Definition Extraction from Mathematical Texts on Graph Theory in German and English

Theresa Kruse
University of Hildesheim
Universitätsplatz 1
31141 Hildesheim, Germany
kruset@uni-hildesheim.de

Fritz Kliche
University of Hildesheim
Universitätsplatz 1
31141 Hildesheim, Germany
kliche@uni-hildesheim.de

Abstract

We extract definitions from text books and scientific publications on mathematics in both, German and English, from the sub-domain of graph theory. Mathematical texts differ from other domains because sentences which appear as definitions from a linguistic perspective are not necessarily definitions in the mathematical sense. For the English texts we train a neural network on existing training data (Vanetik et al., 2020). For the German texts we semi-automatically generate training data using patterns for the extraction of definitions. We show that this is a feasible approach for the domain of mathematical texts which generally makes extensive use of formalized language patterns. We measure precision and recall on a random sample to evaluate our results. The F-Score is similar for both languages but precision and recall are closer to each other for the German data. Further comparisons are made with a term list automatically extracted from the data. We conclude that our approach can be used to extract candidate sentences for further postprocessing.

1 Introduction

In this paper, we combine two domains where definitions play a significant role: lexicography and mathematics. In lexicography, definitions provide dictionary users with information about a term. In mathematics, definitions are crucial to ensure a common understanding of the domain's concepts. We extract definitions from texts on graph theory to use them in a domain-specific dictionary. In Section 2, we give an overview of related work on types and forms of definitions and on extraction methods. In Section 3.1, we describe our data. Sections 3.2 and 3.3 present our method for the extraction of definitions from the English and the German data. Section 4.1 gives a qualitative analysis of the results and Section 4.2 a quantitative evaluation. We conclude in Section 5.

2 Background and Related Work

2.1 Definitions in Mathematical Texts

Mathematical texts consist of corollaries, lemmas, propositions, theorems, proofs and definitions (Solow, 1990). Usually, a numbering indicates the types (e.g. *Theorem 2.1*) but this is not necessarily the case for definitions. Some authors include them in the numbering and others simply give them in the text. Some authors highlight defined terms, e.g. by means of italics. Kruse and Heid (2020) analyze mathematical definitions for lexicographic purposes. They conclude that *analytical definitions* (or *logical definitions* as they are also called) in the Aristotelian scheme are mostly used. *Single-clause when-definitions* appear to define adjectives and verbs, contrary to usual practice in dictionaries for general language, as Dziemianko and Lew (2006) proposed. For definition extraction, however, it is more relevant to distinguish a definition from a non-definition rather than paying attention to the different types of definitions. Nevertheless, some sentences are definitions from a linguistic perspective but not from a mathematical perspective in content. This constitutes a special challenge for automatic definition extraction.

Different kinds of sentences may be regarded as definitions in mathematics: The first kind defines a term which is used in the rest of the text. Often, one finds similar definitions for the same terms in different works of a sub-domain. Definition 1 is an example of such a definition from the sub-domain of graph theory. It defines the term *semiregular* as a property of *bipartite graphs*. We want to find such definitions in our extraction experiments. They follow the Aristotelian scheme with a definiendum (the term defined) and the definiens (the part defining, cf. e.g. Meyer, 2001, p. 283).

The second kind of definitions follows the Aristotelian scheme on a syntactic level but requires anaphora resolution to be comprehensible as it refers to something mentioned earlier or later in the text. For this kind, the context is indispensable to understand its meaning. Definition 2 constitutes such an example. These definitions are not useful for our dictionary project because they do not contain any relevant semantic information.

The third kind of definitions defines a variable which also cannot be used for the dictionary. Even if some variables often refer to the same objects (e.g. G for *graph*) they are not considered as terminology in our dictionary project as they rather resemble named entities. For example, Definition 3 formally defines $A(G)$ but is only relevant in the context of the particular paragraph, i.e., a proof or a construction, and not for the conceptual sphere of the domain.

We thus aim to find definitions following the scheme of Definition 1. However, the structure of Definition 1 is ambiguous between definition and non-definition. As an example consider Definition 4. It appears to be a definition with *tree* as the definiendum and the subject as the definiens. This example is taken from Saha Ray (2013) where it actually is a theorem which requires a proof as *tree* has been defined before (cf. Definition 5). It is not obvious that a graph meets the criteria from Definition 5 if it already has the properties from Definition 4. We do not expect our system to differentiate between these two kinds of definitions because this would require an analysis of the context beyond sentence level.

Which aspects are used in a definition and which aspects are left to be proven depends on the author's preferences for introducing a concept. The decision seems arbitrary at first sight but depends on the author's intended target group of the text (Rey, 1995; Solow, 1990; van Dormolen and Arcavi, 2000). Some general aspects can be considered for the decision because a mathematical definition should meet certain criteria (van Dormolen and Zaslavsky, 2003): *Hierarchy, existence, equivalence* and *axiomatization* are necessary, whereas *minimality, elegance* and *degeneration* are common but not required.

A *hierarchy* between the defined terms is inherent to the Aristotelian scheme. Further, a definition is only meaningful if the term defined does actually *exist*. *Equivalence* refers to the above-mentioned aspect that different definitions may exist for the same concept. It has to be shown that they are actually equivalent. The criterion of *axiomatization* is related to *hierarchy*: It is possible to define more and more general hypernyms. In order to stop this chain at one point axioms are needed, usually related to set theory.

The following criteria are not mandatory: *minimality* requires that only necessary properties are mentioned in a definition without redundancies. *Elegance* is hardly an objective property but can be taken into consideration when deciding which of several possible definitions is to be taken and which is left to be proven. "*Degenerations* are instances of a concept which are not expected to be included when defining the concept. They are a logical consequence from the definition. One might not want the occurrence of such instances and therefore change the definition in order to exclude them. Describing an instance as a degeneration is, of course, highly subjective and there is no objective criterion for this decision" (van Dormolen and Zaslavsky, 2003). These criteria combined with the individual preferences and ideas of concepts sum up to the final definitions which a mathematician writes.

Mathematical definitions are usually unambiguous within a certain sub-domain. Nevertheless, homonymy may occur between different sub-domains. For example, the German *Körper* is translated into English as *solid figure* in geometry but as *field* in algebra. Another example is the adjective *complete* used as an attribute to *metric spaces* or *graphs*. The definitions differ considerably in both cases, although the same mental concept underlies both. As we work only with one mathematical sub-domain we can neglect homonymy.

Examples of definitions

1. We call a bipartite graph semiregular if it has a proper 2-colouring such that all vertices with the same colour have the same valency.

2. We call the above procedure branching-search.

3. Let $A(G)$ be an incidence matrix of a connected graph G with n vertices.

4. A connected graph with n vertices and $n-1$ edges is a tree.

5. A tree is a connected acyclic graph.

6. The floor function $\lfloor x \rfloor$, also called the greatest integer function or integer value, gives the largest integer less than or equal to x.

7. Similarly, define the points A_c, B_c, B_a, C_a, C_b so that the points A_c and B_c lie on the extended segment AB, the points B_a and C_a lie on the extended segment BC, and the point C_b lies on the extended segment CA, and we have $AA_c = a$, $BB_c = b$, $BB_a = b$, $CC_a = c$ and $CC_b = c$.

2.2 Definition extraction

Definition extraction originally started with pattern-based approaches. The patterns were then combined with a grammar analysis for e.g. apposition and anaphora resolution or syntactic features. These methods have been applied to several languages like English (Klavans and Muresan, 2001), German (Storrer and Wellinghoff, 2006), Spanish (Alarcón et al., 2009) and Dutch (Fahmi and Bouma, 2006). Examples for such patterns in English texts are *is called*, *is the term used to describe*, *is defined as*, *is the term for*. In German, the following patterns can be indicative for definitions: *bedeuten*, *begreifen als*, *bekannt als*, *benennen*, *beschreiben*, *bestehen aus*, *bezeichnen als*, *charakterisieren als*, *definieren als*, *gebrauchen*, *heißen*, *nennen*, *sein*, *spezifizieren als*, *sprechen von*, *Terminus einführen*, *verstehen unter*, *verwenden als*, *vorstellen als*[1].

Pattern-based approaches have been used in a wide range of applications. They were among others used by Meyer et al. (1999), Meyer (2001), or Pearson (1998) and are still applied today (Christensen, 2019). Barbaresi et al. (2018) extract "definitory contexts" for words from a broad range of domains (e.g. *Auseinandersetzungsbilanz* or *Palletheizung*) in the context of lexicography using patterns such as *a* X_1 *is a* X_2. In line with this approach, definitions in mathematical texts can be regarded as knowledge-rich contexts which can be used in pattern-based approaches for information extraction. (Meyer, 2001; Meyer et al., 1999). Cramer (2011, 183 ff.) analyzes linguistic features of definitions. Schumann (2014) describes (corpus-)linguistic analyses for the detection of text passages containing description (thus not explicitly definitions) of terminologically relevant concepts.

Other approaches combine pattern-based extraction and machine learning (e.g., Westerhout, 2009). Boella and Di Caro (2013) combine syntactic dependencies with a Support Vector Machine classifier without using patterns. Fišer et al. (2010) combine morphosyntactic patterns, automatic terminology recognition and semantic tagging with WordNet senses for their work on Slovene Wikipedia texts.

Today, learning algorithms and neural networks are frequently used for definition extraction. Borg et al. (2010) use genetic programming and genetic algorithms to train their system on grammatical rules. Navigli and Velardi (2010) introduce Word-Class Lattices, an approach based on word lattices generalizing over lexico-syntactic definitional patterns which outperforms traditional extraction methods. Reiplinger et al. (2012), however, compare two methods, one based on bootstrapping lexico-syntactic patterns and the other based on deep analysis, and do not find major differences in the performances. Espinosa-Anke et al. (2015) use a weakly supervised bootstrapping approach and Espinosa-Anke and Schockaert (2018) combine Convolutional and Recurrent Neural Networks for definition extraction.

Del Gaudio and Branco (2009) suggest that definition extraction is language and domain independent. But Vanetik et al. (2020) show that this does not hold for definition extraction from mathematical texts. They work on a corpus crawled from *Wolfram MathWorld*[2] and indicate whether a certain sentence is a definition.[3] They conclude "that mathematical definitions require special treatment, and that using cross-domain learning for detection of mathematical definitions is inefficient".

3 Experiments

3.1 Data preprocessing

Our work is based on two comparable corpora, one in German, one in English, with texts from the mathematical sub-domain of graph theory. The German corpus contains about $700,000$ tokens with about $30,000$ types and consists of lecture notes and (parts of) nine text books. Parts of books, as opposed to the entire book, were used when only some chapters cover graph theory. The English

[1] Engl. *mean, understand as, known as, designate, describe, consist of, refer to as, characterize as, define as, use, be called, state, be, specify as, speak of, introduce a term, understand by, use as, conceive as*

[2] https://mathworld.wolfram.com

[3] The data of Vanetik et al. (2020) is publicly available on GitHub https://github.com/uplink007/FinalProject/tree/master/data/wolfram

corpus consists of eight text books and 26 scientific papers, totaling about one million tokens with about 30,000 types.

Our goal was to create corpora of a similar size. The exact number of tokens depends on how formulas are counted. We chose material from text books and literature students at our institution work with, as students are the target group of our dictionary. The choice of texts was based on a survey we carried out with the students (Kruse and Giacomini, 2019). Although many students indicated that they use Wikipedia for their studies we decided against including it into our corpus because we have less control on its quality from an academic perspective. Due to copyright restrictions we cannot make our corpus publicly available but in can be reproduced as we used published material.

Our source files are machine-readable PDF documents, scans and plain texts. As the data is not homogeneous, we had to use different workflows to integrate them into the corpus depending on the source file. We used inftyreader[4] and Tesseract (Smith, 2007) to convert PDF documents into plain text. The mathematical formulas produced some obstacles, e.g., Tesseract had difficulties to convert fractions into plain text as it works line by line. Inftyreader is specialized in processing mathematical texts and converts formulas according to the W3C standard MathML[5] but has difficulties with low quality scans. In the latter cases we used Tesseract which ignored the formulas. Thus, some errors remain in the texts due to errors in the optical character recognition (OCR). Afterwards, we did some semi-manual post-processing to eliminate the most common errors but could not cover all of them. Thus, some errors remain as can be seen in Examples 8 and 9.

We remove Latex commands for typesetting and document layout, while commands for mathematical formulas (e.g. `\sum`) are kept to preserve parts of the formulas in the input for the classifier. We split the data into sentences using the tokenizer described by Schmid (2000). Some issues with the automatic split into sentences remain, e.g., the exclamation mark is used for calculating factorials, or sentences with the following structure appear: *We can say that G is bipartite (why?) and continue the following way...*, where *(why?)* should motivate the reader to realize the truth of the given statement. As it would cost too much effort to go through these cases manually we leave them unchanged but they should be kept in mind when discussing quantitative results such as the number of sentences in the corpus because a different tokenizer might yield different results.

3.2 Definition Extraction from the English Corpus

We use the training data compiled by Vanetik et al. (2020) for the extraction of definitions in the English corpus. The training data consists of 1,793 sentences of which 811 are definitions. We count the sentences with domain-specific definition patterns in the training data using the following pattern indicators: *abbreviate, termed, determine, definition, refer, name, the term, associate, consist, said to be, then .* is, denote, known as, given by, is a(n), define, call, is the*. 72.87 % of the definition sentences and 28.21 % of the non-definition sentences contain at least one of the patterns. This legitimates our workflow to semi-automatically create the German training data by extracting sentences containing definition patterns.

We further analyze the training data and find definitions in which none of the patterns appears. They often contain the verbs *is*/*are*, *has*/*have* not followed by an article and thus deviate from the standard pattern. We exclude these verbs in our set of defining verbs to avoid too many false positives.

As mentioned above, some non-definitions also contain the patterns. One of these cases is the verb *call* which in the non-definitions is frequently combined with *also*, as in Definition 6 where it is followed by a synonym but not an actual definition. In our lexicographic application, synonyms are dealt with separately from definitions. Similar reasons hold for the other definition patterns in the non-definitions. We give an example for the indicator *define* in Definition 7 which constitutes a typical example of defining a variable, as described in Section 2 (cf. Definition 3).

After the pre-processing as described in Section 3.1, our corpus contains 56,978 English sentences to be classified. We use the SimpleTransformers implementation[6] of BERT (Devlin et al., 2019)[7] with one epoch. 11,936 (20.95 %) of the sentences were classified as definitions and 45,042 sentences

[4] https://www.inftyreader.org
[5] https://www.w3.org/TR/MathML3/

[6] https://github.com/ThilinaRajapakse/simpletransformers
[7] https://huggingface.co/bert-base-cased

(79.05 %) as non-definitions.

Again, we count the sentences containing at least one definition pattern: 51.42 % of the sentences classified as definitions contain a pattern but only 15.98 % of those classified as non-definitions. Like in the training data, more sentences classified as definitions contain one of the patterns. Yet, this holds for only half of the sentences unlike the 72.87 % in the training data. The amount of sentences classified as non-definitions containing a pattern is significantly lower which might be also a consequence of noise in the data.

We measure precision and recall on an exemplary random sample.[8] We manually collect 100 definitions and 100 non-definitions from our data set. To that end, we randomly sequence the sentences in the corpus and find definition sentences with help of patterns. For the non-definitions we randomly extract 200 sentences from the corpus and manually annotate if they are definitions. We take the first 100 of them for the evaluation. Thus, we have a random sample of 100 definitions and 100 non-definitions.

We measure precision and recall for the labels these 200 sentences were assigned with by the BERT classifier. The results for the definition sentences are given in Table 1. If we evaluate, in turn, the classification of non-definitions we get a precision of 0.8857 and a recall of 0.62 resulting in an F-Score of 0.7229. The higher precision of the non-definitions can probably be explained with the much higher number of non-definitions in the data compared to the number of definitions. Likewise, the high recall for the definitions can be explained by the fact that we calculate the values on a balanced random sample. We would get more realistic results if we would select 200 sentences completely random for this evaluation but in that case we run the risk of having almost no definitions in the sample which would not give reliable results.

3.3 Definition Extraction from the German Corpus

For the German corpus we create our own training data. To that end, we collect sentences containing at least one form for the following lemmas: *bestehen, bezeichnen, definieren, heißen, nennen, sagen, sprechen, verstehen.*[9] We randomly extract a maximum of 100 sentences for each indicator verb and manually annotate them as definitions or non-definitions following the criteria detailed in Section 2. Additionally, we manually search the corpus for examples of definitions which do not contain an indicator verb, e.g., because they contain the verb *sein* (engl. *be*). Again, we did not include all sentences containing *sein* to avoid false positives. Further, non-definitions without any indicator verbs are added to the data set. In sum, we collect 799 sentences of which 256 are definitions.

We use the pre-trained model `bert-based-german-cased`[10] from the Hugging Face library and one epoch of training. All results are summarized in Table 1. 90.54 % of the sentences are labeled as non-definitions and 9.46 % as definitions. 47.79 % of the sentences labeled as definitions contain at least one of the patterns whereas this is the case for only 4.75 % of the sentences labeled as non-definitions which matches the expectation as this percentage is higher for definitions. For measuring precision, recall and F-Scores we evaluate again a random sample of 100 sentences for each category. We yield a similar F-Score as for the English data. But precision and recall for German are closer to each other, i.e., the precision is slightly higher and the recall slightly lower. This might be explained by the fact that the percentage of sentences labeled as definitions is lower in the German data set. However, this comparison is only valid if we expect the same percentage of definitions in both corpora.

4 Discussion

4.1 Qualitative Analysis

Both, the English and German results have lower values for precision but higher values for recall. Thus, the definitions are usually found but false positives need to be filtered. We take a closer look at the false negatives in our evaluation samples. The German sample contains nine and the English sample only 15 false negatives (cf. Examples 8 to 11). Example 8 repeats the distributive law which is not defined in this sentence. Example 9 states that two already defined terms describe the same concept. This is another example where definitions and theorems are not distinguishable. The same holds for Example 10. Four of the nine false negatives in

[8] We thank the anonymous reviewers for their useful comments on the evaluation and discussion sections.

[9] Engl. *consist, denote, define, call, name, (to) name, say, speak, understand*

[10] `https://huggingface.co/bert-base-german-cased`

		German	English
Number...	...of sentences	36,103	56,978
	...classified as definition	3,417 (9.46 %)	11,936 (20.95 %)
	...classified as non-definition	32,686 (90.54 %)	45,042 (79.05 %)
Patterns in sentences...	...classified as definitions	47.79 %	51.42 %
	...classified as non-definitions	4.75 %	15.98 %
Evaluation of random sample	precision	0.7522	0.7054
	recall	0.8500	0.9100
	F-Score	0.7981	0.7948

Table 1: Overview of extraction results

the random sample contain the phrase *we say that*. We searched for this phrase in the training data and found that no example containing this phrase is included. This might be because the data was extracted from *Wolfram MathWorld* and not from scientific publications or textbooks. This might hint at differences in the "language for definitions" in different resources.

Examples 12 to 14 are false positives. Example 12 contains tokens which could also occur in definitions (e.g. *ist eine Zahl*, Engl. *is a number*). Example 13 is a similar case (*nennt man*, Engl. *is called*). Example 14 is an example from the English evaluation sample. It contains the expression *is defined* which is also indicative for a definition. Furthermore, the English sample includes several false positives beginning with *If*. In the whole data set, 2,719 sentences contain this feature; 66.50 % of them are classified as a definition. This ratio may be a result from the training data which contains 52 sentences with an initial *If* which are labeled in 78.85 % of the cases as definitions.

Examples

8. Es gilt das Distributivgesetz: a-(b +c) = (a-b) + (a-c) für alle a, b, c eK.
   The distributive law holds: a-(b +c) = (a-b) + (a-c) for all a, b, c eK.

9. Damit beschreiben die Ausdrücke { Ecken)-3-panzyklisch und { Ecken)-panzyklisch den gleichen Sachverhält.
   Thus, the expressions { node)-3-pancyclic and { node)-pancyclic describe the same state of affairs.

10. Die einzigen 3-kritischen Graphen sind Kreise ungerader Länge.
    The only 3-critical graphs are circles of odd length.

11. We say that a graph G is reconstructible if every reconstruction of G ' is isomorphic to G, in other words, if G can be ' reconstructed up to isomorphism from its vertex-deleted subgraphs.

12. Jeder Buchstabe ist eine Zahl zwischen 1 und n.
    Each letter is a number between 1 and n.

13. In diesem speziellen Fall nennt man die Menge {x,y} auch das ungeordnete Paar von x und y.
    In this particular case, the set {x,y} is also called the unordered pair of x and y.

14. The matrix Miq is defined dually.

4.2 Quantitative Analysis

For a quantitative analysis we extract 1,000 keywords and 1,000 multiword terms from our data for each language using the corpus web tool *Sketch Engine* (Kilgarriff et al., 2014) which includes a function for keyword extraction. One rater evaluates in two rounds if these automatically found "keywords" are terminologically relevant. In the German list, 0.4705% of the keywords were relevant in this sense, and 0.5350% in the English list. These values are quite similar. Most of the false positives are variables and multiword expressions like *following graph*.

The chosen terms are manually divided into nine semantic categories:

- ACTIVITY: events which can be performed in graph theory, mostly verbs, e.g. *connect*

- ALGORITHM: domain-specific algorithms having a given name, e.g. *Dijkstra's algorithm*
- GENERAL: mathematical terminology which is not particularly attributed to the domain of graph theory, e.g. *disjoint*
- MAPPING: mappings in the mathematical sense, e.g. *edge contraction*
- PART: elements a graph is composed of, e.g. *edges*
- PERSON: mathematicians who worked in graph theory and related areas, e.g. *Dijkstra*
- PROBLEM: mathematical problems having a given name, e.g. *Traveling Salesman Problem*
- PROPERTY: descriptions of a graph, mostly adjectives, e.g. *regular*
- THEOREM: mathematical theorems having a given name, e.g. *Kirchhoff's matrix tree theorem*
- TYPE: names for special kinds of graphs, e.g. *Petersen graph*

We expect to find definitions for ACTIVITIES, GENERAL TERMS, MAPPINGS, PARTS, PROPERTIES and TYPES. ALGORITHMS, PERSONS, PROBLEMS and THEOREMS are usually not defined in mathematics. Thus, we analyze the terms in the sentences considered as definitions.

Table 2 shows the percentage of lemmas in the sentences classified as definitions. The value is higher for the English data which can be explained with the higher amount of definition sentences and the slightly lower precision indicated by the random sample. Thus, the probability to find a word in this set is generally higher. Figures 1 and 2 show which lemmas are found grouped by category. This matches our hypothesis that definitions mostly lack for the categories PERSON, PROBLEM, THEOREM and ALGORITHM. The results are much clearer for the German data which matches the results for precision and recall (cf. Table 1). We conclude that the definition extraction worked well for the majority of sentences which is reflected by the values for recall.

Still, some aspects affect the results, e.g., we did not exclude variants in our simple search. So, there is for example a definition for *1-Faktor-Satz* but not for *1-Faktorsatz*; and some multiword terms appear in the lemma list as a compound but are separated in the definition.

5 Conclusion and future work

Our approach yields higher values for recall but lower values for precision. We conclude that our semi-automatic approach can be used for finding candidates for mathematical definitions but they require a subsequent manual or automatic post-processing in order to distinguish definitions from sentences with a similar syntactic structure and vocabulary. An active learning approach in which parts of the results are evaluated in order to increase the training data iteratively could improve the approach.

We get different results for the English and the German data. We see several reasons for that: The German training data was semi-automatically generated using sentences from the sample on which the trained model was subsequently applied. Therefore, the same rules for annotating definitions were used for the generation of training data and for the evaluation of the results. For our English training data provided by Vanetik et al. (2020) we only had few indications of the annotation guidelines. Furthermore, the German training data contained only half as many sentences as the English data. In combination with the fact that the training data and evaluation data stem from the same corpus, there might be some over-specification to the data set. It might be interesting to train a network on this data and to apply the model on mathematical texts from different sub-domains.

About $20\,\%$ of the English sentences are classified as a definition, but only about $10\,\%$ of the German sentences. A reason for this difference may be the number of sources: The German corpus comprises of only ten texts while the English corpus contains 34 texts which are shorter. A reason for the different lengths are the text types as we used more text books for German and more scientific papers for English. The number of definitions in a mathematical text also depends on its type. In general, we would expect scientific papers to contain less definitions when compared to textbooks because they can pick up prior knowledge of their readers whereas textbooks are mostly targeted at learners with less prior domain knowledge. However, our results do not confirm this hypothesis as there are more sentences classified as definitions in

	German Data	English Data
number of definitions	3,417	11,936
number of lemmas	1,070	933
percentage of lemmas found in definitions	70.63%	90.47%

Table 2: Amount of lemmas in data

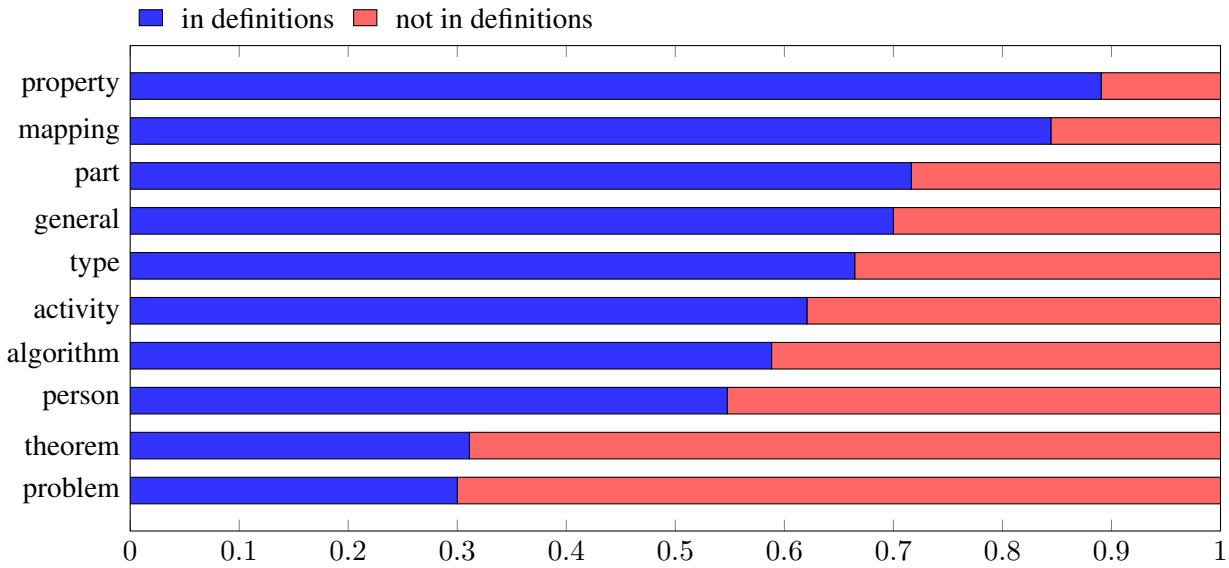

Figure 1: Distribution of German lemmas in definitions over categories

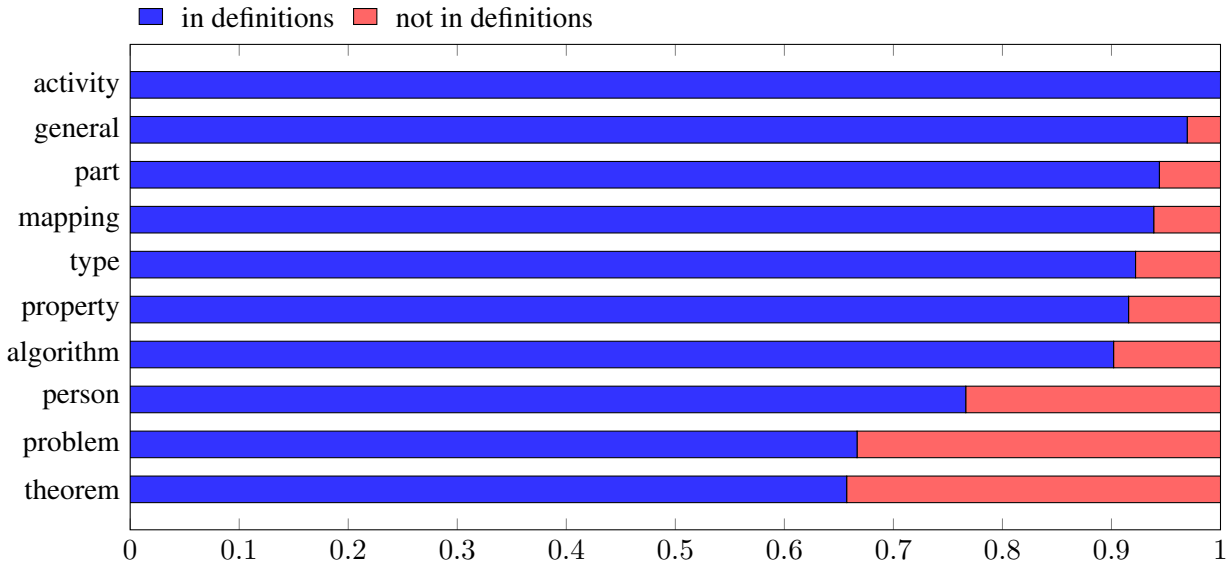

Figure 2: Distribution of English lemmas in definitions over categories

the English data than in the German. This might be related to the lower precision for the experiments on English texts. It would be interesting to investigate empirically if the percentage of definitions varies across different mathematical text types.

Furthermore, for our English corpus we had to rely more on OCR than for the German data. This may result in more mistakes which cause difficulties for the classifier. Interesting further research would be to analyze if the English extraction results differ when the training data is taken from the same corpus or from a corpus of the same sub-domain or type of resource. Maybe the results of Vanetik et al. (2020) can be interpreted in the more general way that the quality of definition extraction increases with the similarity between training data and evaluation data even for a highly formalized language like mathematics.

We can conclude that patterns are good indicators for mathematical definitions in German and English and can be used to generate training data. Nevertheless, automatic solutions are still needed for definition extraction in mathematics as some sentences are definitions from a linguistic perspective but not intended as such by their author.

References

Rodrigo Alarcón, Gerardo Sierra, and Carme Bach. 2009. Description and evaluation of a pattern based approach for definition extraction. In *Proceedings of the 1st Workshop on Definition Extraction*, pages 7–13, Borovets, Bulgaria. Association for Computational Linguistics.

Adrien Barbaresi, Lothar Lemnitzer, and Alexander Geyken. 2018. A database of German definitory contextsfrom selected web sources. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC'10)*, Miyazaki, Japan.

Guido Boella and Luigi Di Caro. 2013. Extracting definitions and hypernym relations relying on syntactic dependencies and support vector machines. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 532–537, Sofia, Bulgaria.

Claudia Borg, Mike Rosner, and Gordon J. Pace. 2010. Automatic grammar rule extraction and ranking for definitions. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta.

Lotte Weilgaard Christensen. 2019. Danish knowledge patterns and word sketches for semi-automatic extraction of terminological information. In Ingrid Simonnæs, Øivin Andersen, and Klaus Schubert, editors, *New challenges for Research on Language for Special Purposes. Selected Proceedings from the 21st LSP-conference 28-30 June 2017 Bergen, Norway*, pages 173–189. Frank & Timme, Berlin.

Irene Cramer. 2011. *Definitionen in Wörterbuch und Text: Zur manuellen Annotation, korpusgestützten Analyse und automatischen Extraktion definitorischer Textsegmente im Kontext der computergestützten Lexikographie*. Dissertation, Technische Universität Dortmund, Dortmund.

Rosa Del Gaudio and António Branco. 2009. Language independent system for definition extraction: First results using learning algorithms. In *Proceedings of the 1st Workshop on Definition Extraction*, pages 33–39, Borovets, Bulgaria. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Joop van Dormolen and Abraham Arcavi. 2000. What is a circle? *Mathematics in School*, 29(5):15–19.

Anna Dziemianko and Robert Lew. 2006. When you are explaining the meaning of a word: The effect of abstract noun definition format on syntactic class identification. In *Proceedings of the 12th EURALEX International Congress*, pages 857–863, Torino, Italy. Edizioni dell'Orso.

Luis Espinosa-Anke, Horacio Saggion, and Francesco Ronzano. 2015. Weakly supervised definition extraction. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 176–185, Hissar, Bulgaria. INCOMA Ltd. Shoumen.

Luis Espinosa-Anke and Steven Schockaert. 2018. Syntactically aware neural architectures for definition extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 378–385, New Orleans, Louisiana.

Ismail Fahmi and Gosse Bouma. 2006. Learning to identify definitions using syntactic features. In *Proceedings of the Workshop on Learning Structured Information in Natural Language Applications*, pages 64–71.

Darja Fišer, Senja Pollak, and Špela Vintar. 2010. Learning to mine definitions from slovene structured and unstructured knowledge-rich resources. In *Proceedings of the Seventh International Conference*

on Language Resources and Evaluation (LREC'10), Valletta, Malta.

Adam Kilgarriff, Vìt Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlý, and Vìt Suchomel. 2014. The sketch engine: ten years on. *Lexicography*, pages 7–36.

Judith L. Klavans and Smaranda Muresan. 2001. Evaluation of the definder system for fully automatic glossary construction. In *Proceedings AMIA Symposium*, pages 324–328.

Theresa Kruse and Laura Giacomini. 2019. Planning a domain-specific electronic dictionary for the mathematical field of graph theory: definitional patterns and term variation. In *Electronic lexicography in the 21st century. Proceedings of the eLex 2019 conference*, pages 676–693, Brno. Lexical Computing CZ, s.r.o.

Theresa Kruse and Ulrich Heid. 2020. Lemma selection and microstructure: Definitions and semantic relations of a domain-specific e-dictionary of the mathematical domain of graph theory. In *Euralex Proceedings*, volume 1, pages 227–233.

Ingrid Meyer. 2001. Extracting knowledge-rich contexts for terminography: A conceptual and methodological framework. In Didier Bourigault, Christian Jacquemin, and Marie-Claude L'Homme, editors, *Recent Advances in Computational Terminology*, volume 2 of *Natural Language Processing*, pages 279–302. John Benjamins, Amsterdam/Philadelphia.

Ingrid Meyer, Kristen Mackintosh, Caroline Barrière, and Tricia Morgan. 1999. Conceptual sampling for terminographical corpus analysis. In *Terminology and Knowledge engineering*, pages 256–267, Vienna. TermNet.

Roberto Navigli and Paola Velardi. 2010. Learning word-class lattices for definition and hypernym extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1318–1327, Uppsala, Sweden.

Jennifer Pearson. 1998. *Terms in Context*, volume 1 of *Studies in Corpus Linguistics*. John Benjamins, Amsterdam / Philadelphia.

Melanie Reiplinger, Ulrich Schäfer, and Magdalena Wolska. 2012. Extracting glossary sentences from scholarly articles: A comparative evaluation of pattern bootstrapping and deep analysis. In *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*, pages 55–65, Jeju Island, Korea.

Alain Rey. 1995. *Essays on Terminology*, volume 9 of *Benjamins Translation Library*. John Benjamins, Amsterdam.

Santanu Saha Ray. 2013. *Graph Theory with Algorithms and its Applications. In Applied Science and Technology*. Springer India, New Delhi/Heidelberg/New York/Dordrecht/London.

Helmut Schmid. 2000. Unsupervised learning of period disambiguation for tokenisation. Technical report, IMS, University of Stuttgart.

Anne-Kathrin Schumann. 2014. *Linguistische Analyse und korpusbasierte Extraktion deutscher und russischer wissenshaltiger Kontexte*. Dissertation, Universität Wien, Wien.

Ray Smith. 2007. An overview of the tesseract ocr engine. In *Ninth International Conference on Document Analysis and Recognition*, pages 629–633, Parana, Brasilien.

Daniel Solow. 1990. *How to read and do proofs. An introduction to mathematical thought processes*, second edition edition. John Wiley & sons, New York.

Angelika Storrer and Sandra Wellinghoff. 2006. Automated detection and annotation of term definitions in german text corpora. In *Proceedings of The 5th Language Resources and Evaluation Conference*, pages 2373–2376.

Joop van Dormolen and Orit Zaslavsky. 2003. The many facets of a definition: The case of periodicity. *The Journal of Mathematical Behavior*, 22(1):91 – 106.

Natalia Vanetik, Marina Litvak, Sergey Shevchuk, and Lior Reznik. 2020. Automated discovery of mathematical definitions in text. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2086–2094, Marseille, France. European Language Resources Association.

Eline Westerhout. 2009. Definition extraction using linguistic and structural features. In *Proceedings of the 1st Workshop on Definition Extraction*, pages 61–67, Borovets, Bulgaria. Association for Computational Linguistics.