# Effective 20 Newsgroups Dataset Cleaning

Khaled Albishre*[x], Mubarak Albathan*[§], Yuefeng Li*
*Science and Engineering Faculty
Queensland University of Technology (QUT)
Brisbane, Australia
[x]Umm Al-Qura University
Saudi Arabia, Makkah
[§]Al Imam Mohammad Ibn Saud Islamic University
Saudi Arabia, P.O.Box 5701, Riyadh 11432
Email: khaledalbishre@gmail.com, mubarak.albathan@hdr.qut.edu.au, y2.li@qut.edu.au

***Abstract*—The rapid increase in the number of text documents available on the Internet has created pressure to use effective cleaning techniques. Cleaning techniques are needed for converting these documents to structured documents. Text cleaning techniques are one of the key mechanisms in typical text mining application frameworks. In this paper, we explore the role of text cleaning in the 20 newsgroups dataset, and report on experimental results.**

## I. Introduction

While a variety of text mining applications seek to achieve highly-accurate results, different cleaning methods should be considered to achieve this effectively and efficiently. The objective is to derive or concentrate organized representation from unstructured plain textual data. Such representations are then suitable for a particular reasoning and algorithm. Cleaning techniques [1], [2], [3] include white space evacuation, stop words removal, case collapsing, stemming, spelling error correction, truncation extension, and negative handling. Natural language processing procedures such as tokenization, grammatical-feature labeling (part-of-speech) and linguistic are a subset of these proceedings obliging the learning of the language to be handled [5].

The dataset is a principle part of any experiment or elevation for a proposed techniques or methods. To date, there are many benchmark datasets such as 20 Newsgroups [4], RCV1 [5], WebKB [6] available for research. The 20 Newsgroups dataset, often called 20NG, is widely used in text mining, information retrieval and machine learning research. In text mining, there are various applications which used 20NG in the literature, for example, classification [7], [8], [9], [10], clustering [11], [12], [13], filtering[14] and sentiment analysis [15]. Also, feature selection research papers have often used 20NG [16], [17], [18], [19], [20].

Text cleaning requires attention to numerous issues. Newswire text corpora frequently contains misspelled words, incorrect punctuation, erratic spacing and other irregularity features. Punctuation generally compares to the use of phonemes features in spoken language; to depend on all around framed sentences delimited by unsurprising punctuation can be exceptionally dangerous. In some corpora, conventional prescriptive standards are regularly overlooked.

The main purpose of document cleaning is to decrease the dimensionality to control the number of terms in the document [21]. Moreover, the cleaning stage will improve the performance and efficiency by making the data uniform. Different cleaning methods [2] widely used in text mining are stop word removal, ignore the short terms, special character removal, and stemming.

Stop words typically point to the most widely-recognized words in a language. One of the challenges in natural language processing (NLP) is that no universal stop word list exists. Every language has its own list, and each may change over time. There is more than one list for English, because the way people communicate has changed. For instance, English language has more than one list, because the communication behaviour between people has changed over the years.

Words such as ”is”, ”which”, ”the”, ”and”, ”of” make text data noisy, and that will reduce the efficient of text data documents. Therefore, stop words removal play an important role for document pre-processing steps for several reasons. The first significant aspect is that reducing the noisy for the text document and maintain the core term in document makes the processing more efficient and effective.

Stemming is another effective technique. The main purpose is to cut words down to their root. For example, in English, the words ”smoker” and ”smoking” have the same root stem ”smoke”. The literature contains different algorithms for this method. Porter stemming [22], [23] algorithm is a popular technique. We used the GERS model [24] to evaluate the cleaning algorithms for the first time. We choose GERS model because it can integrate both terms and *n*-grams.

The rest of our paper is organized as follows. In the next section, we will present the 20 Newsgroups dataset format. Section III will present the methods we used for preparing the data. In Section IV, we will evaluate the dataset and then, in Section V, we state our conclusion.

## II. Data Format in 20 Newsgroups

The 20 Newsgroups dataset is commonly used for text mining applications. It was collected by Ken Lang [4]. The 20 Newsgroups data set is a test collection of approximately 20,000 newsgroups documents that 1000 documents were taken from each of the newsgroups. It is divided across 20 different newsgroups. The category topics are related to computers, politics, religion, sports, and science. Each document belongs to exactly one newsgroup, but there is a small fraction

TABLE I. 20 NEWSGRROUPS CATEGORY

| | | |
|---|---|---|
| comp.graphics<br>comp.os.ms−windows.misc<br>comp.sys.ibm.pc.hardware<br>comp.sys.mac.hardware<br>comp.windows.x | rec.autos<br>rec.motorcycles<br>rec.sport.baseball<br>rec.sport.hockey | sci.crypt<br>sci.electronics<br>sci.med<br>sci.space |
| misc.forsale | talk.politics.misc<br>talk.politics.guns<br>talk.politics.mideast | talk.religion.misc<br>alt.atheism<br>soc.religion.christian |

TABLE II. DOCUMENTS IN 20 NEWSGROUPS

| Topic | # documents in testing | # documents in training | # documents |
|---|---|---|---|
| alt.atheism | 319 | 480 | 799 |
| comp.graphics | 389 | 584 | 973 |
| comp.os.ms-windows.misc | 394 | 591 | 985 |
| comp.sys.ibm.pc.hardware | 392 | 590 | 982 |
| comp.sys.mac.hardware | 385 | 578 | 963 |
| comp.windows.x | 395 | 593 | 988 |
| misc.forsale | 390 | 585 | 975 |
| rec.autos | 396 | 594 | 990 |
| rec.motorcycles | 398 | 598 | 996 |
| rec.sport.baseball | 397 | 597 | 994 |
| rec.sport.hockey | 399 | 600 | 999 |
| sci.crypt | 396 | 595 | 991 |
| sci.electronics | 393 | 591 | 984 |
| sci.med | 396 | 594 | 990 |
| sci.space | 394 | 593 | 987 |
| soc.religion.christian | 398 | 599 | 997 |
| talk.politics.misc | 310 | 465 | 775 |
| talk.politics.guns | 364 | 546 | 910 |
| talk.politics.mideast | 376 | 564 | 940 |
| talk.religion.misc | 251 | 377 | 628 |
| **Total** | **7532** | **11314** | **18846** |

```
From: ldo@waikato.ac.nz (Lawrence D'Oliveiro, Waikato
University)
Subject: Re: Spigot on LC III
Organization: University of Waikato, Hamilton, New Zealand
Lines: 18

In article <1993Apr15.214724.10871@aristo.tau.ac.il>,
isaaci@ccsg.tau.ac.il (barash isaac) writes:
> A friend of mine has problems running Spigot LC on an LC
III.
> His configuration is:
>
> Spigot LC / LC III, System 7.1
> Video Spigot Extension 1.0
>
> I would appriciate if I can get any postitive/negative
experience with this
> setup.

Somebody in comp.multimedia was also having trouble using a
Spigot in his
LC III. It turned out he needed the latest version of
ScreenPlay (1.1.1),
which fixed things.

Lawrence D'Oliveiro                     fone: +64-7-856-2889
Computer Services Dept                   fax: +64-7-838-4066
University of Waikato            electric mail:
ldo@waikato.ac.nz
Hamilton, New Zealand    37^ 47' 26" S, 175^ 19' 7" E, GMT+
12:00
```

Fig. 1. Sample Document

of the articles belong to more than one category. The data collection is the well-known 20-Newsgroups (20NG) dataset. The categories of the dataset are shown in Table I. Some newsgroups, for example, the category *comp.sys.ibm.pc.hardware* and *comp.sys.mac.hardware* are very similar to each other. An example of the 20 newsgroups dataset document shown in Fig1. From the example document, it contains more than one headers such as subject and from. Subject header holds the title of document and from header holds the email address for the sender.

Moreover, different versions[1] are available to use. In the first version, the original dataset contained 19997 documents. The second version that shown in Table II contains 18846 documents after removing some headers. In this version which called ”bydate”, they organize the data into 60% for the training and 40% for the testing. The last version that contains 18828 documents, they removed the duplicated and just remain only the ”From” and ”Subject” header.

## III. APPLIED METHODS FOR CLEANING

In this phase, the purpose is to make the unstructured document more persistent to promote text representation. There are different approaches using with text cleaning such as stop word removal and stemming. Stop word removal or stopping word aims to eliminate extremely common words by using a stop word list that has partial predictive worth for classification. In stemming, selected words are reduced to their word stem such as the word ”waiting”, ”waits”, and ”waited” would all to be reduced to single feature which is ”wait”.

[1]http://qwone.com/ jason/20Newsgroups/

Since the 20 newsgroups dataset contains only e-mail documents, the main issue we need to manage is the weight connected to headers. The header contains a ”subject” and a ”from” field. The ”from” field contains the sender’s name or email address (or both), along these lines no data about the substance; hence, we disregard it. Weight is connected only to the expressions of the ”subject” field.

The first step that we used was to remove special characters such as ”#”, ”@”, and ”/” and irrelevant information such as email addresses and numbers. In this stage, we used Java to read all the dataset. Then, we used tokenization the text to read words and check whether these were relevant or irrelevant term. We ignored the subject header, email addresses, numbers, and punctuation. We recognized the size of the dataset was reduced to 32.5 MB (uncompressed folder).

The next step that we applied was the removal of stop words. Stop words are fundamentally a situated of generally utilized words as a part of any dialect, not just English. The motivation behind why using stop words are basic to numerous applications is that, on the off chance that we uproot the words that are generally utilized as a part of a given dialect, we can concentrate on the imperative words. Stop-word lists contain function words such as articles, pronouns, conjunctions and other non-informative terms that we applied at this stage. A common stop word list that we used in our cleaning techniques is in [25]. We also mention the size of the dataset was reduced to 23.4 MB (uncompressed folder).

The third step was a word stemmer. In this stage, we applied Porter algorithm [22] which is one of the popular stemmer algorithm using in text mining research. The Porter

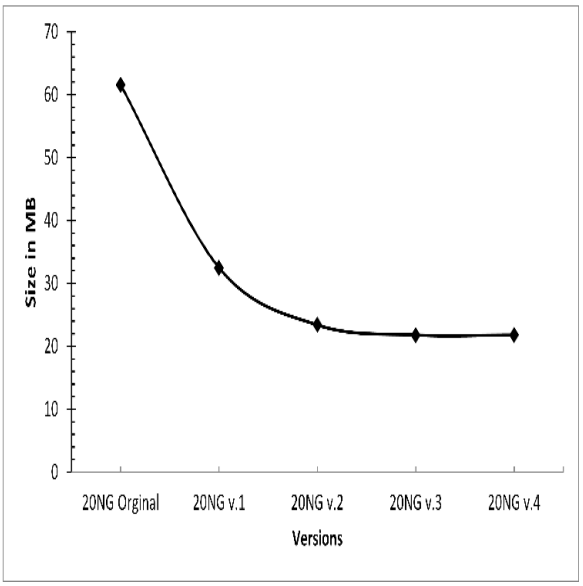


Fig. 2. Comparison the datasets size after using text cleaning techniques

algorithm is very fast and can handle removal of double letters in words such as successful. It has five stages; during each step, the rules are reviewed until the word passes all conditions. If a rule is recognized, the prefix or suffix is deleted, and the next step is performed, until finally the stemmed word is returned. The Porter stemming algorithm is available online in Java[2].

Finally, we eliminated 111 empty documents after we finished the text prepossessing technicians. Fig 2 shows the size of the dataset how decreased from 61.6 to 21.8 MB. The documents totalled 18,734, with 7,467 in testing and 11,267 in training.

## IV. Evaluation

In this section, we will demonstrate the examination environment and describe the experiment results and the discussions. The principal model applied in to conducted the results was the known method which is GERS.

In order to conduct the experiment, we used a popular dataset which is 20 Newsgroups that we are describe it in Section II.

GERS model [24] is a method to extract *n*-gram from text documents. This method consists of two main stages: first, selecting the best low-level terms, which attempts to improve the quality of extracting *n*-grams and thereby reduce the computational complexity and noisy features. Then these selected features are used to extract high-level features (*n*-grams) using Extended Random Set theory (ERS) to reweigh the extracted *n*-grams by calculating the probability of extracted high-level features and their low-level contents. Using the ERS theory improves the system performance significantly compared with other feature selection methods [26].

Before applying our technique, distinctive operations that we showed in Section III have been led on the 20 newsgroups dataset, for example, cleaning the data and eliminating a stopwords list.

[2]http://tartarus.org/martin/PorterStemmer/java.txt

### A. Measurements

The next measures are some broadly acknowledged and settled assessment measurements. The evaluation metrics which are the average precision of the *top-20* return documents, the *break-even point (b/p)* , *interpolated Precision on 11-points*, Mean Average Precision (MAP), and *F-scores* the $F_1$-*score* measure are extensively used in information retrieval and text mining research. As shown below, every measurement emphases on an alternate part of the framework execution.

Precision $p$ takes all retrieved documents into account. It can also be evaluated at a given cut-off rank, considering only the top-most results returned by the system. This measure is also called *top-n precision*. In this paper, we used the *top-20*. Another metric is the *break-even point (b/p)* which is used when the *precision* and *recall* are equal. In addition, Mean Average Precision (MAP) is figured by measuring the *precision* of each relevant document then averaging the precision over all the subjects. It consolidates precision, relevance ranking and general review together to measure the nature of the retrieval engines. Moreover, the Interpolated Average Precision (IAP) was computed by measuring the precision relative to the *recall*.

The final measure is that the *F-measure* or often called $F_1$-*score* is the harmonic mean of *Recall* and *Precision*. It is a measure of a test's accuracy. The metric is computed as follows:

$$F_1 = \frac{2pr}{p+r} \tag{1}$$

### B. Results

This section shows the results of the GERS method employing the 20 Newsgroups dataset. Table III summarize the results gained when GERS method is applied to the 20 Newsgroups dataset versions, respectively. In addition, the results of the performances are shown in Fig 3.

Table III present a comparison result between the 20NG versions where *v.1* the dataset after removing irrelevant information, *v.2* after applied stop word removal, *v.3* after applied Stemmer algorithm, and *v.4* after removing empty documents. The results in the table show that the 20NG versions performances are still keep the same as the original version. The versions results that have applied text cleaning were very near the original versions' result.

Fig 2 shows the size of the dataset versions how were reduced after applied text cleaning techniques. The size of the original dataset was 61.6 MB. In *v.1*, the size of the dataset reduced to 32.5 MB, about 50%. In addition, *v.2* decreased to 23.4 MB, and the *v.3* size reduced to 21.8 MB as well as the *v.4*. Therefore, irrelevant information removal and the stop word removal are effective techniques to reduce the size of the corps.

TABLE III. COMPARISON OF THE 20 NEWSGROUPS VERSIONS DEPEND ON GERS METHOD

| Dataset | *top-20* | *b/p* | *MAP* | *F_score* | *IAP* |
|---|---|---|---|---|---|
| **20NG Orginal** | 0.513 | 0.513 | 0.517 | 0.509 | 0.555 |
| **20NG v.1** | 0.513 | 0.504 | 0.513 | 0.508 | 0.549 |
| **20NG v.2** | 0.513 | 0.504 | 0.513 | 0.508 | 0.549 |
| **20NG v.3** | 0.510 | 0.505 | 0.513 | 0.508 | 0.548 |
| **20NG v.4** | 0.510 | 0.503 | 0.513 | 0.508 | 0.548 |

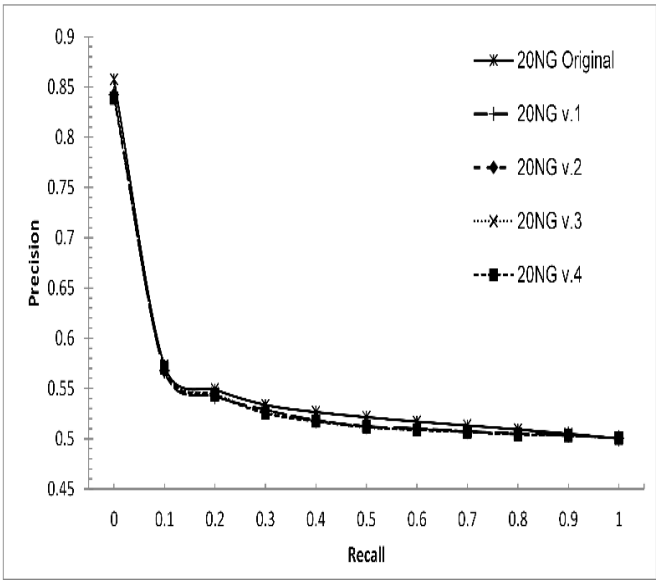


Fig. 3. Comparison results of GERS model on 20NG

## V. CONCLUSION

This paper offerings methods for text cleaning 20 Newsgroups dataset for reorganized the document in structure way. The first stage is to remove irrelevant information from the dataset. The second step is to eliminate the stop word list. The third phase is to convert the words to the root by applying stemmer algorithm. Finally, it is to exclude the empty documents from the dataset.

We conducted experiments the results show that text cleaning techniques have significantly reduced the dataset size. In addition, the 20 Newsgroups dataset versions were tested via a model which is GERS. The experimental results show that the GERS method is still slightly effects.

## REFERENCES


[1] G. Carvalho, D. M. de Matos, and V. Rocio, "Document retrieval for question answering: a quantitative evaluation of text preprocessing," in *Proceedings of the ACM first Ph. D. workshop in CIKM*. ACM, 2007, pp. 125–130.

[2] R. Feldman and J. Sanger, *The text mining handbook: advanced approaches in analyzing unstructured data*. Cambridge University Press, 2007.

[3] E. Clark and K. Araki, "Text normalization in social media: progress, problems and applications for a pre-processing system of casual english," *Procedia-Social and Behavioral Sciences*, vol. 27, pp. 2–11, 2011.

[4] K. Lang, "Newsweeder: Learning to filter netnews," in *Proceedings of the 12th international conference on machine learning*, 1995, pp. 331–339.

[5] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li, "Rcv1: A new benchmark collection for text categorization research," *The Journal of Machine Learning Research*, vol. 5, pp. 361–397, 2004.

[6] M. Craven, A. McCallum, D. PiPasquo, T. Mitchell, and D. Freitag, "Learning to extract symbolic knowledge from the world wide web," DTIC Document, Tech. Rep., 1998.

[7] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *The Journal of Machine Learning Research*, vol. 2, pp. 45–66, 2002.

[8] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using em," *Machine learning*, vol. 39, no. 2-3, pp. 103–134, 2000.

[9] A. Dasgupta, P. Drineas, B. Harb, V. Josifovski, and M. W. Mahoney, "Feature selection methods for text classification," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2007, pp. 230–239.

[10] D. Zhang, X. Chen, and W. S. Lee, "Text classification with kernels on the multinomial manifold," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2005, pp. 266–273.

[11] Y.-W. Seo and K. Sycara, "Text clustering for topic detection," 2004.

[12] L. AlSumait and C. Domeniconi, "Text clustering with local semantic kernels," in *Survey of Text Mining II*. Springer, 2008, pp. 87–105.

[13] X. Ji and W. Xu, "Document clustering with prior knowledge," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2006, pp. 405–412.

[14] C. Lanquillon, "Enhancing text classification to improve information filtering," Ph.D. dissertation, Otto-von-Guericke-Universität Magdeburg, Universitätsbibliothek, 2001.

[15] C.-L. Liu, W.-H. Hsaio, C.-H. Lee, G.-C. Lu, and E. Jou, "Movie rating and review summarization in mobile environment," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 42, no. 3, pp. 397–407, 2012.

[16] W. Zong, F. Wu, L.-K. Chu, and D. Sculli, "A discriminative and semantic feature selection method for text categorization," *International Journal of Production Economics*, vol. 165, pp. 215–222, 2015.

[17] S. Li, R. Xia, C. Zong, and C.-R. Huang, "A framework of feature selection methods for text categorization," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*. Association for Computational Linguistics, 2009, pp. 692–700.

[18] A. K. Uysal and S. Gunal, "A novel probabilistic feature selection method for text classification," *Knowledge-Based Systems*, vol. 36, pp. 226–235, 2012.

[19] H. Liu and H. Motoda, *Computational methods of feature selection*. CRC Press, 2007.

[20] C. Shang, M. Li, S. Feng, Q. Jiang, and J. Fan, "Feature selection via maximizing global information gain for text classification," *Knowledge-Based Systems*, vol. 54, pp. 298–309, 2013.

[21] Y. Li, A. Algarni, M. Albathan, Y. Shen, and M. A. Bijaksana, "Relevance feature discovery for text mining," *IEEE Transactions on Knowledge & Data Engineering*, no. 1, pp. 1–1, 2015.

[22] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.

[23] P. Willett, "The porter stemming algorithm: then and now," *Program*, vol. 40, no. 3, pp. 219–223, 2006.

[24] M. Albathan, Y. Li, and A. Algarni, "Enhanced n-gram extraction using relevance feature discovery," in *AI 2013: Advances in Artificial Intelligence*. Springer, 2013, pp. 453–465.

[25] B. Liu, *Web data mining: exploring hyperlinks, contents, and usage data*. Springer Science & Business Media, 2007.

[26] M. Albathan, Y. Li, and Y. Xu, "Using extended random set to find specific patterns," in *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014 IEEE/WIC/ACM International Joint Conferences on*, vol. 2. IEEE, 2014, pp. 30–37.