Keyword Extraction in German: Information-theory vs. Deep Learning

Max Kölbl[a], Yuki Kyogoku, J. Nathanael Philipp[b], Michael Richter[c], Clemens Rietdorf and Tariq Yousef[d]

Institute of Computer Science, NLP Group, Universität Leipzig, Augustusplatz 10, 04109 Leipzig, Germany
{*jonas_nathanael.philipp, tariq.yousef*}*@uni-leipzig.de, richter@informatik.uni-leipzig.de,*
{*max.w.koelbl, kyogoku11, clemens.rietdorf*}*@gmail.com*

Keywords: Keyword Extraction, Information Theory, Topic Model, Recurrent Neural Network.

Abstract: This paper reports the results of a study on automatic keyword extraction in German. We employed in general two types of methods: (A) an unsupervised method based on information theory (Shannon, 1948). We employed (i) a bigram model, (ii) a probabilistic parser model (Hale, 2001) and (iii) an innovative model which utilises topics as extra-sentential contexts for the calculation of the information content of the words, and (B) a supervised method employing a recurrent neural network (RNN). As baselines, we employed TextRank and the TF-IDF ranking function. The topic model (A)(iii) outperformed clearly all remaining models, even TextRank and TF-IDF. In contrast, RNN performed poorly. We take the results as first evidence, that (i) information content can be employed for keyword extraction tasks and has thus a clear correspondence to semantics of natural language's, and (ii) that - as a cognitive principle - the information content of words is determined from extra-sentential contexts, that is to say, from the discourse of words.

1 INTRODUCTION

How can the content of a document be captured in a few words? Keyword extraction can be an answer to this research question and is, due to the rapidly increasing quantity and availability of digital texts and since the pioneering work of (Witten et al., 2005), a vital field of research on applications such as automatic summarisation (Pal et al., 2013), text categorisation (Özgür et al., 2005), information retrieval (Marujo et al., 2013; Yang and Nyberg, 2015) and question answering (Liu and Nyberg, 2013). Methodically, they are two general lines of research: supervised and unsupervised approaches. In this study, we propose an innovative unsupervised approach to keyword extraction that utilises Shannon information theory (Shannon, 1948).

In previous studies, it came to light that Shannon Information can provide an explanatory source of natural language phenomena such as sentence comprehension and verbal morphology (Hale, 2001; Jaeger and Levy, 2007; Levy, 2008; Jaeger, 2010). Based on these observations, we hypothesise that above-average informative words capture the meaning of a text and can be taken as keyword-candidates.

In well-known linguistic approaches to information structure of sentences, new information is considered the relevant one. New information is part of the common opposite pairs *given - new* or *topic - comment*, respectively. Within sentences, new information - after the setting of something given - can be said to form the message, i.e. the new information that the human language processor is awaiting. Within *alternative semantics* (Rooth, 1985; Rooth, 1992), the focus-position is filled by that new information: "Focus indicates the presence of alternatives that are relevant for the interpretation of linguistic expressions" (Krifka, 2008). That is to say, the more alternatives there are, the higher the relevance of the actually occurring word is, and this relationship, in fact, meets Shannon definition of information (Shannon and Weaver, 1948). High relevant words are accordingly be at the same time high informative.

We will compare our information theory-based, unsupervised approach against a supervised deep-learning approach that employs a recurrent neural network (RNN). We focus on the German language. The research question is whether keyword extraction using our simple information theory-based approach is able to compete with a state of the art deep-learning

[a] https://orcid.org/0000-0002-5715-4508
[b] https://orcid.org/0000-0003-0577-7831
[c] https://orcid.org/0000-0001-7460-4139
[d] https://orcid.org/0000-0001-6136-3970

technique.

In contrast, in the deep learning approach, there is no explicit hypothesis w.r.t. the semantics of words. In order to avoid the RNN to perform keyword generation instead of keyword extraction, the algorithm is solely trained on document-keyword-pairs.

Two baseline methods are used in order to evaluate the quality of the aforementioned approaches: (1) TF-IDF-measure (Salton and Buckley, 1988; Witten et al., 2005; Özgür et al., 2005) and (2) TextRank (Mihalcea and Tarau, 2004). We chose the latter since it is a highly influential graph-based-ranking-approach on keyword extraction.

We utilise Shannon's definition of information (given in (1)): Shannon information content (SI) in bits is the negative logarithm of the probability of a sign w in its context(Shannon, 1948).

$$SI(w_i) = -\log_2 (P(w_i|context)) \quad (1)$$

The amount of surprisal that a word causes is equivalent to its *SI* and proportional to the difficulty when that word is mentally processed (Hale, 2001; Levy, 2008): a sign is more informative if it is more surprising, i.e. if its probability in a given context is smaller. The definition of contexts of target words is a *conditio sine qua non* in the determination of their *SI*. Contexts can be n-grams of terminal symbols, n-grams of part-of-speech tags (Horch and Reich, 2016), but also the syntactic context(Celano et al., 2018; Richter et al., 2019b; Richter et al., 2019a). Furthermore, contexts can be limited to the sentence with the target word, but can also be extra–sentential i.e. they can also include preceding and subsequent sentences and even complete documents the target words occur in. Unlike n-gram models, probabilistic parser models (Hale, 2001) calculate the information content of a word by determining the change in probability of a parse tree when that word is added to the sentence. Probabilistic parser models are based on statistical frequencies in corpora and are formed from phrase structure rules or dependency rules to which probabilities are assigned (Hale, 2001; Levy, 2008). The information theory-based approach to keyword extraction put forward in this study employs three models with different context definitions:

1. a bigram model that has yielded promising results in a previous pilot study (Rietdorf et al., 2019), henceforth referred to as 'bigram model',

2. a probabilistic parser model based on phrase structures, for which (Hale, 2001) claims psycholinguistic plausibility, henceforth referred to as 'parser model',

3. an innovative extra-sentential topic model based on Latent Dirichlet Allocation (LDA) (Blei et al., 2003) that defines as contexts the topics in documents that contain the target words, henceforth referred to as 'topic model'.

The idea of topic contexts is to determine how informative / surprising a word w is, given the topics within all its discourses, i.e. the documents, in which w occurs. Let's assume that w is 'polar bear' and occurs in two documents of a corpus within the context of three topics in total: document d_1 has two topics, that we might interpret as something like 'climate change' and 'health', and document d_2 has only one topic, which we might interpret as something like as 'Arctic' or 'Antarctic'. An interpretation of which topics are involved is necessary since a topic model like LDA just outputs abstract topics, i.e. the strength of assignment of words to non-labelled topics.

The measure we apply to calculate average *SI* of 'polar bear' is *Average Information Content* (Piantadosi et al., 2011), see formula 3 below. In the above example, $\frac{2}{3}$ of the w's topics occur in d_1, $\frac{1}{3}$ in d_2. Applying (1), $SI(w)$ in d_1 is $-\log_2 \frac{2}{3}$ and in d_2 it is $-\log_2 \frac{1}{3}$ and $\overline{SI}$ is the mean of $SI(w)$ in d_1 and d_2.

A topic model thus aims to extract a concrete keyword, in other words a concrete topic, extracted from a context of abstract topics.

2 RELATED WORK

To the best of our knowledge, information theory has rarely been utilised for keyword extraction so far. Mutual information has been used by (Kaimal et al., 2012) and by (Huo and Liu, 2014) for abstractive summarisations. However, the calculation of mutual information does not take extended and extra-sentential contexts of target words into account, as, in contrast, our approach does (see above). In general, pioneering work in supervised approaches to keyphrase extractions comes from (Witten et al., 2005) who introduced the KEA-algorithm that is based on the features 'TF-IDF' and 'First Occurrence' of key phrases, and employs a Bayesian-classifier. Nowadays, graph-based approaches such as TextRank (TR) (Mihalcea and Tarau, 2004) are state of the art. TR is based on co-occurrences of words, that is, as directed graph, on the amount of incoming and outgoing links to neighbors to both sides of the target words. A highly effective graph-based approach is introduced by (Tixier et al., 2016) who utilise k–trusses (Cohen, 2008) within k–degenerate graphs for keyword extraction. The authors propose that TR is not optimal for keyword extraction since it fails to detect 'dense substructures' or, in other words, 'influential spreaders' / 'influential' nodes (Tixier et al., 2016)

within the graph. The idea is to decompose a graph of words with maximum density to the core (Hulth, 2003).

3 DATASET

We collected 100,673 texts in German language from heise.de and split them into a training set containing 90,000 texts and a validation set with 10,673 texts. For each text we have the headline and the text body, for 56,444 texts we also have the lead text of which 50,000 are in the training set. The number of characters of each text varies between 250 and 5,000 characters. The keywords were extracted from the associated meta-tag. There are 50,590 keywords in total. For this paper, we focused on the keywords that can be found in the headline, the lead and the text, resulting in 38,633 keywords. The corpus contains a total of $1,340,512$ word types when splitting on blanks and $622,366$ when filtering using the regex `[\w-]+` with unicode support.

The frequency of the keywords varies extremely. The three most common keywords are Apple, Google and Microsoft with a frequency of 7,202, 5,361 and 4,464 respectably. On the other hand 24,245 keywords only occur once. 25,582 keywords are single words and the longest keyword is 'Bundesanstalt für den Digitalfunk der Behörden und Organisationen mit Sicherheitsaufgaben'.

4 METHOD

4.1 Baseline

The first baseline approach we employed is TextRank (Mihalcea and Tarau, 2004). For keyword extractions, the TextRank-algorithm builds a (directed) graph with words (or even sentences) for nodes within a text of a paragraph. The weight of a word is determined within a sliding context window and results essentially from the number of outgoing links of the words directly preceding the target word.

The second baseline we utilised, was the TF-IDF-ranking function of words (Sparck Jones, 1972). This measure is the product of the frequency of a term within a specific document and the log of the quotient of the total number of documents in our universe and the number of documents that contain that term.

4.2 Information Theory based Methods

(I) Bigram Model. We determine the probability of a word on the basis of the probability that it occurs in the context of the preceding word. We have chosen a bigram model, because the chosen corpus contains many technical words, which occur only rarely. This leads to the undesired fact that when calculating with 3-grams (or higher) many of the calculated probabilities are 1 and consequently the information content of these words is 0. Thus we calculate the information content of a word with:

$$I(w_i) = -\log_2 (P(w_i|w_{i-1})) \tag{2}$$

For the calculation, all bigrams from the headings, leads, and texts of the corpus were extracted and preprocessed. Then their frequency within the corpus was counted. During the preprocessing, the words were lowercased in order not to distinguish between upper and lower case forms of the same words. Furthermore, punctuation and special characters were removed. A calculation of the information content of these signs would not be meaningful, since they are not suitable as candidates for keywords. Digits were also replaced by a special character ('$') in order not to distinguish between individual numbers (e.g. 1234 and 1243) in the information calculation, which would lead to a disproportionately high information content due to their rarity. All keyword occurrences where the keyword consists of more than a word were combined into a single token.

The five most informative words of each text were chosen as keywords.

(II) Parser Model. Another method we used is based on (Hale, 2001). Hale points out that n-gram models do not have any notion of hierarchical syntactic structure. Stolcke's probabilistic Earley parser which Hale examines in his paper, on the other hand, covers the shortcomings of n-gram models by taking into consideration hierarchical structures in form of parse trees. The idea is that the model measures the change of the parse tree of a sentence when another word is added. Specifically, that means that – given a sentence $s = w_1 \dots w_n$ with w_i being words – a parse tree is made for every subsentence $w_1 \dots w_m$ with $m \leq n$. Then, a probability (the *prefix probability*) is assigned to every parse tree. Finally, to compute the information content of a word w_i in s, the prefix probability of $w_1 \dots w_i$ is divided by the one of $w_1 \dots w_{i-1}$. A prefix probability is computed from probabilities of the individual rules of the associated parse tree, which come from the frequency of these rules in the training corpus. We used the de_core_news_md language model from spaCy to create the parse trees. Then, we filtered out punctuation marks, some irrelevant words using a

stoplist, and words which consist mostly of non-letter symbols, like dates or IP addresses. From the remaining words, we extracted the keywords of a given text by choosing the words with the highest information which together make 3% of the total information in the text.

(III) Topic Model. The model defines the context as a topic and calculates the average *SI* for each word depending on the contexts / topics in which it occurs within the discourse of the complete set of texts. Three main steps are taken, starting with preprocessing to clean and prepare the dataset for the next step which is LDA and finally calculating the *SI* for each word and extracting the word with the highest *SI* in each document as a predicted keyword. The preprocessing consists of removing non-alphabetical tokens, stopwords, verbs, adjectives, and prepositions. Then we have carried out topic modelling using the LDA algorithm with different numbers of topics. The accuracy of the model depends on the number of topics / contexts we use. We experimented with different number of topics and we got the best results when we used 500 topics. Then we calculated the $\overline{SI}$ for every word using the formula 3 where n is the number of contexts the word w occurs in and t is a topic:

$$\overline{SI}(w) = -\frac{1}{n}\sum_{i=1}^{n}\log_2(w|t_i) \tag{3}$$

Then for each token in the document a score is calculated by multiplying the $\overline{SI}$ with the word frequency in the document (see 4), where $c_d(w)$ is the frequency of word w in document d.

$$score(w_d) = \overline{SI}(w) \cdot c_d(w) \tag{4}$$

The words with the highest scores are taken as keywords.

4.3 Neural Network

To be in line with the other methods, the neural network also tries to follow the extractive approach. Similar to the approach of (Zhang et al., 2016) the neural network predicts if a word in the input sequence is a keyword or not. Instead of working on word level we chose the characterwise approach. Hence the neural network is fairly small because the amount of word types is in German is significantly larger.

The network architecture is straightforward. The network has three inputs and outputs, one for each part of a text e.g. headline, lead and text. First comes an embedding layer and then a bidirectional GRU, these two are shared over all three inputs. The output layers are dense layers where the number of units corresponds to the maximum length of each part, e.g. 141, 438 and 5,001.

For the training all characters that have an occurrence of less than 80 in the whole dataset where treated as the same character. The network was trained for three epochs.

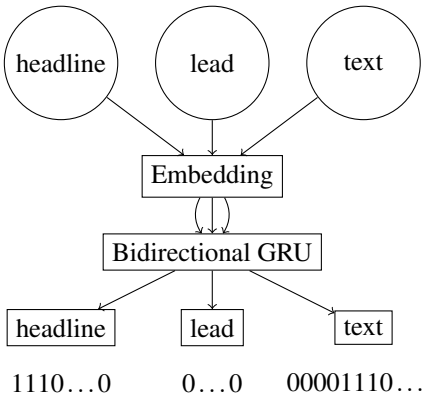

Figure 1: Schematic RNN network architecture.

4.4 Evaluation Method

As evaluation measures we used 1. Precision (Prec), Recall (Rec) and F1, and 2. accuracy. We determined the latter as follows: accuracy 1 (A1) is the percentage of the model generated keyword sets for which there is at least an intersection of one word with the respective keyword set from the dataset. For A2 and A3 we require at least two and three intersections, respectively.

5 RESULTS

The performances of the respective models are given in table (1) and table (2). The topic model clearly outperforms all remaining models in all measures that we applied. The two baseline models TextRank and TF-IDF perform considerably better than the bigram- and the parser model. The results of the RNN are of poor quality. It is striking that all models yield low precision, recall, and F1-scores.

6 CONCLUSION AND DISCUSSION

In this study we compared two methods in general, that is, a supervised method, i.e. a recurrent network, and an unsupervised method based on information theory. In order to estimate the quality of the

Table 1: Precision (Prec), recall (Rec), F1 of the employed methods.

	Prec	Rec	F1
TextRank	6.99%	6.78%	7.35%
TF-IDF	3.3%	3.2%	3.2%
RNN	0.92%	0.92%	0.92%
Bigram model	3%	17%	5.1%
Parser model	3.4%	3.2%	3.3%
Topic model (50)	17.39%	17.42%	17.39%
Topic model (100)	19.35%	19.30%	19.71%
Topic model (300)	20.65%	20.59%	20.62%
Topic model (500)	21.48%	21.42%	21.45%

Table 2: The three accuracy–values (a1 – a3) of the employed methods.

	a1	a2	a3
TextRank	22.15%	1.97%	0.12%
TF-IDF	18.54%	2.54%	0.26%
RNN	1.10%	0.10%	0.10%
Bigram model	11%	3.2%	0.7%
Parser model	8.23%	0.57%	0.04%
Topic model (50)	54.5%	16.2%	2.88%
Topic model (100)	57.3%	18.6%	3.66%
Topic model (300)	59.25%	19.85%	4.15%
Topic model (500)	60.89%	21.15%	4.58%

results, we utilised as baselines TextRank and the TF-IDF ranking function.
An n-gram model, a probabilistic parser model (Hale, 2001) and an innovative model based on the topics in the respective document, i.e. an extra-sentential model, were used as methods based on information theory.

It turned out that the topic model yielded by far the best results. The bigram and the probabilistic parser model, on the other hand, performed poorly, and the supervised RNN model was the taillight. Interestingly, our Topic model also outperformed the two baseline models, TextRank and TF-IDF. TextRank performed poorly in comparison, for example, to the results in (Mihalcea and Tarau, 2004).

These results have first of all a cognitive implication: Our study provides first evidence that when determining information of words it seems that extra-sentential contexts are superior, in this case the complete document. In particular the documents' topics contexts are exploited. The outcome indicates that topic-contexts yield a promising approximation to human keyword extraction. This raises the question whether we can consider it a plausible cognitive model.

In our study, two additional information theory-based models, i.e. probabilistic parser model inspired by (Hale, 2001) and an n-gram model did not provide convincing evidence of cognitive plausibility: the bigram model does not seem to be able to model the information content of words because the contexts is simply too small.

The probabilistic parser model, though it captures syntactic intricacies very well, does not seem to be fit for tasks involving semantics alone in the syntactically homogeneous discourse which is technology news. Apart from that, for long sentences with more than 80 tokens the probabilities of the prefix trees were rounded down to 0 which rendered them unusable.

The poor results of the two baseline models, on the other hand, are more difficult to explain. One possible reason is that German has different features compared to English, in which they have achieve good results. German is morphologically more complex than the English language and has a considerably larger number of possible tokens. Which might have influence on the information distribution in sentences. Secondly the possibility of combining words with hyphens is used much more frequently in German than in English, for example in combinations such as *Ebay–Konzern* and *PDF–files*.

The bad results could also be due to a corpus bias: the long keywords are characteristic for texts on technologic topics.

The poor performance of the neural network is partly due to the choice that the network works characterwise. In some cases the neural network predicted that for example ' FDP ' is a keyword. Since the whitespaces are not part of the keyword the network predicted a wrong keyword. In contrast if the network would work on tokens this would not happend, but the network would be significantly larger. Additionally there was very little time to test various hyper parameters.

Whether these or other features influence the performance of the models, is a topic of future research.

ACKNOWLEDGEMENTS

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), project number: 357550571.

The training of the neural network was done on the High Performance Computing (HPC) Cluster of the Zentrum für Informationsdienste und Hochleistungsrechnen (ZIH) of the Technische Universität Dresden.

REFERENCES

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Celano, G. G., Richter, M., Voll, R., and Heyer, G. (2018). Aspect coding asymmetries of verbs: the case of russian.

Cohen, J. (2008). Trusses: Cohesive subgraphs for social network analysis. *National security agency technical report*, 16:3–1.

Hale, J. (2001). A probabilistic earley parser as a psycholinguistic model. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, pages 1–8. Association for Computational Linguistics.

Horch, E. and Reich, I. (2016). On "article omission" in german and the "uniform information density hypothesis". *Bochumer Linguistische Arbeitsberichte*, page 125.

Hulth, A. (2003). Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 216–223. Association for Computational Linguistics.

Huo, H. and Liu, X. H. (2014). Automatic summarization based on mutual information. In *Applied Mechanics and Materials*, volume 513, pages 1994–1997. Trans Tech Publ.

Jaeger, T. F. (2010). Redundancy and reduction: Speakers manage syntactic information density. *Cognitive psychology*, 61(1):23–62.

Jaeger, T. F. and Levy, R. P. (2007). Speakers optimize information density through syntactic reduction. In *Advances in neural information processing systems*, pages 849–856.

Kaimal, R. et al. (2012). Document summarization using positive pointwise mutual information. *arXiv preprint arXiv:1205.1638*.

Krifka, M. (2008). Basic notions of information structure. *Acta Linguistica Hungarica*, 55(3-4):243–276.

Levy, R. (2008). Expectation-based syntactic comprehension. *Cognition*, 106(3):1126–1177.

Liu, R. and Nyberg, E. (2013). A phased ranking model for question answering. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 79–88. ACM.

Marujo, L., Bugalho, M., Neto, J. P. d. S., Gershman, A., and Carbonell, J. (2013). Hourly traffic prediction of news stories. *arXiv preprint arXiv:1306.4608*.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Özgür, A., Özgür, L., and Güngör, T. (2005). Text categorization with class-based and corpus-based keyword selection. In *International Symposium on Computer and Information Sciences*, pages 606–615. Springer.

Pal, A. R., Maiti, P. K., and Saha, D. (2013). An approach to automatic text summarization using simplified lesk algorithm and wordnet. *International Journal of Control Theory and Computer Modeling (IJCTCM)*, 3(4/5):15–23.

Piantadosi, S. T., Tily, H., and Gibson, E. (2011). Word lengths are optimized for efficient communication. *Proceedings of the National Academy of Sciences*, 108(9):3526–3529.

Richter, M., Kyogoku, Y., and Kölbl, M. (2019a). Estimation of average information content: Comparison of impact of contexts. In *Proceedings of SAI Intelligent Systems Conference*, pages 1251–1257. Springer.

Richter, M., Kyogoku, Y., and Kölbl, M. (2019b). Interaction of information content and frequency as predictors of verbs' lengths. In *International Conference on Business Information Systems*, pages 271–282. Springer.

Rietdorf, C., Kölbl, M., Kyogoku, Y., and Richter, M. (2019). Summarisation by information maps. a pilot study.

Rooth, M. (1985). Association with focus.

Rooth, M. (1992). A theory of focus interpretation. *Natural language semantics*, 1(1):75–116.

Salton, G. and Buckley, C. (1988). Term weighting approaches in automatic text retrieval, information processing and management, vol. 24.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423.

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Tixier, A., Malliaros, F., and Vazirgiannis, M. (2016). A graph degeneracy-based approach to keyword extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1860–1870.

Witten, I. H., Paynter, G. W., Frank, E., Gutwin, C., and Nevill-Manning, C. G. (2005). Kea: Practical automated keyphrase extraction. In *Design and Usability of Digital Libraries: Case Studies in the Asia Pacific*, pages 129–152. IGI Global.

Yang, Z. and Nyberg, E. (2015). Leveraging procedural knowledge for task-oriented search. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 513–522. ACM.

Zhang, Q., Wang, Y., Gong, Y., and Huang, X. (2016). Keyphrase extraction using deep recurrent neural networks on twitter. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 836–845.