Proceedings of the TERM21 Workshop , pages 19–25
Language Resources and Evaluation Conference (LREC 2022), Marseille, 20-25 June 2022

A Dataset for Term Extraction in Hindi

Shubhanker Banerjee, Bharathi Raja Chakravarthi, John Philip McCrae
ADAPT Centre
National University Of Ireland Galway
shubhanker.banerjee@adaptcentre.ie

Abstract

Automatic Term Extraction (ATE) is one of the core problems in natural language processing and forms a key component of text mining pipelines of domain specific corpora. Complex low-level tasks such as machine translation and summarization for domain specific texts necessitate the use of term extraction systems. However, the development of these systems requires the use of large annotated datasets and thus there has been little progress made on this front for under-resourced languages. As a part of ongoing research, we present a dataset for term extraction from Hindi texts in this paper. To the best of our knowledge, this is the first dataset that provides term annotated documents for Hindi. Furthermore, we have evaluated this dataset on statistical term extraction methods and the results obtained indicate the problems associated with development of term extractors for under-resourced languages.

Keywords: automatic term extraction, under-resourced, Hindi

1. Introduction

Automatic Term Extraction (ATE) is the task of extracting relevant terms from domain specific corpora. Terms can be defined as linguistic units that refer to domain specific concepts in a world model (Cabré, 1999; Cram and Daille, 2016; Peñas et al., 2001b). To illustrate, in the domain of education an institution that imparts education to children is a concept and we refer to this concept by the word *school* in English. Thus, identification of a word or a multiword expression as a term is highly dependent on the individual's subjective notion of the concept (Peñas et al., 2001b), for example how does the individual define education and whether institution is an important concept in this domain as per the subjective opinion of the person. This in turn makes ATE a more challenging problem to tackle.

Term extractors also play a critical role in ontology engineering as identification of terms and relationships amongst them can be used to identify important concepts and conceptual relations which in turn serve as building blocks for ontologies (Pazienza et al., 2005). They are also used in the development of language technology such as machine translation (Oliver, 2017) and summarization systems (Jacquemin and Bourigault, 2005). Furthermore, they serve as the key building blocks of information retrieval systems as they allow efficient indexing of relevant documents (Jacquemin and Bourigault, 2005) together on the basis of terms thus playing a critical role in reducing the overall search time and improving the scalability of these systems.

Although term extraction has been an active area of research in the past few decades, most of the research in this domain has primarily been focused on English (Pazienza et al., 2005; Šajatović et al., 2019; Astrakhantsev, 2018; Zhang et al., 2018). Out of the 7,100+ languages[1] being used around the world most are under-resourced with limited access to language technology tools. In this paper, we present a novel dataset for term extraction in the education domain for the Hindi language. Furthermore, we have carried out experiments with statistical term extraction systems on this dataset to demonstrate the challenges in building term extractors. We hope that by releasing this dataset we can contribute towards the research in resource creation as well as development of better algorithms for term extraction for under-resourced languages.

The related works have been discussed in Section 2, Section 3 details the dataset collection techniques as well as processing carried before conducting the experiments. Furthermore, we go onto list the data statistics in terms of the total number of annotated documents and the methodology followed while annotating the documents. The algorithms used to carry out the experiments are discussed in Section 4 followed by a discussion on the evaluation metric and the experiments in Section 5 and Section 6 which reviews the results obtained. Lastly, future directions of research and open problems are described in Section 7.

2. Related Works

ATE has been an active area of research since the last decade of the previous millennium (Daille, 1994; Evans and Lefferts, 1995; Pazienza, 1998) with almost all of the research in this domain being focused on statistical techniques; both supervised and unsupervised.

[1] `https://www.ethnologue.com/`

Earlier research in this domain was focused around frequency based measures such as Term Frequency - Inverse Document Frequency (TF-IDF) (Evans and Lefferts, 1995) and linguistic filter based methods (Daille, 1994). The key idea behind the TF-IDF based methods is that terms representative of important concepts have high document term frequency in a few documents. The methods based on linguistic filtering exploit general syntactic patterns observed in terms across domains for example associating noun phrases with terms. Bordea et al. (2013) propose a term recognition algorithm based on the the identification of termhood of the term constituents. In recent years, this ongoing research has culminated in the form of various term extraction toolkits, namely: TermSuite (Cram and Daille, 2016), Simple Extractor[2], SDL MultiTerm Extract[3], Terminus[4], JATE (Zhang et al., 2016), Rainbow [5] and ATR4S (Astrakhantsev, 2018) which have advanced the state-of-the-art in term extraction tasks.

The experiments carried out in this paper are based on the frequency based algorithms demonstrated by Astrakhantsev (2018). They carry out experiments with various methods such as methods based on occurrence frequency, methods based on topic modelling and context modelling based methods.

Term annotated datasets are available for popular highly resourced languages such as English, however not a lot of progress has been made with regards to curation of term annotated datasets for under-resourced languages. GENIA (Kim et al., 2003) is term annotated dataset for the domain of biomedicine in English. It contains 2000 abstracts taken from the MEDLINE database comprising of over 400,000 tokens and annotated with 93,293 terms. The CRAFT corpus (Bada et al., 2012), belonging to the biomedical domain is another popular term annotated dataset for domain specific terminology in English. Similary, there are other datasets available for English such as ACL RD-TEC (**?**) and ACTER [6]. This research is closely related to the work done by McCrae and Doyle (2019) who introduce a term annotated dataset for Irish (ISO 639-3 language code for Irish is gle), an under-resourced language of the Goidelic family of languages. The Goidelic languages are a part of the larger Celtic family of languages used primarily in the British Isles. Irish, Scottish Gaelic and Manx are the 3 languages which constitute the Goidelic family. They demonstrate term extraction on this dataset using various methods such as frequency based measures and topic modelling based approaches. Also, they propose the inclusion of morphological features in the term recognition pipeline in order to improve the performance of the statistical term recognition system. In this paper, we have used frequency and background corpus based measures for term extraction.

2.1. Hindi

Hindi is an under-resourced language from the Indo-European family of languages primarily used in the northern and north-western parts of the Indian subcontinent (Kachru, 2006). There are 528 million native speakers of the language in the Indian subcontinent as per the census of 2011 conducted by the Government of India[7]. Hindi is written in the Devanagari script which is a phonetic script; the writing system reflects the pronunciations (Bright, 1996). For example, dog is written as श्वान in Hindi and prounounced as *shvaan*.

3. Dataset

The dataset introduced in this paper is a collection of documents from the domain of education. The choice of this particular domain is not due to a specific scientific reason but influenced by the authors' expertise in this domain by virtue of previous and current academic affiliations. Data required for annotation has been collected from Wikipedia using its standard api [8]. A total of 71 Wikipedia documents comprising of 11,960 words were collected and manually annotated. Wikipedia pages are classified into categories to group pages from the same domain together. In Hindi Wikipedia, category is known as श्रेणी (transliteration - shreni, translation - category) and setting this parameter to शिक्षा (transliteration - shiksha, translation - education) in the API GET request enables us to download the documents belonging to this domain. During annotation we referred to the fundamental definition of a term: *term is a surface representation of a concept* (Pazienza, 1998). Of course, the definition of concepts is subjective and reflects the annotators' notion of termhood in the given domain. It is also important note that we haven't posed any syntactic structure on term selection. This has been done to increase the coverage and allow the terms representative of a variety of different concepts to be annotated. However, during

[2] https://www.dail.es/en/artificial-intelligence/

[3] https://docs.rws.com/binary/796827/807059/sdl-multiterm-2021-sr1/sdl-multiterm-extract-tools-user-guide

[4] http://terminus.iula.upf.edu/cgi-bin/terminus2.0/terminus.pl?lInt=En

[5] https://okapiframework.org/wiki/index.php/Rainbow

[6] https://clarin.eurac.edu/repository/xmlui/handle/20.500.12124/24

[7] https://censusindia.gov.in/2011Census/Language_MTs.html

[8] https://hi.wikipedia.org/w/api.php

annotation we observed that almost all the annotated terms are noun phrases. In fact, a total of 926 annotated terms were noun phrases, 25 verb phrases were annotated as terms and 2 adjectives were also annotated as terms. As a part of the data cleaning, the English words in the downloaded data have been filtered out using a unicode based character filtration.
A total of 71 documents were annotated with 953 terms. Also, it is also important to note that the annotation was performed by a single annotator. For under-resourced languages like Hindi, it is difficult to onboard trained expert annotators due to the scarcity of domain experts and high expenses associated with their recruitment. In this case these challenges have limited the size of the dataset and since the dataset has been manually annotated by one annotator therefore it is difficult to ascertain the quality of annotations and the possibility of noisy annotation cannot be ruled out.
The aforementioned problems with manual annotation can hinder the learning of any meaningful representations and lead to degraded performance in the supervised learning domain. However, self-supervised learning algorithms which are robust to noisy annotation and can learn meaningful representations by seeding on the initial annotated dataset (Tan et al., 2021) can be used to train machine learning models for the task at hand.
Lastly, although the size of the dataset is relatively small, we hope that it can propel research interest in this domain for the Hindi language. The dataset and necessary code developed during its curation are available publicly [9].

4. Methodology

We evaluated the performance of frequency and reference corpora based term extraction approaches discussed by Astrakhantsev (2018) using the dataset introduced in Section 3 as the gold standard. The following are the steps involved in the term extraction pipeline:

- Pre-processing
- Term candidate selection
- Term candidate scoring and ranking[10]

The methods detailed in sections 4.2.3, 4.2.4 and 4.2.5 are based on a general domain background corpus. We used 2,411 Wikipedia articles comprising of 7,28,055 words spread across multiple domains as our background corpus [11].

[9] `https://github.com/zigzagthad/Hindi_Term-Extract`

[10] In this paper, we have used one method for term scoring therefore ranking is trivial. However, in methods where multiple term scoring methodologies are involved, term ranking becomes complicated.

[11] `https://rb.gy/d5o4yi`

4.1. Term Candidate Selection

We used part-of-speech chunking for filtering the term candidates. Firstly, we annotated the documents with the TnT tagger (Brants, 2000) which is available as a part of the NLTK package. Next, we used the RegexpParser also available as a part of the NLTK package to perform chunking on the annotated documents. Precisely, all noun phrases were considered as term candidates to be scored using the methods discussed in the subsequent sections. Here it is important to note that the tnt tagger for the Hindi language is trained on 540 annotated sequences. This impedes the performance of the part-of-speech tagging step and reflects resource constraints in under-resourced scenarios.
Also, it was ensured that the selected term candidates have a length of at least 3 and in case of multi-word expressions it was ensured that all individual words constituting the expression have a length of 3 at least.

4.2. Term Scoring and Ranking

The term candidates selected in the previous step were scored with the following 5 different methods as proposed by (Astrakhantsev, 2018):

- Term Frequency - Inverse Document Frequency (TF-IDF)
- Residual Inverse Document Frequency (RIDF)
- Domain Pertinence
- Weirdness
- Relevance

For a particular scoring algorithm the term candidates were ranked in decreasing order of the scores achieved by them.

4.2.1. TF-IDF

As a part of the experiments, we carried out term scoring using the term frequency-inverse document frequency algorithm (Evans and Lefferts, 1995). It's an information retrieval algorithm that assigns higher values to terms that have high occurrence frequency in a few documents according to Equation 1. The intuition behind using this algorithm for term extraction is that terms that represent concepts in a specific domain have a high occurrence frequency in the domain-specific documents.

$$TF \cdot IDF(t) = TF(t) \cdot \log_2 \frac{D}{DTF(t)} \qquad (1)$$

where TF(t) is the term frequency, D is the total number of document in the collection, DTF(t) is a number of documents in which the term occurs.

4.2.2. RIDF

The RIDF algorithm first proposed by (Church and Gale, 1999) was used by (Zhang et al., 2016) for term extraction. The key idea behind using this approach for term extraction is that the IDF that is observed for terms has a greater deviation from a standard Poisson deviation as compared to the deviation observed for non-terms as shown in Equation 2.

$$RIDF(t) = TF(t) \cdot \log_2 \frac{D}{DTF(t)} + \log_2(1 - e^{-ATF(t)}) \quad (2)$$

where ATF is the normalized term frequency, normalized the number of documents in which a term occurs.

4.2.3. Domain Pertinence

Domain pertinence (Meijer et al., 2014) is a background corpus based term extraction method. The key idea behind background corpus based methods is that terms in a domain specific collection are different from non-terms with regards to their occurrence statistics in a background collection. Equation 3 shows the calculation of Domain pertinence for linguistic units in a domain specific corpus. For a specific term candidate, domain pertinence is calculated as a ratio of term frequency in the given corpus and the term frequency in the background corpus.

$$DomainPertinence(t) = \frac{TF_{target}(t)}{TF_{reference}(t)} \quad (3)$$

where TF_{target} is the term frequency in the domain specific corpus and $TF_{reference}$ is the term frequency in the general background corpus.

4.2.4. Weirdness

Khurshid et al. (2000) normalizes the term frequencies by the total number of the words in the respective collection.

$$Weirdness(t) = \frac{NTF_{target}(t)}{NTF_{reference}(t)} \quad (4)$$

where NTF_{target} is the term frequency in the domain specific corpus normalized by the total number of words in the domain specific corpus and $NTF_{reference}$ is the term frequency in the general background corpus normalized by the total number of words in the background corpus.

4.2.5. Relevance

(Peñas et al., 2001a) is a modification to domain pertinence and the weirdness algorithm as it takes into account document frequency in the calculation, that is the number of documents in which a term occurs.

$$Relevance(t) = 1 - (\log_2(2 + \frac{NTF_{target}(t) \cdot DF_{target}(t)}{NTF_{reference}(t)}))^{-1} \quad (5)$$

where NTF_{target} is the term frequency in the domain specific corpus normalized by the total number of words in the domain specific corpus and $NTF_{reference}$ is the term frequency in the general background corpus normalized by the total number of words in the background corpus and DF_{target} is the number of documents in the domain corpus in which a term occurs.

5. Experiments

Term extraction can be viewed as a retrieval of terms from text documents. There are primarily two kinds of retrieval evaluation algorithms, namely ranked and unranked (Manning et al., 2010) evaluation metrics. Unranked evaluation metrics don't take into account the relative ranks of the term candidates, that is the score attained by the term candidates as per scoring algorithms does not contribute to the evaluation. On the contrary, these metrics are evaluated on the basis of term candidate lists returned by the retrieval algorithm (in this case the chunker). In this paper, we have used 3 unranked evaluation algorithms namely Precision, Recall and F1 score. Precision essentially calculates the proportion of relevant terms out of the total number of retrieved terms (Manning et al., 2010) as given in Equation 6

$$Precision = \frac{Number\ of\ relevant\ items\ retrieved}{Total\ number\ of\ retrieved\ items} \quad (6)$$

Recall calculates the proportion of relevant terms out of the total number of relevant terms (Manning et al., 2010) as given in Equation 7

$$Recall = \frac{Number\ of\ relevant\ items\ retrieved}{Total\ number\ of\ relevant\ items} \quad (7)$$

F1-score is an unranked evaluation score that is calculated as the harmonic mean of Precision and Recall (Manning et al., 2010). Relative ranks of the term candidates don't contribute towards the calculation of these scores and therefore their values are same across different scoring algorithms and are illustrated in Table 1.

Ranked evaluation algorithms on the other hand take into account the score generated by the scoring algorithm and calculate the metric for the most relevant terms(ones with the highest score). The key idea behind these metrics is that the user is interested in the top k terms out of the complete term list returned by the filter (chunker in this case).

Table 1: Unranked Evaluation Results

Precision	Recall	F1
0.106	0.023	0.037

Table 2: Ranked Evaluation Results

Algorithm	MAP	MAR	MAF1
TF-IDF	0.079	0.016	0.031
RIDF	0.079	0.016	0.031
Domain Pertinence	0.091	0.018	0.037
Weirdness	0.091	0.018	0.037
Relevance	0.089	0.018	0.036

In this paper we have used k = 5 for evaluation of the metrics. This means that the metrics are evaluated for each of the top-5 terms in the list of retrieved term candidadates sorted in decreasing order of their scores. In cases where the total size of the term list returned by the filter is less than 5 then we have set k = length of the filtered term candidate list.

As a part of the experiments we have used 3 different ranked evaluation metrics namely Mean Average Precision (MAP), Mean Average Recall (MAR) and Mean Average F1-score (MAF1). MAP is the mean of Average Precision@k (AP@k given by Equation 8) over all the documents of the collection. Similarly, MAR is the mean of Average Recall@k (AR@k given by Equation 9) over all the documents of the corpus. MAF1 is the harmonic mean of AP@k and AR@k over all the documents in the collection.

$$Average\ Precision(k) = \sum_{i=1}^{k} \frac{\frac{TP@i}{TP@i+FP@i}}{k} \quad (8)$$

$$Average\ Recall(k) = \sum_{i=1}^{k} \frac{\frac{TP@i}{TP@i+FN@i}}{k} \quad (9)$$

where TP is the number of true positives, FP is the number of false positives and FN is the number of false negatives.

6. Results and Discussion

The results obtained are illustrated in Table 1 and Table 2. As can be seen the scores for all the algorithms are not very high. Furthermore, it can be observed that the values for precision are higher than the recall values for both ranked and unranked evaluation metrics. This is primarily due to the sub-optimal selection of term candidates for each document. As discussed previously the term candidates were filtered by first annotating the documents with the tnt tagger, followed by chunking performed using the RegexpParser (both tnt tagger and RegexpParser are available as a part of the NLTK package) to select the noun phrases as term candidates. However it was observed that tagger had limited capacity and a large number of chunks were annotated with *<UNK>* tokens (unknown tokens). As a result a very low number of term candidates (noun phrases) were filtered for each document which in turn brought down the recall scores leading to very high false negatives.

It is also interesting to note that the values for the ranked metrics is lower is than the values for the unranked metrics which indicates that scoring algorithms don't reflect the gold standard lists; they assign higher ranks to non-term entities. There are two possible reasons for this; firstly, the list of filtered term candidates is not representative of the gold standard and as result the performance is low irrespective of the rank assigned by the scoring algorithm; and secondly another reason could be that termhood in this domain is not reflected by frequency of occurrence. However, on closer inspection of the results we found that where an appropriate list of term candidates had been filtered out, the term scores were high and which in turn indicated that term candidates have high occurrence frequency in both the target as well as the background corpus thus ruling out the possibility of the second reason mentioned previously.

Furthermore, as illustrated in Table 2 algorithms belonging to the same class; frequency based approaches namely TF-IDF and RIDF exhibit similar performance and similarly background corpus based approaches namely Domain Pertinence, Weirdness and Relevance have similar performance. This is because of similar ranking patterns across a specific class of algorithms.

Also, it is interesting to note that background corpus based methods have a slightly better performance than the frequency based approaches, this is indicative of the positive influence of the background corpus on the task at hand.

7. Conclusion and Future Work

To conclude the dataset described here is the first term annotated dataset for Hindi. During evaluation of this dataset with unsupervised algorithms we observed that the score of frequency and background corpus based methods is not high. As discussed previously, this is primarily due to the suboptimal performance of tagger leading to inefficient selection of term candidates. Another important aspect is the search criteria for chunking, introduction of more complicated noun phrasal structures can improve performance of the term extractors.

Annotation for under-resourced languages is one of the most challenging problems in natural language processing (NLP). It is difficult to find trained expert annotators in order to ensure a high quality of annotation of the datasets. In this research, the dataset has been annotated by one annotator and we are aware that there can be bias in the dataset. However, the annotations provided here can serve as seed annotations for more sophisticated self-supervised and semi-supervised algorithms which we hope can then establish state-of-the-art benchmarks for under-resourced term extraction. Also, it is important to note that supervised learning algorithms are used to noisy annotate datasets in NLP, however terms are references to domain concepts and we are not aware of any machine learning algorithm that can essentially map concepts; it is one of the longstanding problems in the area of artificial intelligence and therefore the manually annotated dataset presented here better models the domain concepts of education.

Finally, this is an ongoing research and we hope to add more annotators as well as develop better annotation guidelines in order to improve the annotation quality of this dataset in the future. Furthermore, we also intend on adding more documents to the collection so that the dataset can be meaningfully used to train deep learning based architectures for term extraction. From an algorithmic perspective, we plan on the development of novel algorithms which can beat the current state-of-the-art on the task of term extraction for under-resourced languages.

8. Acknowledgement

Author Shubhanker Banerjee was supported by Science Foundation Ireland under Grant Agreement No. 13/RC/2106_P2 at the ADAPT SFI Research Centre at National University Of Ireland Galway. ADAPT, the SFI Research Centre for AI-Driven Digital Content Technology, is funded by Science Foundation Ireland through the SFI Research Centres Programme.

9. Bibliographical References

Astrakhantsev, N. (2018). ATR4S: Toolkit with State-of-the-Art Automatic Terms Recognition Methods in Scala. *Lang. Resour. Eval.*, 52(3):853–872, sep.

Bada, M., Eckert, M., Evans, D., Garcia, K., Shipley, K., Sitnikov, D., Baumgartner, W. A., Cohen, K. B., Verspoor, K., Blake, J. A., and Hunter, L. E. (2012). Concept annotation in the craft corpus. *BMC Bioinformatics*, 13(1):161, Jul.

Bordea, G., Buitelaar, P., and Polajnar, T. (2013). Domain-independent term extraction through domain modelling. In *The 10th international conference on terminology and artificial intelligence (TIA 2013), Paris, France*. 10th International Conference on Terminology and Artificial Intelligence.

Brants, T. (2000). TnT – a statistical part-of-speech tagger. In *Sixth Applied Natural Language Processing Conference*, pages 224–231, Seattle, Washington, USA, April. Association for Computational Linguistics.

Bright, W. (1996). The devanagari script. *The world's writing systems*, pages 384–390.

Cabré, M. T. (1999). *Terminology: Theory, methods, and applications*, volume 1. John Benjamins Publishing.

Church, K. and Gale, W., (1999). *Inverse Document Frequency (IDF): A Measure of Deviations from Poisson*, pages 283–295. Springer Netherlands, Dordrecht.

Cram, D. and Daille, B. (2016). Terminology extraction with term variant detection. In *Proceedings of ACL-2016 system demonstrations*, pages 13–18.

Daille, B. (1994). Study and implementation of combined techniques for automatic extraction of terminology. In *The balancing act: Combining symbolic and statistical approaches to language*.

Evans, D. A. and Lefferts, R. G. (1995). Clarit-trec experiments. *Information processing & management*, 31(3):385–395.

Jacquemin, C. and Bourigault, D. (2005). Term Extraction and Automatic Indexing.

Kachru, Y. (2006). *Hindi*, volume 12. John Benjamins Publishing.

Khurshid, A., Gillman, L., and Tostevin, L. (2000). Weirdness indexing for logical document extrapolation and retrieval. In *Proceedings of the Eighth Text Retrieval Conference (TREC-8)*.

Kim, J.-D., Ohta, T., Tateisi, Y., and Tsujii, J. (2003). GENIA corpus—a semantically anno-

tated corpus for bio-textmining. *Bioinformatics*, 19(suppl$_1$) : $i180 - -i182, 07$.

Manning, C., Raghavan, P., and Schütze, H. (2010). Introduction to information retrieval. *Natural Language Engineering*, 16(1):100–103.

McCrae, J. P. and Doyle, A. (2019). Adapting Term Recognition to an Under-Resourced Language: The Case of Irish. In *Proceedings of the Celtic Language Technology Workshop*, pages 48–57.

Meijer, K., Frasincar, F., and Hogenboom, F. (2014). A semantic approach for extracting domain taxonomies from text. *Decision Support Systems*, 62:78–93.

Oliver, A. (2017). A system for terminology extraction and translation equivalent detection in real time: Efficient use of statistical machine translation phrase tables. *Machine Translation*, 31(3):147–161.

Pazienza, M. T., Pennacchiotti, M., and Zanzotto, F. M. (2005). Terminology extraction: An analysis of linguistic and statistical approaches. In Spiros Sirmakessis, editor, *Knowledge Mining*, pages 255–279, Berlin, Heidelberg. Springer Berlin Heidelberg.

Pazienza, M. T. (1998). A domain-specific terminology-extraction system. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 5(2):183–201.

Peñas, A., Verdejo, F., and Gonzalo, J. (2001a). Corpus-based terminology extraction applied to information access.

Peñas, A., Verdejo, F., Gonzalo, J., et al. (2001b). Corpus-based terminology extraction applied to information access. In *Proceedings of corpus linguistics*, volume 2001, page 458.

Šajatović, A., Buljan, M., Šnajder, J., and Dalbelo Bašić, B. (2019). Evaluating automatic term extraction methods on individual documents. In *Proceedings of the Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019)*, pages 149–154, Florence, Italy, August. Association for Computational Linguistics.

Tan, C., Xia, J., Wu, L., and Li, S. Z. (2021). Co-learning: Learning from noisy labels with self-supervision. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1405–1413.

Zhang, Z., Gao, J., and Ciravegna, F. (2016). JATE 2.0: Java automatic term extraction with Apache Solr. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2262–2269, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Zhang, Z., Petrak, J., and Maynard, D. (2018). Adapted textrank for term extraction: A generic method of improving automatic term extraction algorithms. *Procedia Computer Science*, 137:102–108. Proceedings of the 14th International Conference on Semantic Systems 10th – 13th of September 2018 Vienna, Austria.

10. Language Resource References

Zhang, Ziqi and Gao, Jie and Ciravegna, Fabio. (2016). *JATE 2.0: Java Automatic Term Extraction with Apache Solr*. European Language Resources Association (ELRA).