Across_IfA.pdf (Downloaded: 30. 5. 2023).

W4. *Springer.* https://www.springer.com/gp/authors-editors/authorandreviewertutorials/writing-a-journal-manuscript/title-abstract-and-keywords/10285522 (Downloaded: 30. 5. 2023).

W5. *Oxford University Press.* https://academic.oup.com/pages/authoring/books/ preparing-your-manuscript/abstracts-and-keywords (Downloaded: 30. 5. 2023).

W6. *MIT Press.* https://mitpress.mit.edu/guidelines-preparing-abstracts-and-keywords/ (Downloaded: 30. 5. 2023).

W7. *Sketch Engine User Guide.* https://www.sketchengine.eu/guide/ (Downloaded: 30. 5. 2023).

W8. *quntoken.* https://github.com/nytud/quntoken (Downloaded: 30. 5. 2023).

**Source list**

ISO 1087 = ISO Central Secretary 2019. *Terminology work and terminology science – Vocabulary*. Standard, International Organization for Standardization, Geneva, CH, https://www.iso.org/standard/62330.html (Downloaded: 20. 2. 2023).

ISO 704 = ISO Central Secretary 2022. *Terminology work – Principles and methods.* Standard, International Organization for Standardization, Geneva, CH, https://www.iso.org/standard/79077.html (Downloaded: 20. 2. 2023).

ISO 860 = ISO Central Secretary 2007. *Terminology work – Harmonization of concepts and terms*. Standard, International Organization for Standardization, Geneva, CH, https://www.iso.org/standard/40130.html (Downloaded: 20. 2. 2023).

W9. http://real.mtak.hu/ (Downloaded: 30. 5. 2023).