International Journal of Computational Linguistics and Applications
vol. 7, no. 2, 2016, pp. 145–165
Received 08/02/2016, accepted 09/03/2016, final 16/09/2016
ISSN 0976-0962, http://ijcla.bahripublications.com

# Term Evaluator: A Tool for Terminology Annotation and Evaluation

DIANA INKPEN[1], T. SIMA PARIBAKHT[1], FARAHNAZ FAEZ[2], AND EHSAN AMJADIAN[3]

[1] *University of Ottawa, Canada*
[2] *University of Western Ontario, Canada*
[3] *Carleton University, Canada*

**Abstract**. There are several methods and available tools for terminology extraction, but the quality of the extracted terms is not always high. Hence, an important consideration in terminology extraction is to assess the quality of the extracted terms. In this paper, we propose and make available a tool for annotating the correctness of terms extracted by three term-extraction tools. This tool facilitates term annotation by using a domain-specific dictionary, a set of filters, and an annotation memory, and allows for post-hoc evaluation. We present a study in which two human judges used the developed tool for term annotation. Their annotations were then analyzed to determine the efficiency of term extraction tools by measures of precision, recall, and F-score, and to calculate the inter-annotator agreement rate.

## 1. Introduction

In the field of Natural Language Processing (NLP), Terminology Extraction (TE) is a subtask of information extraction. Its goal is to automatically extract relevant terms from a given corpus. The present study is part of an endeavor towards finding available efficient terminology extraction software tools for extracting subject-specific terminology from academic textbooks. Hence, an immediate concern

was to assess the extraction performance of these tools. In this paper, we present a tool that we developed to facilitate the annotation task and the term extraction evaluation.

A significant component of any academic and educational subject is its terminology. Knowledge of the terminology of a field enables students to engage with their discipline more effectively by enhancing their ability to understand the related academic texts and lectures, and allowing them to use the subject-specific terminology in their discussions, presentations and assignments. Therefore, generating lists of terminology specific to various fields of study is a significant endeavor. However, these lists have often been generated manually or through corpus-based studies, which are time consuming, labor-intensive, and prone to human error. Therefore, an automatic terminology extraction procedure can facilitate this work to a great extent.

Terminology extraction has many direct applications in NLP, such as information retrieval, machine translation, parsing sublanguages, question-answering, and ontology construction. It underwent a rapid rise and growth throughout the nineties, and computational terminology diversified into many subtasks (Nazarenko and Zargayouna, 2009), including relation extraction, variation calculus, and term normalization. We recognize the subtask decomposition protocol (see section 2 for details) proposed by Nazarenko and Zargayouna (2009), but in this study we focus only on evaluating the terminology extraction subtask.

Terminology extraction has traditionally been accomplished by using three different methods, namely, linguistic, statistical, and hybrid, and according to two major criteria: termhood and unithood (Castellví et al., 2001, Chung, 2003). These TE methods have been applied to both monolingual and multilingual corpora (Ljubešic et al., 2012). Termhood is the degree of a linguistic unit being related to a domain-specific concept, and unithood is defined as the degree of stability of the syntagmatic combination (Kageura and Umino, 1996).

In the next section, we discuss previous work related to TE evaluation. In Section 3, we introduce the tools under evaluation and provide details on their extraction methods. Section 4 briefly outlines our corpus and how it was compiled. Section 5 introduces the developed term evaluator tool, some of its main functionalities, and its user interface. Section 6 provides the details of the annotation process.

The analysis and the results are presented in Section 7. Section 8 concludes the presented work and discusses its future directions.

## 2. Related Work on Terminology Evaluation

CoRReCT was one of the first to present a data set and protocol for term recognition in corpora. The task consisted of taking a corpus and terminology as inputs and indexing the corpus with the terms in their standard and variant forms (Enguehard, 2003).

CESART offered a complete evaluation project (Mustafa et al., 2006), involving 3 tasks: term extraction, controlled indexing, and relation extraction, but only the first task led to an evaluation. CESART proposed a protocol for term extraction. A gold standard and a corresponding acquisition corpus were developed for a specific domain.

Loginova et al. (2012) manually created Reference Term Lists (RTLs) to serve as gold standards for TE evaluation of monolingual term candidate lists automatically extracted from Spanish texts in the wind energy domain. Their domain-specific text was automatically obtained by a web crawler. Their RTLs included both single-word and multi-word terms, as well as their graphical, morphological, and syntactic variants. They also accounted for paradigmatic variants of multi-word terms. To create the RTLs, they performed tokenization, part-of-speech (POS) tagging, and lemmatization on the crawled text. Terms were extracted using POS patterns. They also used "weirdness ratio" as a filter on the extracted terms. Creating gold standard RTLs has its own challenges, especially with large corpora. If it is done entirely manually, it is time-consuming; if some NLP systems are used (e.g., lemmatizers and POS taggers), their errors are escalated (Loginova et al., 2012) and some patterns may be missed. Moreover, TE tools may return some correct terms that have not been detected by the search procedure adopted to create the RTLs, and as a result a correct term may be dismissed.

Two types of error usually occur in term extraction (Love, 2000): Silence[1] is the error where the system fails to extract terminological

[1] Corresponds to false negatives in the confusion matrix for an information extraction task.

units in the text. Noise[2] is the error where the system extracts a non-terminological unit. These two errors mirror recall and precision, respectively, that are often used for measuring the performance of different methods (Frantziy et al., 2000, Fedorenko et al., 2013). To compute the performance of the tools under evaluation, we adopted the standard set of scores: precision, recall, and F-score.

## 3. Term Extraction Methods and Tools

Term extraction methods usually extract candidate terms and rank them in order to keep only those that can be considered domain-specific terms (Vasiļjevs et al., 2014). Tokenization, part-of-speech tagging and lemmatization are often employed in term extraction algorithms. To extract terms, statistical methods compare the frequency of candidate terms in the target corpus against a general reference corpus (Fedorenko et al., 2013). Linguistic methods use linguistic patterns to detect and extract terminology.

After an initial evaluation of a number of TE tools, we chose to further evaluate the capability of four promising ones for our purposes, namely, AntConc, Topia, TermoStat, and Sketch Engine, each of which is discussed below. We chose these tools because they were available for download and because they employ different term extraction methods. However, since Sketch Engine extracted a limited number of terms (see below for further details), we did not evaluate its output. Lack of availability or limited input method, size, and format were some of the disadvantages of the other tools that we looked at.

### 3.1. AntConc

AntConc (Anthony, 2012) is the first tool we examined for our term extraction. This tool is widely used in linguistics and corpus linguistics. AntConc has a dedicated keyword extraction module, but it only extracts keywords (composed of one word). Thus, we could not use this functionality as we were interested in terms[3] composed of one or more

[2] Corresponds to false positives in the confusion matrix for an information extraction task.

[3] There is a further distinction between keywords and terms. Keywords are usually extracted from one text in order to show what the text is

words. We used AntConc to extract single-word and multi-word terms by using the "Word List" and "N-Grams" modules respectively, which list the words and multi-word expressions sorted by the frequency of occurrence in the corpus. We designate this approach implemented by AntConc as our evaluation baseline which reflects the role of pure frequency for term extraction in this experiment.

### 3.2. Topia

Topia is a hybrid term extraction tool, and uses simple linguistic and statistical procedures to extract terms. We performed the term extraction task by Topia[4] using the topia.termextract 1.1.0 library. Topia uses a simplistic POS tagger which operates after tokenization; for each word, its most frequent tag is assigned as its POS tag. Then, some simple rules are applied to extract terms (e.g., excluding terms with frequency 3 and below). We modified the implementation of the Topia library by adding some checking statements (i.e., a filter) to change all the terms which contained numbers and special Unicode characters. We replaced these characters with white space and removed all the terms that included only one or two letters. Topia extracts multiword terms as well as single-word terms, and outputs a single list of terms; therefore, we implemented a script to split the list into four lists, corresponding to one of our four term categories, namely 1-word, 2-word, 3-word, and 4-word terms (see below).

### 3.3. TermoStat

TermoStat is a non-commercial web-based terminology extraction software program, and takes a single corpus file as input. It is also a hybrid term extraction system that uses both linguistic clues and statistical techniques to extract candidate terms. TermoStat extracts single-word terms, as well as multi-word terms. For extracting multi-word terms, it restricts the lexical items that can appear inside candidate terms. If a candidate term is included in a longer candidate term and never occurs independently, it is a term fragment and is consequently excluded from the candidate-term list (Drouin, 2003).

---

about; they are not necessarily domain-specific. Terms are domain-specific and are usually extracted from large corpora of the domain.

[4] Available at: https://pypi.python.org/pypi/topia.termextract/.

TermoStat computes the specificity of a (multi)word in a corpus with reference to a general corpus (described below) by means of a statistical test developed to target highly specific technical terms (See Drouin, 2003, for more details on the statistical test). There are three outcomes: SP0, SP+, and SP-, meaning the observed frequency in the corpus is consistent, significantly higher, or significantly lower, respectively, with regard to the reference corpus. SP+ constructs a corpus-specific vocabulary which Drouin (2003) calls Specialized Lexical Pivots or SLPs for short.

The reference corpus contains approximately 8 million tokens, corresponding to approximately 465,000 different word forms. It is a non-technical corpus, half of which comes from newspaper articles on a variety of subjects from the Montreal daily newspaper '*The Gazette'* published between March 1989 and May 1989. The other half of the corpus comes from the British National Corpus (BNC).

TermoStat uses Brill's Tagger to POS-tag its corpora. Any noun in SLPs may be considered a headword. It locates all the headwords within the corpus, and starts the term extraction process from right to left. TermoStat uses both the POS of the words, as well as the results of its statistical process and some part of the formatting of the corpus to determine boundaries which may delimit candidate terms. Only terms in SLPs may be qualified as boundaries. The linguistic structure of the candidate terms retrieved by TermoStat is as follows:

i. (A|N)? (A|N)? (A|N)? (A|N)? (A|N)? N[5]

All the elements in the formal language must exist in SLPs as dictated by TermoStat's formal grammar, and as observed by the above regular expression; the length of six words for a candidate term is imposed. Our corpus was fed to TermoStat for term extraction. We updated our script to split the terms extracted by TermoStat into the following categories: 1-word, 2-word, 3-word, and 4-word terms.

### 3.4. Sketch Engine

This is a tool that we investigated but did not experiment with as it proved not suitable for our purposes. Sketch Engine uses a lemmatizer,

[5] 'A' is an adjective, 'N' is a noun, '(A|N)' is a noun or an adjective, ? represents zero or one occurrence of the element immediately preceding, '___' is an element that belongs to the SLP set.

TreeTagger[6] (Schmid, 1995) for POS tagging, and the following statistical method for computing the specificity of the terms[7]:

$$(1)\quad \text{Specificity Score} = \frac{fpm_{focus} + n}{fpm_{ref} + n}$$

where:

- $fpm_{focus}$ is normalized (per million) frequency of word in focus corpus;
- $fpm_{ref}$ is normalized (per million) frequency of word in reference corpus;
- n[8] is a simple smoothing parameter to avoid division by zero (by default n = 1).

For a quick experiment with this tool, we used the default value for n. As for reference corpora, we used 3 corpora in 3 different settings and the outcome was almost identical. We used the Brown corpus (small size, approximately 1 million tokens), the British National Corpus (BNC, medium size, approximately 100 million words), and the Web corpus English TenTen 2012 (EnTenTen, large size, approximately 13 billion tokens).

Sketch Engine extracted a total of only 36 multiword terms (excluding single-word terms) and this size is not comparable to the outputs of the other 3 tools (i.e., TermoStat: 1109, Topia: 724, and AntConc: 707). This minimalism may be due to precision/recall trade off enforced by its algorithm for practical purposes. We did not further evaluate Sketch Engine's output for the comparability reason stated above.

---

[6] Sketch Engine uses the grammatical relations (extracted by its engine) for multi-word term extraction.

[7] More about Sketch Engine statistics may be found at: https://www.sketchengine.co.uk/documentation/attachment/wiki/SkE/DocsIndex/ske-stat.pdf?format=raw

[8] We tested various values for this parameter but they had no significant effect on the number of extracted terms.

## 4. Corpus

The corpus that we used for evaluating the three term extraction tools comprised of five English high school mathematics textbooks: Small et al., 2005; Small and Kirkpatrick, 2007; Small et al., 2007; Kirkpatrick et al., 2007a; Kirkpatrick et al., 2007b. We converted the PDF files of the books into plain text, and then concatenated all the text files into one corpus consisting of 1,127,987 tokens.

## 5. Our Term Evaluator Tool

Term Evaluator is a tool we developed and made publically available[9] to facilitate the procedure of comparing the performance of the term extraction tools. It provides a user-friendly interface that speeds up the annotation process. We call this semi-automatic approach "post-hoc evaluation" and describe it below in more details.

The extracted terms were fed into Term Evaluator for annotation. Term Evaluator allows a user to start a fresh evaluation, resume a previous one, load a saved evaluation, and compare two or more evaluations. Users can also configure the term filters and load term lists. A technical dictionary comprised of three merged online mathematics dictionaries[10] is built in the tool. A secondary list of terms on whose correctness the annotators had already agreed (from previous annotation experiments, if any) may also be uploaded. Users can, however, choose not to use the built-in math dictionary, replace it with another dictionary for the same domain, or a dictionary for another domain. If required, Term Evaluator can perform two automatic operations (filtering) on any input: a) It filters out every term from the list that is a stop word (omission of terms), and b) It drops the stop

[9] TermEvaluator can be downloaded and used for free at https://sourceforge.net/projects/termevaluator/

[10] The dictionary belongs to the mathematics domain and was retrieved and compiled from the following three sources:
-Illustrated Mathematics Dictionary. (n.d.). Retrieved 2013. (http://www.mathsisfun.com/definitions/index.html)
-Mathwords. (n.d.). Retrieved 2013. (http://www.mathwords.com/)
-Math Dictionary. (n.d.). Retrieved 2013. (http://www.mathematicsdictionary.com/math-vocabulary.htm)

word portion if a term starts or ends with a stop word (change of terms). Figure 1 presents the evaluation interface where annotators can assess the extracted terms. They have access to the rank, frequency, and the termhood score of the term at hand and they can mark the category of each term as Yes (technical[11]), Non-Technical, No (non- term[12]), and Not Sure.

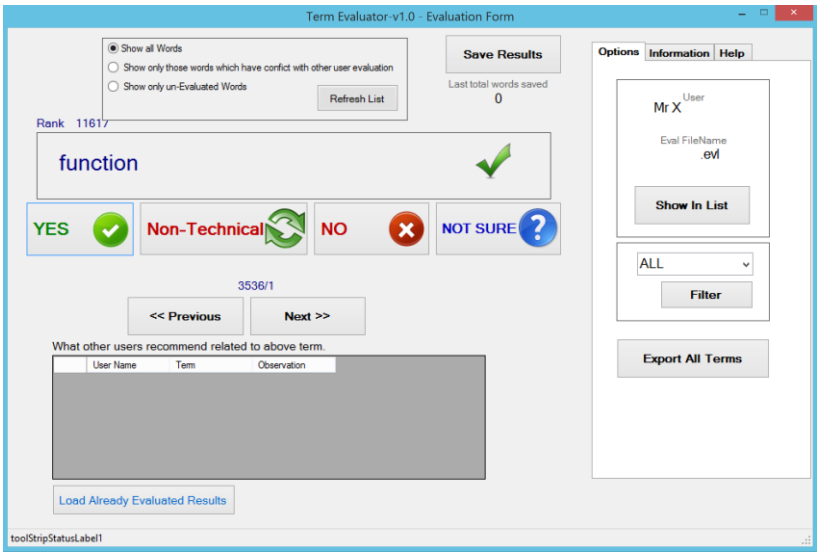


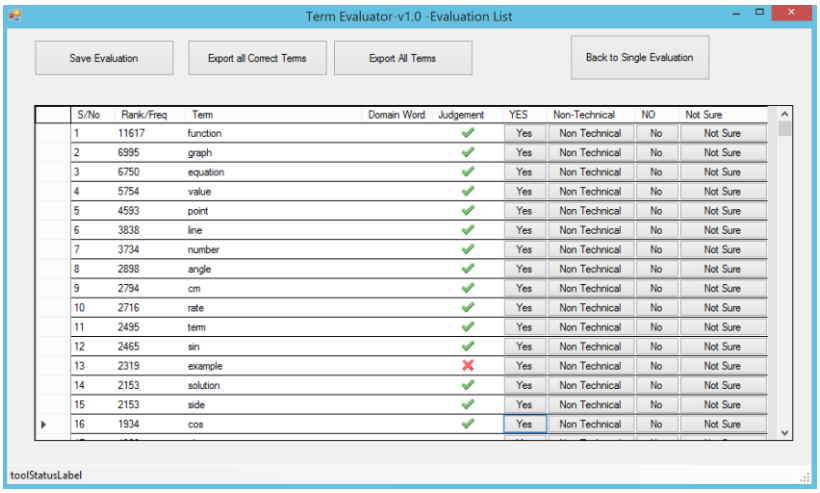

Figure 1. The annotation window

The annotators have the option to view only the items not evaluated before or only those in conflict with other annotators' evaluations. They can also save the evaluation and return to it at a later time. In addition, a list view is available to show all the terms including those annotated or to be annotated (see Figure 2). Correct terms may be exported at any time during annotation.

Term Evaluator can compare different evaluations, show the number of agreements/disagreements, intersection of annotation

[11] Also referred to as "correct" in our tool
[12] Also referred to as "wrong term" in our tool

decisions, inter-annotator agreement rate,[13] and a few more statistics and comparison details (see Figure 3).




| S/No | Rank/Freq | Term | Domain Word | Judgement | YES | Non-Technical | NO | Not Sure |
|---|---|---|---|---|---|---|---|---|
| 1 | 11617 | function | | ✔ | Yes | Non Technical | No | Not Sure |
| 2 | 6995 | graph | | ✔ | Yes | Non Technical | No | Not Sure |
| 3 | 6750 | equation | | ✔ | Yes | Non Technical | No | Not Sure |
| 4 | 5754 | value | | ✔ | Yes | Non Technical | No | Not Sure |
| 5 | 4593 | point | | ✔ | Yes | Non Technical | No | Not Sure |
| 6 | 3838 | line | | ✔ | Yes | Non Technical | No | Not Sure |
| 7 | 3734 | number | | ✔ | Yes | Non Technical | No | Not Sure |
| 8 | 2898 | angle | | ✔ | Yes | Non Technical | No | Not Sure |
| 9 | 2794 | cm | | ✔ | Yes | Non Technical | No | Not Sure |
| 10 | 2716 | rate | | ✔ | Yes | Non Technical | No | Not Sure |
| 11 | 2495 | term | | ✔ | Yes | Non Technical | No | Not Sure |
| 12 | 2465 | sin | | ✔ | Yes | Non Technical | No | Not Sure |
| 13 | 2319 | example | | ✘ | Yes | Non Technical | No | Not Sure |
| 14 | 2153 | solution | | ✔ | Yes | Non Technical | No | Not Sure |
| 15 | 2153 | side | | ✔ | Yes | Non Technical | No | Not Sure |
| 16 | 1934 | cos | | ✔ | Yes | Non Technical | No | Not Sure |

Figure 2. Annotation list view

## 6. The Annotation Process

After term extraction was performed by the term extraction tools, a cut-off value was applied to each of the four word categories (i.e., 1-word, 2-word, 3-word, and 4-word). The outputs that were already below the threshold, remained intact. The cut-off value was set at 500[14] (if the list of candidate terms was shorter than the cut-off value, the whole list was retained). For each of the three tools, 4 files were submitted to the annotators corresponding to one of the four term categories.

Two human annotators (one male and one female) judged the terms extracted by the term extraction tools. The annotators were instructed to use the Term Evaluator software to judge the terms using one of the

[13] Agr = Na / (Na+Nd) where a:agreement and d:disagreement

[14] In a further experiment discussed at the end of section 6, we were also able to annotate all the terms extracted by TermoStat (4011 terms), and the task was still feasible for our annotators.

following four options that are provided as buttons in Figure 1 and Figure 2: A) YES [technical term], B) Non-Technical [generic English term], C) NO (non-term, D) Not Sure.

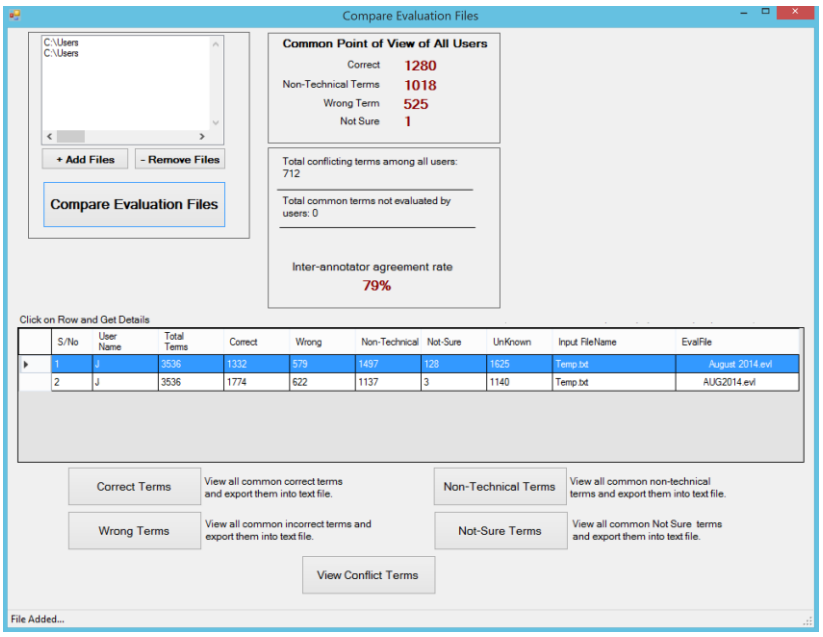



Figure 3. The comparison window[15]

The definitions of these options were provided and the annotators were asked to use their background knowledge of mathematics as the primary source of their judgment. In case of confusion, they could consult a Mathematics dictionary of their choice.

## 7. Results and Analysis

We computed precision, relative recall and balanced F-score for each tool. Relative recall is computed against the union of all the predicted

[15] In this figure the term "correct" refers to technical terms and "wrong terms" refer to non-terms.

correct terms among the term extraction tools, with two categories: correct[16] and incorrect[17]. The performance of the 3 tools is compared in Figures 4-7 below and in Table 1. Figure 4 shows the performance of the tools for extracting terms that contain only one word. Topia, with the added filter (see section 3.2 for details) outperformed the other tools for single-word terms, and had the highest precision, recall and F-score. This is interesting, considering that Topia does not use any sophisticated algorithm. In terms of precision, AntConc comes second and TermoStat last and regarding recall, TermoStat performs better than AntConc. This is also interesting. As mentioned earlier (section 3.1), AntConc extracts terms based on basic frequency. The fact that TermoStat has a better recall than AntConc (53% vs. 51% respectively) can be an indication that bare frequency may not be sufficient to extract correct terms in a technical corpus. On the other hand, the fact that AntConc achieved a better precision than TermoStat (41% vs 37% respectively) confirms the intuition that single words that are frequent in a technical corpus have a high chance of being identified as a term specific to that corpus.

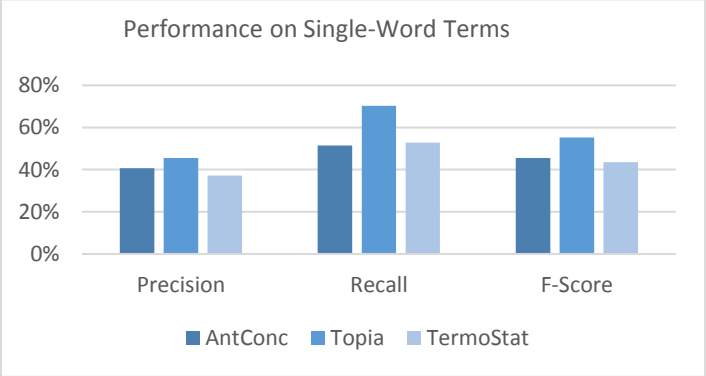



Figure 4. Comparison of the performance of the term extraction tools in extracting single-word terms

[16] Technical terms
[17] Non-technical, non-term, and not sure

Figure 5 presents the performance of the tools in extracting two-word terms from the math corpus. There are a few points that deserve further attention. TermoStat shows a leap from single-word (F-score of 44%) to two-word term extraction (F-score of 67%). Its precision has improved with 31 percentage points and its recall with 13 percentage points. This makes TermoStat the highest performing tool for the two-word term category. This high performance manifests an adequate account of termhood and unithood. Topia is keeping up although it suffers from a simplistic POS tagger as compared to TermoStat that features the well-known and well-performing Brill's tagger (Brill, 1992). POS tagging comes more into play as the number of terms in a multi-word expression increases. The other factor worth mentioning is the competitive precision of AntConc (albeit its low recall scores) that postulates frequent n-grams have a high chance of being terms. It is possible that AntConc's high performance on single-word terms is due to chance (i.e., unigrams); after all, frequent words are probable to be terms.

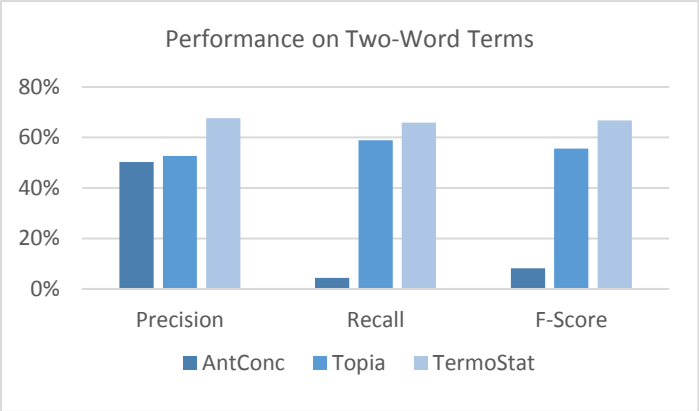



Figure 5. Comparison of the performance of the term extraction tools in extracting two-word terms

Figure 6 depicts the performance of the tools in extracting three-word terms. What appears striking at first glance is Topia's extreme reduction in performance. TermoStat consistently has the highest precision and recall.

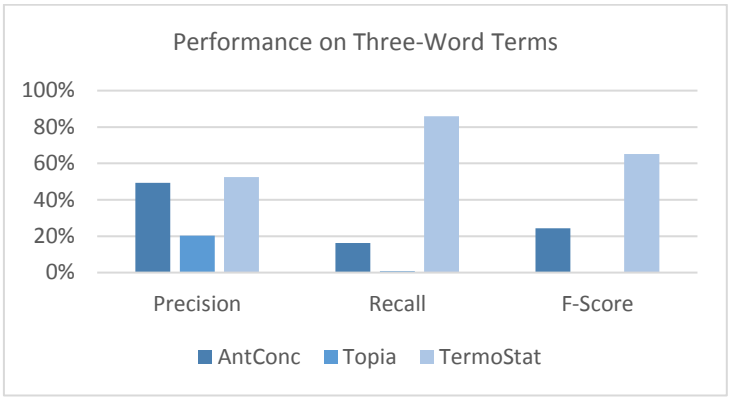



Figure 6. Comparison of the performance of the term extraction tools in extracting three-word terms

Figure 7 presents the performance of the tools in extracting four-word terms. TermoStat still has the lead in both precision and recall. AntConc still has a higher precision than recall and keeps following the same trend as in the two-word and three-word categories. Therefore, except for the single-word terms, n-gram raw frequency does not seem to compete with a proper term extraction algorithm. Topia's performance stays poor for the four-word category (11% precision and 6% recall).

We computed the overall performance of each tool (Table 1). TermoStat achieved the highest scores due to its solid statistical measure, good performing POS tagger, and its extraction patterns. Topia achieved higher than AntConc n-grams for one- and two-word categories. Nevertheless, Topia's low performance in extracting 3-word and 4-word terms coupled with a somewhat constant precision of AntConc n-grams over the 4 categories, gave AntConc the second place in overall performance. Topia had a better overall recall score than AntConc, but a worse precision.

Table 2 presents the agreement[18] scores in percentage between the annotators as provided by the Term Evaluator. The bottom row shows

[18] Since non-expert judges were used in this study, when computing agreement scores we collapsed the categories Non-Technical, Wrong

the overall agreement for each tool across all categories. Annotators agreed on AntConc results the most, followed by Topia, and TermoStat. We consider our data non-sequential and have computed Cohen's kappa statistics for inter-annotator agreements (Carrillo et al., 2014, Viera and Garrett, 2005).

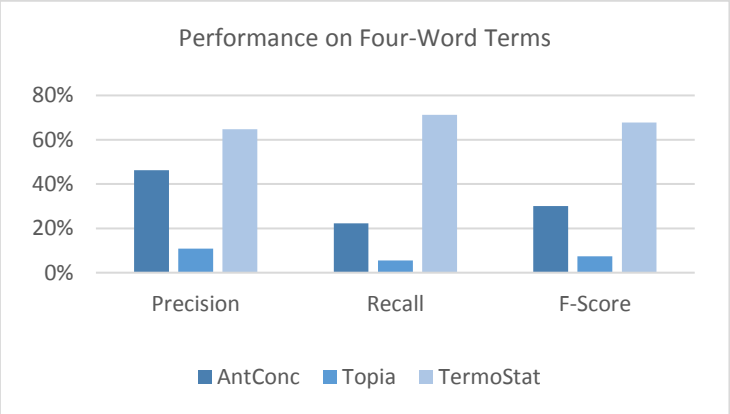



Figure 7. Comparison of the performance of the term extraction tools in extracting four-word terms

Table 3 shows the kappa statistics for agreement scores for each word category and each tool. The kappa values are consistent with Term Evaluator's agreement scores in that the highest overall agreement belongs to AntConc, followed by Topia, and TermoStat comes last (see Table 3). It is noteworthy that the kappa value for Topia for the 4-word category is very low, which coincides with the lowest performance in Table 1. Low agreements may often occur in term extraction (Vivaldi and Rodriguez 2007, and Loginova et al., 2012), but this specific case is due to the very high P(e) value for Topia in the 4-word category, which is equal to 0.81. This partially originates from the tool's noisy output for this word category which resulted in a very low correctness score (7%) for one of the annotators.

---

Term (i.e. non-terms/No), and Not Sure (see section 5 figures 1 and 2) into one, called *incorrect*.

Table 1. Overall performance (in percentage) of the term extraction tools in our corpus

| Overall Performance | Precision | Recall | F-score |
|---|---|---|---|
| AntConc | 47% | 45% | 46% |
| Topia | 32% | 55% | 41% |
| TermoStat | **55%** | **64%** | **59%** |

Table 2. Agreement scores in percentage between annotators for each word category and each tool, and overall agreement

| **Inter-annotator agreement** | AntConc | Topia | TermoStat |
|---|---|---|---|
| 1-word | 92% | 85% | 78% |
| 2-word | 93% | 83% | 73% |
| 3-word | 87% | 84% | 66% |
| 4-word | 73% | 81% | 68% |
| Overall | 86% | 84% | 72% |

Table 3. Cohen's kappa statistics for annotators per word category and per tool, and overall

| **Kappa** | AntConc | Topia | TermoStat |
|---|---|---|---|
| 1-word | 0.84 | 0.70 | 0.53 |
| 2-word | 0.86 | 0.67 | 0.40 |
| 3-word | 0.73 | 0.54 | 0.48 |
| 4-word | 0.47 | 0.05 | 0.36 |
| All | 0.71 | 0.62 | 0.49 |

We investigated the cause of disagreements between the two annotators, by asking them to discuss the cases of disagreement. Annotator 1 evaluated fewer terms as "Not Sure", while Annotator 2 was more uncertain about whether the corresponding terms pertained to mathematics. Annotator 2 had an issue with 2-word terms in which one word was a mathematics word and the other was not. Examples include the candidate terms "combined function" and "resultant velocity". Another source of confusion was the signs and symbols that cannot be considered words. One annotator marked many of them as "Non-Term", and the other as "Not Sure". Numerals, such as "ii" also caused

problems: Annotator 1 marked them as "Not Sure", while Annotator 2 considered them mathematics terms.

In a follow up experiment, we asked our human judges to annotate all the 4011 terms extracted by TermoStat, in order to measure how much time they save by using our Term Evaluator tool. From these, 475 terms had already been filtered out by our tool because they started or ended in stop words, 501 had automatically been marked as good terms because they were found in the domain dictionaries included in the tool, and 368 had also automatically been marked because they were in the secondary list of terms already evaluated as correct terms in the previous experiments. This left 2667 terms to be annotated, which represents a saving of 33%.

## 8. Conclusion and Future Work

This study investigated the performance of three terminology extraction tools on a corpus of school mathematics textbooks. An evaluation tool (TE) was developed and made publically available to facilitate and speed up the annotation task. The tool benefits from a default domain term dictionary and a secondary list (term memory), which can hold in memory all the terms previously marked as correct by annotators. The results indicated that our Term Evaluator eliminated the need to annotate 1344 of the 4011 words, representing 33% of the terms extracted by TermoStat, which resulted in a significant saving in evaluation time.

The results also suggest that of the three tools examined, TermoStat, with stable high precision and recall scores, is the most suitable tool for technical term extraction in a corpus of mathematics textbooks, validating the efficiency of its patterns and statistical test. The apparent lower performance of TermoStat for the single-word category may have been caused by some term extraction and annotation related issues. For instance, words such as 'two' or 'three' had been extracted by the other tools and marked as correct terms by annotators, but TermoStat regards these words general-domain terms. Another issue is some inconsistency in annotation that can be prevented by the Term Evaluator's memory, if used. There were terms marked as incorrect for TermoStat and correct for the other tools by the annotators (e.g., calculator, speed). A further issue may have arisen due to lack of

efficient preprocessing in Topia and AntConc. AntConc and Topia extract terms like "zeros" as technical, whereas TermoStat does not. That is, AntConc and Topia do not recognize inflection, which in turn results in candidate terms such as "zero" and "zeros" both being evaluated as correct by the annotators. TermoStat, on the other hand, benefits from proper preprocessing and recognizes "zeros" as an inflected form of "zero. This can make TermoStat's recall seem lower than it actually is. Finally, it is worthwhile noting that various single-word mathematics terms (e.g., addition, number, and calculator) may be hard to judge as technical or not, especially since these terms are frequently used in general English.

In future work, we plan to modify and improve the present study's best performing term extraction algorithm to achieve a higher performance. We will expand the study to other technical domains, will use judges with expertise in mathematics for annotation, and will compare the results with those obtained in this study. Term extraction evaluation in other languages (e.g., French) would be a further direction of this research. The tool currently memorizes only the correct terms for automatic domain-specific annotation. In future research, we intend to assign other automatic decision categories to the tool as well. Another future improvement can be augmenting the tool with other agreement coefficients.

## References

1. Anthony, L. 2012. AntConc (Version 3.3.0) [Computer Software]. Waseda University, Tokyo, Japan. Available at http://www.laurenceanthony.net/
2. Brill., E. 1992. A simple rule-based part of speech tagger. In *Proceedings of the third conference on Applied natural language processing* (ANLC '92). Association for Computational Linguistics, Stroudsburg, PA, USA, 152-155.
3. The British National Corpus, version 3 (BNC XML Edition). 2007. Distributed by Oxford University Computing Services on behalf of the BNC Consortium. URL: http://www.natcorp.ox.ac.uk/
4. Carrillo, H., Brodersen, K., H., and Jose A. Castellanos. 2014. Probabilistic performance evaluation for multiclass classification using the posterior balanced accuracy. Advances in Intelligent Systems and Computing, 252:347-361.

5. Chung, T., M. 2003. A corpus comparison approach for terminology extraction. Terminology, 9(2):221–246.
6. Drouin, P. 2003. Term extraction using non-technical corpora as a point of leverage. *Terminology, 9*(1):99-115.
7. El Hadi W., M., Timimi, I., Dabbadie, M., Choukri, K., Hamon, O., and Chiao, Y. 2006. Terminological resources acquisition tools: Toward a user-oriented evaluation model. *In Proceedings of the Language Resources and Evaluation Conference (LREC'06)*, 945–948. Genova, Italy.
8. Engelmore, R., and Morgan, A. eds. 1986. *Blackboard Systems.* Reading, Mass.: Addison-Wesley.
9. Enguehard, C. 2003. Correct: Demarche cooperative pour l'evaluation de systemes de reconnaissance de termes. In Actes de la 10eme conference annuelle sur le Traitement Automatique des Langues (TALN 2003), 339–345. Nancy.
10. Fedorenko, D., Astrakhantsev, N., and Turdakov, D. 2013. Automatic recognition of domain-specific terms: an experimental evaluation. Proceedings of the Ninth Spring Researcher's Colloquium on Database and Information Systems, Kazan, Russia.Frantziy, K., Ananiadouy, S., and Mimaz, H. 2000. Automatic Recognition of Multi-Word Terms: the C-value/NC-value Method. *International Journal on Digital Libraries*, 3(2):115-130.
11. Francis, W., N., and Kučera, H. 1964, 1971, 1979. A Standard Corpus of Present-Day Edited American English, for use with Digital Computers (Brown). Compiled by Brown University. Providence, Rhode Island.
12. Kageura, K., and Umino, B. 1996. Methods of Automatic Term Recognition. *Papers of the National Center for Science Information Systems*: 1-22.
13. Kilgarriff, A., Baisa, V., Bušta, J., Jakubíček, M., Kovář, V., Michelfeit, J., Rychlý, P., Suchomel, V. 2014. The Sketch Engine: ten years on. In Lexicography: 1–30.
14. Kirkpatrick, C., Alldred, B., Chilvers, C., Farahani, B., Farentino, K., Lillo, A., Macpherson, I., Rodger, J., and Trew, S. 2007. Nelson Advanced Functions.
15. Kirkpatrick, C., Crippin, P., Donato, R., and Wright, D. 2007. Nelson Calculus and Vectors.
16. Ljubešic, N., Vintar, Š., Fišer, D. 2012. Multi-word term extraction from comparable corpora by combining contextual and constituent clues. 5th Workshop on Building and Using Comparable Corpora, 143-147.
17. Loginova, E., Gojun, A., Blancafort, H., Guegan, M., Gornostay, T., and Heid, U. 2012. Reference Lists for the Evaluation of Term Extraction Tools. Proceedings of the 10th Terminology and Knowledge Engineering Conference. Madrid, Spain.

18. Love, S. 2000. Benchmarking the performance of two automated term-extraction systems: LOGOS and ATAO. Master's Thesis. Université de Montréal.
19. Masaryk University, NLP Centre. 2011. enTenTen, LINDAT/CLARIN digital library at Institute of Formal and Applied Linguistics, Charles University in Prague, http://hdl.handle.net/11858/00-097C-0000-0001-CCDF-8.
20. Small, M., and Kirkpatrick, C. 2007. Nelson Functions 11.
21. Nazarenko, A., Zargayouna, H. 2009. Recent Advances in Natural Language Processing (RANLP) 299–304.
22. Schmid, H. 1995. Improvements in Part-of-Speech Tagging with an Application to German. Proceedings of the ACL SIGDAT-Workshop. Dublin, Ireland.
23. Small, M., Kirkpatrick, C., Zimmer, D., Chilvers, C., D' Agostino, S. Duff, D., Farentino, K., Macpherson, I., Tonner, J., Williamson, J., and Yeager, T., A. 2005. Nelson Principles of Mathematics 9.
24. Small, M., Kirkpatrick, C., Dmytriw, A., Farahani, B., Lillo, A., Minter, K., Pilmer, D., and Walker, N. 2007. Nelson Functions and Applications 11.
25. Stephan Richter, Russ Ferriday and the Zope Community. 2009. [Computer Software]. Available from: https://pypi.python.org/pypi/topia.termextract/
26. Teresa, M., Castellví, C., Bagot, R., E., Palatresi, J., V., 2001. Automatic Term Extraction: a review of current systems. In Bourigault, D.; Jacquemin, C.; L'Homme, M-C (eds). *Recent Advances in Computational Terminology*, 53-88.
27. Vasiļjevs, A., Pinnis, M, Gornostay, T. 2014. Service model for semi-automatic generation of multilingual terminology resources. Terminology and Knowledge Engineering (TKE).
28. Viera, A., J., and Garrett, J., M. 2005. Understanding inter-observer agreement: the kappa statistic. *Family Medicine*, 37:360-363.
29. Vivaldi, J, and Rodriguez, H. 2007. Evaluation of terms and term extraction systems: A practical approach. *Terminology*, 13(2).

**DIANA INKPEN**
UNIVERSITY OF OTTAWA,
CANADA
E-MAIL: <DIANA.INKPEN@UOTTAWA.CA>

**T. SIMA PARIBAKHT**
UNIVERSITY OF OTTAWA,
CANADA
E-MAIL: <SIMA.PARIBAKHT@UOTTAWA.CA>

**FARAHNAZ FAEZ**
UNIVERSITY OF WESTERN ONTARIO,
CANADA
E-MAIL: <FFAEZ@UWO.CA>

**EHSAN AMJADIAN**
CARLETON UNIVERSITY,
CANADA
E-MAIL: <EHSAN.AMJADIAN@CARLETON.CA>