Fine-grained algorithm for improving knn computational performance on clinical trials text class.pdf

by

Submission date: 28-Mar-2022 10:27AM (UTC+0700)

Submission ID: 1794633650

File name: Fine-grained algorithm for improving knn computational performance on clinical trials text class.pdf (1.84M)

Word count: 4365

Character count: 28419

big data and cognitive computing

MDPI

Article

Fine-Grained Algorithm for Improving KNN Computational Performance on Clinical Trials Text Classification

Jasmir Jasmir [1,2,*], Siti Nurmaini [3] and Bambang Tutuko [3]

[1] Doctoral Program of Engineering Science, Faculty of Engineering, Universitas Sriwijaya, Palembang 30128, Indonesia
[2] Computer Engineering, Universitas Dinamika Bangsa, Jambi 36138, Indonesia
[3] Intelligent System Research Group, Universitas Sriwijaya, Palembang 30128, Indonesia; sitinurmaini@gmail.com (S.N.); bambang_tutuko@unsri.ac.id (B.T.)
* Correspondence: ijay_jasmir@yahoo.com

Abstract: Text classification is an important component in many applications. Text classification has attracted the attention of researchers to continue to develop innovations and build new classification models that are sourced from clinical trial texts. In building classification models, many methods are used, including supervised learning. The purpose of this study is to improve the computational performance of one of the supervised learning methods, namely KNN, in building a clinical trial document text classification model by combining KNN and the fine-grained algorithm. This research contributed to increasing the computational performance of KNN from 388,274 s to 260,641 s in clinical trial texts on a clinical trial text dataset with a total of 1,000,000 data.

Keywords: text classification; clinical trials; supervised learning; KNN; fine-grained algorithm; improving computational performance

check for updates

Citation: Jasmir, J.; Nurmaini, S.; Tutuko, B. Fine-Grained Algorithm for Improving KNN Computational Performance on Clinical Trials Text Classification. *Big Data Cogn. Comput.* **2021**, *5*, 60. https://doi.org/10.3390/bdcc5040060

Academic Editor: Min Chen

Received: 28 August 2021
Accepted: 25 October 2021
Published: 28 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Text classification is defined as labeling natural-language text documents with classes or categories of predetermined sets [1]. Text classification is an important component in many NLP applications, such as sentiment analysis [2], relationship extraction [3], and spam detection [4,5]. Text classification has also attracted the attention of researchers to continue to develop innovations and testing, including those sourced from clinical texts, commonly referred to as clinical trials.

A clinical trial is a type of research that studies how safe it is to help test care given to patients [6]. Clinical trials play an important role in translating scientific research into the practice of medical outcomes [7]. In clinical trials, the most important part is called the eligibility criteria, which determine the cost, duration, and success of the clinical trial process [8].

Research on eligibility criteria in clinical trials is usually written in free text, but it is difficult if interpreted by a computer. A popular method of processing eligibility criteria is knowledge representation, which often requires extensive knowledge and hard work from experts in the sector of medical coding to identify eligibility criteria. In solving the problem of the feasibility analysis of clinical trials, the optimal methods are artificial intelligence, such as rule-based systems, traditional machine learning algorithms, and representation learning, such as deep learning architecture [9,10].

Deep learning technology [11] has achieved extraordinary results in many areas, such as computer vision [12], speech recognition [13], and text classification [14]. Menger [15] states that, in some cases, approaches with deep learning techniques applied to the classification of clinical texts can produce conclusions that match expectations but will be different if tested on other clinical datasets and with different domains and different sizes.

The problem raised by Menger above can be seen in the research of Bustos and Pertusa [16]. They conducted research on clinical trials. In this study, they trained, validated,

and compared various classification models, namely *k*-nearest neighbor (KNN), support vector machine (SVM), convolutional neural network (CNN), and FastText. This research utilized a dataset from a "clinical trial". The calculated values were precision, recall, F1, and Cohen's K. SVM produced the lowest accuracy results, and KNN obtained the highest accuracy performance similar to the CNN model, albeit with the lowest computational performance.

However, the value of computational performance can be increased by one of the methods discussed by Sutanto [17] in his paper on the fine-grained algorithm(FGA) approach, which utilizes the ability of search engines to handle big data efficiently. However, this paper only tested the problem of unsupervised learning clustering.

Based on the problems from Bustos and Pertusa above, we saw an opportunity to conduct further research, namely by increasing the computational value using the fine-grained algorithm, which was tested on the supervised learning method, especially KNN, which is the main contribution of this research.

In this study, we use two supervised learning classification methods, namely KNN and SVM, based on the supervised learning method used by Bustos and Pertusa [16].

K-nearest neighbor is a simple algorithm that stores all available cases and classifies a new case based on a similarity measure (e.g., distance functions) [18]. To classify an unknown document, the KNN algorithm identifies the *k*-nearest neighbors in a given document space. The KNN algorithm uses a similarity function such as Euclidean distance or cosine resemblance to obtain neighbors. The best option of selecting the grade of k depends on the dataset or application. The implementation of the KNN algorithm is very easy, but it is computationally intensive, especially as the size of the training documents grows.

A case is classified by majority vote of its neighbors, with the case being assigned to the class most common among its *K*-nearest neighbors measured by a distance function. If $K = 1$, then the case is simply assigned to the class of its nearest neighbor.

$$\text{Euclidean} = \sqrt{\sum_{i=1}^{k}(xi - yi)^2} \tag{1}$$

$$\text{Manhattan} = \sum_{i=1}^{k}|xi - yi| \tag{2}$$

Support vector machine is a relatively new classification method [19,20]. Although it is a complex algorithm, SVM reaches high classification levels in many areas. SVM is a two-class linear classifier. Among the likely hyperplanes between two classes, SVM obtains the optimal hyperplane between two classes by maximizing the margin among the closest points of classes. The points that prevaricate on the hyperplane boundaries are named support vectors.

For the linear kernel, the equation for the prediction of a new input using the dot product between the input (x) and each when two classes are not linearly partible, SVM projects data points into a higher-dimensional scope so that the data points become linearly partible by utilizing kernel techniques. There are several kernels that can be used for the SVM algorithm. Support vector (xi) is calculated as follows:

$$f(x) = B(0) + \sum(ai * (x, xi)) \tag{3}$$

The polynomial kernel can be written as

$$K(x, xi) = 1 + \sum(x * xi)^d \tag{4}$$

and the exponential as

$$K(x, xi) = \exp\left(-\gamma * \sum(x - x^2)\right). \tag{5}$$

2. Materials and Methods

The entire classification process is shown in Figure 1 below. Figure 1 shows the framework of the process of improving KNN computational performance using the fine-grained algorithm in clinical trial text classification.

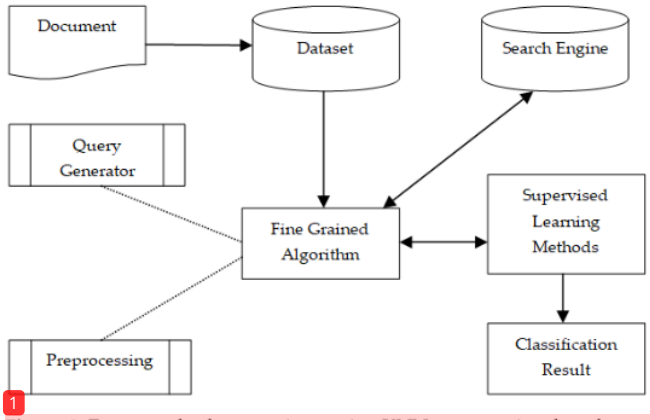

Figure 1. Framework of process improving KNN computational performance.

Dataset: Data were taken from clinical statements. A total of 6,186,572 data were extracted from 49,000 clinical trial protocols on cancer originating from the National Library of Medicine, National Institutes of Health (Bethesda, MD, USA) [21]. These data came from the fields of intervention, conditions, and feasibility, written in unstructured free-text language. Information on the eligibility criteria comprised a series of phrases and or sentences that were displayed in a free format, such as paragraphs, bulleted lists, enumeration lists, etc.

Preprocessing: Preprocessing plays a very important role in the technique and application of text mining. This is the first step in the process of mining text. In this paper, we discuss the three main steps of preprocessing, namely stop words, stemming, and TF/IDF [22]. All eligibility criteria were converted into a sequence of simple words. Information on study interventions and types of cancer was added to each feasibility criterion by separating the text into statements, then removing punctuation, white-space characters, all nonalphanumeric symbols, separators, and single-character words from the extracted text. All words were lowercase letters. We did not delete stop words "because", "like", "or", "and", and "for" because they are semantically relevant to clinical statements. We then changed numbers, arithmetic marks, and comparators to text.

The following stage is labeling or class. The available dataset has 6 million clinical statements containing a total of 148 million words. The vocabulary consists of 49,000 different words. In carrying out the labeling stage, we separated clinical statements into 2 classes: previously processed from eligibility criteria, study conditions, and interventions as "Eligible" (inclusion criteria) or "Not Eligible" (exclusion criteria):

- Their positions were in relation to the phrases "inclusion criteria" or "exclusion criteria", which usually preceded the respective lists. If those phrases were not found, then the statement was labeled "Eligible".
- Negation identification and transformation: negated inclusion criteria starting with "no" were transformed into positive statements and labeled "Not Eligible".

All other possible means of negating statements were expected to be handled intrinsically by the classifier, where Class = 0 is "Eligible", and Class = 1 is "Not Eligible". The division of the classes was unequal. Only 39% were labeled "Not Eligible", while 61% were labeled "Eligible". Since this dataset is quite large, we corrected for it using random

balanced under sampling, which resulted in a reduced dataset size with 4 million labeled samples. A snippet of clinical statements and classes can be seen in Figure 2 below.

Document	Class
voluntarily signed and dated written informed consent prior to any study specific procedure	0
unstable cardiac disease pulse_oximetry saturation less_than ninety at rest	1
renal dysfunction in the form of elevated serum creatinine	1
recovery to grade less_than one from any adverse event derived from previous treatment excluding	0
previous enrolment in the present study	1
patients receiving live vaccines within thirty days prior to the first dose of study therapy and while p	1
patient must have fully recovered from acute toxic effects of all prior chemotherapy immunotherapy	0
participation to study involving medical or therapeutic intervention in the last thirty days	1
men or women of childbearing potential who are not using an effective method of contraception wo	1
enrollment in any other clinical study from screening up to day one hundred unless pi judges such e	1
dental treatment anticipated after evaluation	1
central nervous system malignancy	1
both women and men must agree to use medically acceptable method of contraception throughout	0
age less than eighteen years or greater than sixty-five years	1
adequate haematological renal metabolic and hepatic function	0
active infection	1
unresectable or metastatic gct and grade iv gct	1
prior bisphosphonate usage except preoperative treatment with zoledronic acid up to three months	1
pregnancy or lactating	1
pathological fracture in gct	0

Figure 2. Snippet of clinical trial dataset.

Fine-grained algorithm (FGA): FGA presumes the turned form of the cluster hypothesis [23], that is, the relevant documents returned in response to a query will be inclined to be similar to one another. FGA uses a combination of loci and relevant cluster concepts to efficiently form clusters. The use of loci makes the computation of cluster representations efficient, since it only utilizes a small set of documents instead of all documents in a cluster. By using the relevant cluster concept, FGA does not require pairwise similarity comparisons between a document and all of its clusters. These strategies allow FGA to generate a fine-grained algorithm solution efficiently. Figure 3 below shows a snippet of part of the fine-grained algorithm.

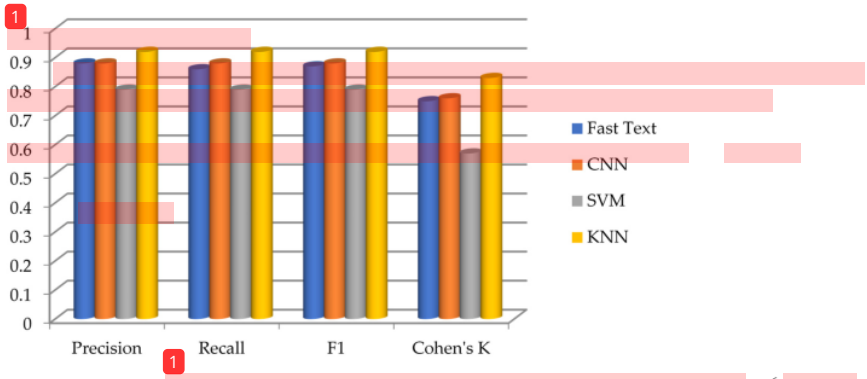

Figure 3. Graph of overall results of the validation set for all classifiers using a dataset of 10^6 samples.

3. Result and Discussion

Table 1 and Figure 3 below are the results of research conducted by Bust[illegible] and Pertusa, who carried out the precision, recall, F1 score, and Cohen's K process [16].

Table 1. Overall results of the validation set for all classifiers using a dataset of 10^6 samples.

Classifier	Evaluation			
	Precision	Recall	F1	Cohen's K
FastText	0.88	0.86	0.87	0.75
CNN	0.88	0.88	0.88	0.76
SVM	0.79	0.79	0.79	0.57
KNN	0.92	0.92	0.92	0.83

In this study, the evaluation model was built with two supervised learning methods, KNN and SVM, with a total of 1,000,000 data, the same amount used by Bustos et al. [16]. Some of these supervised learning methods were combined with the fine-grained algorithm. The specifications of the computer used areas follows: processor: Intel® Core™ i7-7700 CPU @ 3.60 GHz (8 CPUs), ~3.6 GHz; RAM: 16 GB; HDD: 1 TB; operating system: Windows 10 Pro 64-bit (10.0 Build 19,042), Python 3.6.6. The model evaluations carried out in this study were accuracy, precision, recall, and F1score.Table 2 and Figure 4 show the values of the accuracy of recall, precision, and F1score after going through the FGA stages, where the highest accuracy values of recall, precision, and F1score were generated by KNN, k = 5, which are 94.5, 93.8, 95.2 and 94.5,while the lowest values were generated by SVM, namely 79.1, 80.7, 82.2 and 81.4.

Table 2. Results of accuracy, precision, recall, and F1score after using FGA.

Evaluation	KNN	SVM
Accuracy	94.5	79.1
Precision	93.8	80.7
Recall	95.2	82.2
F1score	94.5	81.4

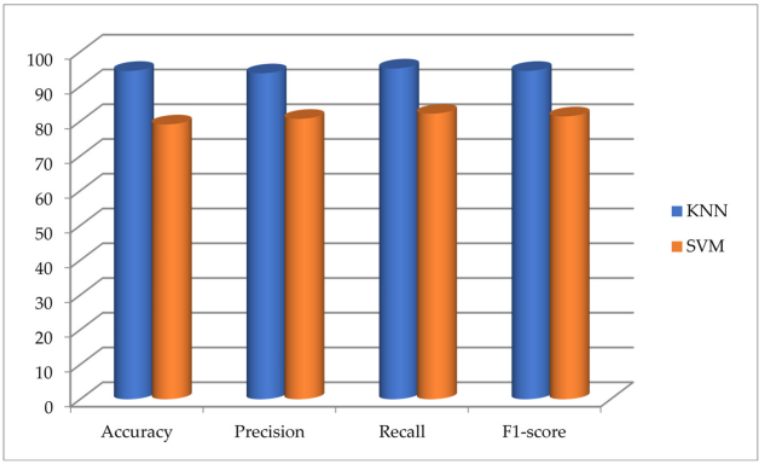

Figure 4. Graph of results of accuracy, precision, recall, and F1score.

The focus of this research was to improve the computational performance of KNN using FGA. In this research, a trial was conducted using the same data as Bustos et al. This study used 1 million data, and processing started from 100,000, 200,000, and 300,000 to 1,000,000 data. Evaluation models such as precision, recall, accuracy, and F1 score were built in this study. The method used was supervised learning, which was then compared by combining FGA features in the process. When conducting the evaluation process, we

also recorded the results of the computational performance for each supervised learning method used.

The first process that was carried out used the KNN method, which was then compared with the KNN process combined with FGA. As shown in Table 3 and Figure 5 below, the performance of KNN + FGA is faster, which is 260,641 s compared to KNN without FGA, which is 388,274 s.

Table 3. Table of computational performance of KNN and KNN using FGA.

No	Data Record	KNN Second	KNN + FGA Second
1	100,000	39,368	32,365
2	200,000	78,132	57,729
3	300,000	116,897	83,093
4	400,000	155,663	108,457
5	500,000	194,427	133,821
6	600,000	233,194	159,185
7	700,000	271,962	184,549
8	800,000	310,731	209,913
9	900,000	349,502	235,277
10	1,000,000	388,274	260,641

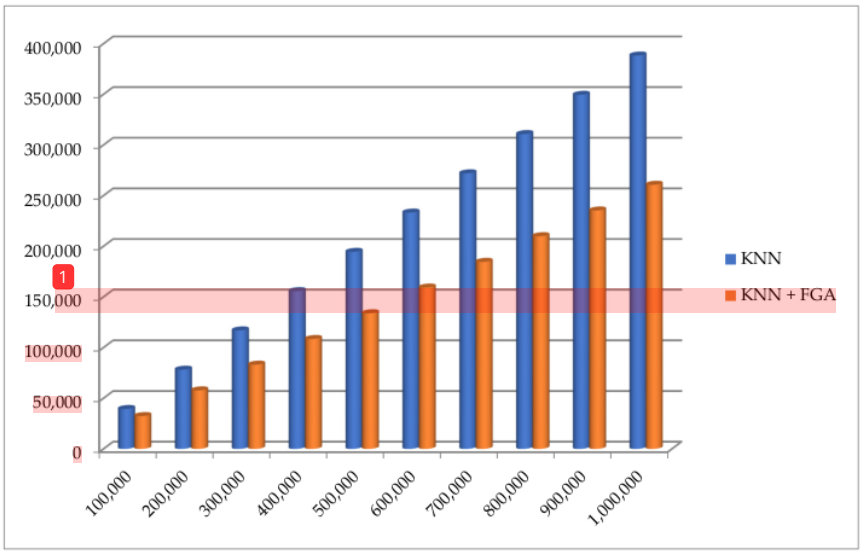

Figure 5. Graph of computational performance of KNN and KNN using FGA.

The following experiment was to build a second supervised learning model, namely a support vector machine. This was because previous research conducted by Bustos and Pertusa also used this model in their experiments. In building this model, the results obtained were that FGA was not able to improve the computational performance of SVM. The SVM structure is more complex than other supervised learning methods and cannot be influenced by FGA. In classification modeling, SVM already has a more mature and mathematically clear concept compared to other classification techniques [24].

As shown in Table 4 and Figure 6 below, for both the smallest and largest datasets, combining FGA and SVM was not able to help accelerate computing performance. One of the biggest pieces of data, SVM computing without FGA, is faster, namely 363,587 s, than SVM using FGA, which is 389,768 s.

Table 4. Table of computational performance of SVM and SVM using FGA.

No	Data Record	SVM Second	SVM + FGA Second
1	100,000	35,294	40,244
2	200,000	71,771	79,080
3	300,000	108,248	117,916
4	400,000	144,725	156,752
5	500,000	181,202	195,588
6	600,000	217,679	234,424
7	700,000	254,156	273,260
8	800,000	290,633	312,096
9	900,000	327,110	350,932
10	1,000,000	363,587	389,768

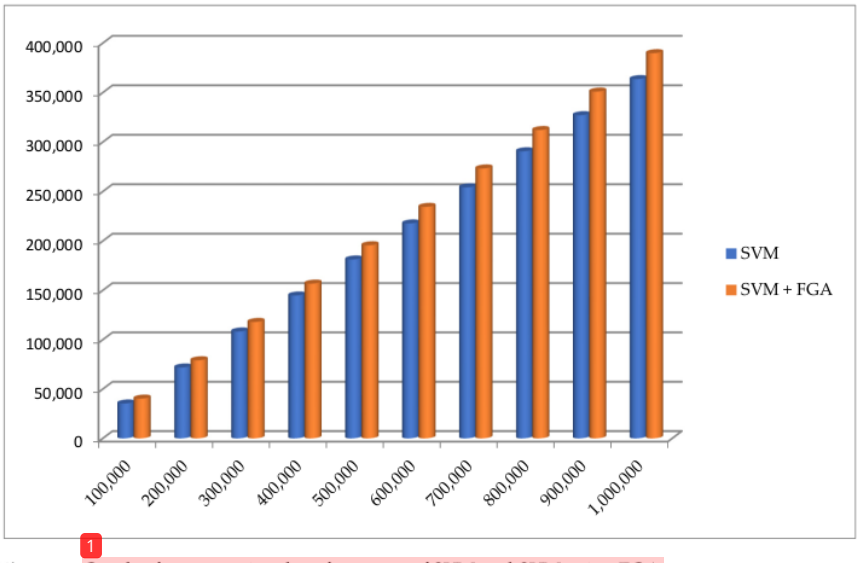

Figure 6. Graph of computational performance of SVM and SVM using FGA.

In accordance with the objectives and contributions of this study, the combination of KNN and FGA was proven to increase computational time, but for other supervised learning methods, this was not proven. However, the concept of combining FGA with KNN opens the potential to explore more ambitious goals by making the additional effort required to build a suitable dataset.

FGA can help and influence KNN in increasing computational performance. This is due to the flexible structure of KNN. If k is small, this flexibility causes KNN to tend to be sensitive to outliers, especially outliers located in the middle of the class. Furthermore, FGA can also affect the increase in KNN computational performance because KNN tends not to implicitly handle missing values [25,26].

4. Conclusions and Future Work

In this study, several classification methods were trained, validated, and compared in a collection of cancer clinical trial protocols (clinicaltrials.gov or www.kaggle.com accessed on 31 July 2021). The effect of the fine-grained algorithm on KNN to increase computational time was successfully carried out but failed to affect other supervised learning methods such as SVM. For KNN, the computational performance increased from 388,274 s to 260,641 for the largest dataset with 1 million data. However, FGA could not help SVM increase the value of computing performance. On the contrary, with the collaboration of FGA and

SVM, the computational performance decreased, from 363,587s to 389,768s, in the largest dataset. FGA was not able to help SVM increase the value of computing performance. This is because the SVM structure is more complex than other supervised learning methods and cannot be influenced by FGA. In classification modeling, SVM already has a more mature and mathematically clear concept compared to other classification techniques. The difference in computing time on the largest dataset reached 127.633 s using the following computer specifications: processor: Intel® Core™ i7-7700 CPU @ 3.60 GHz (8 CPUs), ~3.6 GHz; RAM: 16 GB; HDD,1 TB; operating system: Windows 10 Pro 64-bit (10.0 Build 19042), Python 3.6.6.

Further research can (1) try the combination of this fine-grained algorithm with other supervised learning methods on other text datasets that are larger in number, (2) apply and compare other supervised learning methods that were not used in this study, and (3) add relevant features to provide better computational value.

Author Contributions: Conceptualization, J.J. and S.N.; methodology, S.N. and B.T.; software, J.J.; validation, J.J., S.N. and B.T.; formal analysis, J.J. and S.N.; investigation, J.J.; resources, J.J.; data curation, J.J.; writing—original draft preparation, J.J.; writing—review and editing, J.J., S.N. and B.T.; visualization, J.J.; supervision, S.N. and B.T.; project administration, J.J.; funding acquisition, J.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset utilized in this study is freely available and can be downloaded from: https://www.kaggle.com/auriml/eligibilityforcancerclinicaltrials (accessed on 31 July 2021).

Acknowledgments: This research was funded by Yayasan Dinamika Bangsa Jambi Indonesia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Toruno, D.; Çak, E.; Ganiz, M.C.; Akyoku, S.; Gürbüz, M.Z. Analysis of Preprocessing Methods on Classification of Turkish Texts. In Proceedings of the 2011 International Symposium on Innovations in Intelligent Systems and Applications, Istanbul, Turkey, 15–18 June 2011; pp. 112–117.
2. Socher, R.; Perelygin, A.; Wu, J.; Chuang, J.; Mnaning, C.D.; Ng, A.; Potts, C. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, Washington, DC, USA, 18–21 October 2013; pp. 1631–1642.
3. Zeng, D.; Liu, K.; Chen, Y.; Zhao, J. Distant Supervision for Relation Extraction via Piecewise Convolutional Neural Networks. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 1753–1762.
4. Wang, A.H. DON'T FOLLOW ME–Spam Detection in Twitter. In Proceedings of the 10th International Conference on Security and Cryptography, Amalfi, Italy, 26 July 2010; pp. 142–151.
5. Xie, S.; Wang, G.; Lin, S.; Yu, P.S. Review Spam Detection via Temporal Pattern Discovery. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining–KDD '12, Beijing, China, 12–16 August 2012; pp. 823–831.
6. Melinda, B. *Clinical Trials*; Bill & Melinda Gates Foundation: Seattle, WA, USA, 2014.
7. Shivade, C.; Hebert, C.; Lopetegui, M.; De Marneffe, M.-C.; Fosler-Lussier, E.; Lai, A.M. Textual inference for eligibility criteria resolution in clinical trials. *J. Biomed. Inform.* **2015**, *58*, S211–S218. [CrossRef] [PubMed]
8. Chondrogiannis, E.; Andronikou, V.; Tagaris, A.; Karanastasis, E.; Varvarigou, T.; Tsuji, M. A novel semantic representation for eligibility criteria in clinical trials. *J. Biomed. Inform.* **2017**, *69*, 10–23. [CrossRef] [PubMed]
9. Mackellar, B.; Schweikert, C. Analyzing conflicts between Clinical Trials from a patient perspective. In Proceedings of the 17th International Conference on E-health Networking, Application & Services (HealthCom) 2015, Boston, MA, USA, 14–17 October 2015; pp. 479–482.
10. Mackellar, B.; Schweikert, C. Patterns for conflict identification in clinical trial eligibility criteria. In Proceedings of the IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom), Munich, Germany, 14–17 September 2016; pp. 1–6.
11. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [CrossRef] [PubMed]

12. Campos, V.; Jou, B.; Giró-I-Nieto, X. From pixels to sentiment: Fine-tuning CNNs for visual sentiment prediction. *Image Vis. Comput.* **2017**, *65*, 15–22. [CrossRef]
13. Brocki, Ł.; Marasek, K. Deep Belief Neural Networks and Bidirectional Long-Short Term Memory Hybrid for Speech Recognition. *Arch. Acoust.* **2015**, *40*, 191–195. [CrossRef]
14. Tai, K.S.; Socher, R.; Manning, C.D. Improved Semantic Representations from Tree-Structured Long Short-Term Memory Networks. *arXiv* **2015**, arXiv:1503.00075.
15. Menger, V.; Scheepers, F.; Spruit, M. Comparing Deep Learning and Classical Machine Learning Approaches for Predicting Inpatient Violence Incidents from Clinical Text. *Appl. Sci.* **2018**, *8*, 981. [CrossRef]
16. Bustos, A.; Pertusa, A. Learning Eligibility in Cancer Clinical Trials Using Deep Neural Networks. *Appl. Sci.* **2018**, *8*, 1206. [CrossRef]
17. Sutanto, T.; Nayak, R. Fine-grained document clustering via ranking and its application to social media analytics. *Soc. Netw. Anal. Min.* **2018**, *8*, 29. [CrossRef]
18. Isa, D.; Lee, L.H.; Kallimani, V.P.; Rajkumar, R. Text Document Preprocessing with the Bayes Formula for Classification Using the Support Vector Machine. *IEEE Trans. Knowl. Data Eng.* **2008**, *20*, 1264–1272. [CrossRef]
19. Ramesh, B.; Sathiaseelan, J. An Advanced Multi Class Instance Selection based Support Vector Machine for Text Classification. *Procedia Comput. Sci.* **2015**, *57*, 1124–1130. [CrossRef]
20. Husni, N.L.; Handayani, A.S.; Nurmaini, S.; Yani, I. Odor classification using Support Vector Machine. In Proceedings of theInternational Conference on Electrical Engineering and Computer Science (ICECOS), Palembang, Indonesia, 22–23 August 2017; pp. 71–76.
21. National Library of Medicine; National Institutes of Health. *XML Schema for ClinicalTrials.gov Public XML*; National Library of Medicine, National Institutes of Health: Bethesda, MD, USA, 2017.
22. Liu, C.-Z.; Sheng, Y.-X.; Wei, Z.-Q.; Yang, Y.-Q. Research of Text Classification Based on Improved TF-IDF Algorithm. In Proceedings of the IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE), Lanzhou, China, 24–27 August 2018; pp. 218–222.
23. Fuhr, N.; Lechtenfeld, M.; Stein, B.; Gollub, T. The optimum clustering framework: Implementing the cluster hypothesis. *Inf. Retr.* **2011**, *15*, 93–115. [CrossRef]
24. Pratama, B.Y.; Sarno, R. Personality classification based on Twitter text using Naive Bayes, KNN and SVM. In Proceedings of the International Conference on Data and Software Engineering (ICoDSE), Yogyakarta, Indonesia, 25–26 November 2015; pp. 170–174.
25. Tan, Y. An Improved KNN Text Classification Algorithm Based on K-Medoids and Rough Set. In *2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*; IEEE: New York, NY, USA, 2018; Volume 1, pp. 109–113.
26. Lin, Y.; Wang, J. Research on text classification based on SVM-KNN. In Proceedings of the IEEE 5th International Conference on Software Engineering and Service Science, Beijing, China, 27–29 June 2014; pp. 842–844.

Fine-grained algorithm for improving knn computational performance on clinical trials text class.pdf

ORIGINALITY REPORT

71%	71%	19%	16%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	mdpi-res.com Internet Source	53%
2	Www.mdpi.com Internet Source	17%
3	Submitted to Sriwijaya University Student Paper	1%
4	www.mdpi.com Internet Source	<1%
5	medium.com Internet Source	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		