JOURNAL OF LINGUISTICS, CULTURE AND COMMUNICATION
Vol.02, No.02, 2024: December: 204-224, E-ISSN:2988-1641
https://jolcc.org/index.php/jolcc/index

Exploring The Impact of Stemming on Text Topic-Based Classification Accuracy

Refat Aljumily
Independent researcher
Email: r_alkind@yahoo.com

Submission Track:
Received: 13-04-2024, Final Revision: 28-06-2024, Available Online: 30-06-2024

ABSTRACT
Text classification attempts to assign written texts to specific group types that share the same linguistic features. One class of features that have been widely employed for a wide range of classification tasks is lexical features. This study explores the impact of stemming on text classification using lexical features. To explore, this study is based on a corpus of thirty texts written by six authors with topics that focus on politics, history, science, prose, sport, and food. These texts are stemmed using a light stemming algorithm. In order to classify these texts according to the topic by means of lexical features, linear hierarchical clustering and non-linear clustering (SOM) is carried out on the stemmed and unstemmed texts. Although both clustering methods are able to classify texts by topic with two models produce accurate and stable results, the results suggest that the impact of a light stemming on the accuracy of text classification by topic is ineffectual. The accuracy is neither increased nor decreased on the stemmed texts, whereby the stemming algorithm helped reducing the dimensionality of feature vector space model.
Keywords: *stemming, classification, clustering, hierarchical, SOM, topic, content words*

INTRODUCTION

The task of quantitative topic classification of written texts has become popular with the huge increase and the variety of written texts of all kinds which may vary according to the use, subject matter, author's knowledge, and textual varieties, or events. All of this has led to the study of different text types, such as narrative, non- fiction, poetry and so on, all with their own lexical and syntactic patterns. A quantitative topic classification relies on methods developed in natural language processing and machine learning to analyse textual documents. While textual documents must be converted into a quantitative form prior to analysing them, several conceptual issues in data creation may hinder any quantitative textual data analysis. For example, the text data can in

general be very sparse because of the large number of redundant lexical features. This can be attributed to the fact the English language has several morphological variants of a single word. Pre-processing procedures such as cleaning and preparing raw texts for analysis, and word stemming are commonly carried out before applying an analytical method to build a robust pattern. The principal is that it is essential to adjust text data by removing repetition and transforming words to their common base or root form through stemming. This is to reduce the dimensionality of the feature dimension to make it easier to analyse and process text and help in grouping variations of words together, which can be useful for tasks like text classification or clustering. However, word stemmer is known to produce nonsense or incomplete words and this is very likely to skew the text data and therefore the classification results based on it. By way of explanation, this study is based on a corpus of thirty texts that focus on the topics of politics, history, science, prose, sport, and food written by six authors. Multivariate analytical methods are used to extract a set of lexical features that define each text so that the thirty texts can be classified using linear hierarchical clustering and non-linear clustering method SOM. In topic classification by lexical features, the time and complexity of classification process are two important problems that affect data analysis. Although this is crucial, easy and short processing should not be accepted at the cost of classification accuracy. As thus, this study is designed to examine the impact of stemming on the text topic-based classification by analysing the thirty texts with and without stemming to determine which courses are more accurate than others. This will be discussed in detail in the subsequent sections.

Research Problems

Text classification attempts to assign written texts to specific group types that share the same linguistic features. To do so, the basic or common approach to is to look at lexical words and their frequencies in a given text. The analyst takes the text to be classified and counts the frequencies of the words and select the most distinguishing words of a given text, followed by some text pre-processing steps to keep the resulting data matrix of a manageable size. Because lexical words and frequency play a role in text classification based on clustering, this can cause conceptual issues in text data creation in at least two ways: (1) the curse of dimensionality and (2) lexical redundancy/ambiguity. Dimensionality is a key issue for data analysis in any given application (Moisl, 2015). In this application the vector space model is used to represent texts and lexical features as

vectors in a multi-dimensional space. Each dimension represents a unique lexical feature frequency in the entire corpus of texts. For example, when analysing written texts verbs, adjectives, nouns, adverbs, prefixes, suffixes, word length, word frequency, word cluster and high frequency word distribution, etc could each be a dimension. Each dimension corresponds to a unique feature, while the texts can be represented as a vector within that space. As the number of lexical features increases, and thus the number of dimensions, moves from low to high dimensional spaces, text data starts to behave differently and make analytics more challenging, as shown in Figure (1) below.

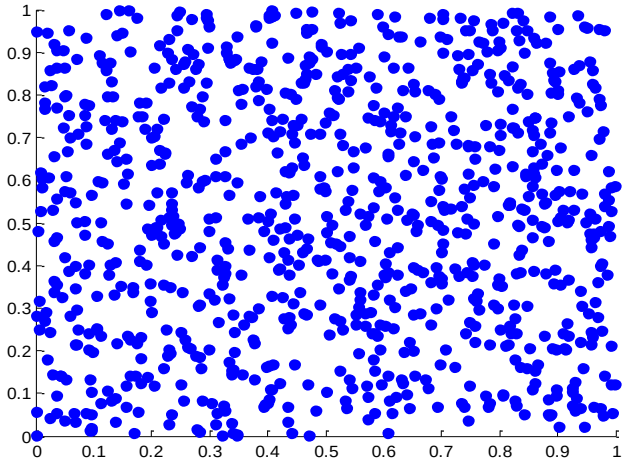

Figure 1 . Lexical features plotted on a 2- dimensional space

For example, lexical items such as 'cat', 'cats', 'catty', 'cattery', and so on which are recognized as distinct lexical types or the morphological variants of the same word 'CAT' will be assigned four dimensions in the data matrix. If each of the four variables take integer values in the range 1...10. The ratio of data points to possible values is 10/(10 x10 x10 x10) =0.001, that is, the data points occupy 0.1% of the data space. It is, therefore, clear that lexical frequency text data will, in general, be very sparse on account of the large number of very infrequent lexical type variables. The known resolution is that dimensionality should be remained as low as possible appropriate to the need to define the field of research suitably (Lüdeling and Kytö: 2009).

The next problem is lexical ambiguity or redundancy which can result from various aspects of English morphology such as the word's stem, affixes, suffixes, bases, inflections and derivations. Written texts generally contain several morphological variants of a single word. As lexical frequency of occurrence is used in text data processing and creation, the existence of many morphological variants of words has negative effects on

the context of the word occurrence. There is an example of lexical ambiguity between two or three or even four distinct lexical types when they are related to the same stem. For example, words such as 'man', 'manly', 'unmanly', 'manliness', and so on are the morphological variants of a stem 'man' but are treated as distinct lexical types and therefore they will not be recognized as equivalent without the NLP tool gaining access to linguistic information or semantic context about them. (Senders, 2021; Jenkins and Smith, 2005). Some examples of lexical ambiguity come from a word includes a stem which denotes to some basic idea of meaning and that particular affixes have been attached to improve the meaning and/or to adjust the word for its syntactic structure. Brown and Miller (1980), for example, use 'cats' and 'attacked' and suggest that 'cat' is noun stem, -s is a plural marker (and pl. is a term in the grammatical category of numbers). 'Attack' is a verb stem, -ed is a past tense marker (and past is a term in the grammatical category of tense). These two examples are instances of inflectional morphology: that part of morphology which deals with the way in which lexical stems are brought together with grammatical markers like those for plural or past. Another examples are 'fearsome' and 'foolish'. The analysis of 'fearsome' as fear-some and of 'foolish' as fool-ish is equally clear and can be pursued in the same terms. The analysis of 'fearsome' and 'foolish', on the other hand, doesn't represent a grammatical analysis but describes the way lexical stems themselves are formed. In the case of fool-ish the addition of adjective –ish to the lexical stem 'fool' forms the new lexical stem 'foolish'. Similarly, in fear-some the adjective formative –some added to the noun 'fear' produces the adjective 'fearsome'. They are instances of derivational morphology, the part of morphology that deals with the way lexical stems are formed. This is a significant problem for text classification based on clustering, since the morphologically related words are treated exactly the same as unrelated ones as shown in Figure (2) which shows words plotted in a 2-dimensional space based on frequency and relevance.

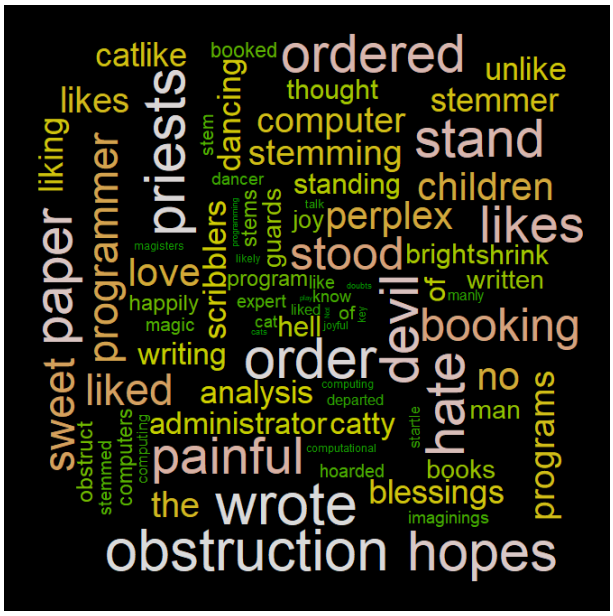

Figure 2. A cloud of most frequent unstemmed words plotted in a 2-dimensional space

The problem is that there is no difference between the semantic similarities among morphological variants of a single stem on the one hand, and those between unrelated stems on the other, for example, the semantic difference between 'likes', ‘liking’, ‘unlike’, ‘catlike’, and ‘liked’ is taken to be the same as that between 'man' and 'order'. When facing such a problem, the solution is usually to apply word stemming to improve text classification and allow for more effective text analysis by reducing different forms of words to their root or base form and reduce dimensionality by converting words to their stems (Singh and Gupta, 2019). Basically, stemming involves simplifying the dataset or projecting it from a multi-dimensional area into a low-dimensional area so that the low-dimensional characterization takes some significant features of the original dataset, in principle close to the number of features needed. In this study, I explore the impact of stemming on text classification accuracy, focusing on two questions. First, does stemming impact (increase, decrease, or neglect) the text classification’s accuracy? Second, can stemming and be used along with a feature reduction method to simplify the number of dimensions in a dataset?

Background and related work

Given its significance in different fields, such as information retrieval, text mining, sentiment analysis, or content analysis, considerable research has been devoted to text classification. Such studies focus on different approaches by using one or more steps of

the classification research project, including text pre-processing, features construction, feature weighting, feature selection, classification pattern and evaluation. Below is some of the existing works in the field from recent years.

Dogan and Uysal (2020) proposed a novel supervised term weighting scheme called TF-MONO and SRTF-MONO and compared its performance against the existing term weighting schemes in the literature using two different classifiers such as SVM and KNN applied on three different datasets named Reuters-21578, 20-Newsgroups, and WebKB. The findings from seven distinct schemes demonstrated that, on average, SRTF-MONO performed better than the other schemes across the three datasets. Furthermore, in comparison to the other five benchmark term weighting techniques, TF-MONO had guaranteed both Micro-F1 and Macro-F1 results, particularly on the Reuters-21578 and 20-Newsgroup's datasets. In their work, HaCohen-Kerner et al. (2020) performed an extensive and systematic set of text classification experiments to assess the impact of all possible combinations of five/six basic preprocessing techniques on the four examined corpora using three machine learning methods. They concluded that it is always advisable to perform an extensive and systematic variety of preprocessing methods combined with text classification because it contributes to improve classification accuracy. In their study, Hartmann et al. (2019) compared the performance of five lexicon-based and five machine learning methods across forty-one social media datasets. The results showed that given small sample numbers, RF consistently performs well for three-class emotion, NB. SVM never performed better than the other techniques. In comparison to machine learning, all lexicon-based methods performed badly, with LIWC performing the worst. Accuracy margins in certain applications were marginally better than chance. The findings implied that marketing research can profit from taking into account NB and RF since additional factors of text classification choice were also in their favour. Wan et al. (2019) conducted a text classification study by using syntactic and unigrams features to obtain what is known as syntax augmented bi- grams (SAB). The experiments showed that the use of syntax was useful in text classification problems. They concluded that it can be used to extract stable phrases for some NLP tasks, like question answering and machine translation, and the χ2.rcf evaluation method of such composite features can assist in reducing the dimensionality of the document vectors by discarding redundant features. In quantitative literary analysis, the work by Ardanuy and

Sporleder (2014) revolved around the task of text clustering by building social networks from novels. Instead of clustering the selected novels by means of content-based features, the authors constructed a vector of features by quantifying their plots and structures. The results of this experiment showed that using such features can be useful in text clustering by topic or authors. In their work on topic identification, Worsham and Kalita (2018) addressed the topic identification problem, which is a very long text classification task that requires both syntactic and thematic analysis in order to assign a literary genre to a book from a corpus. They assigned the literary classification to a full-length book belonging to a corpus of literature, where the works on average are well over 200,000 words long and genre is an abstract thematic concept. Along with the genre identification problem, different machine learning approaches were addressed and evaluated as solutions for assigning genre. The study found that for the task of classifying long full-length books by genre, gradient boosting trees are superior to neural networks, including both CNNs and LSTMs. The study not only demonstrated that the use of words from all chapters was beneficial for the classification task, but it also showed that traditional machine learning methods can achieve a better performance than more complex deep learning models.

Materials and procedure

1. *Corpus*

The examined corpus in this study were thirty electronic essays that focus on the topics of politics, history, science, prose, sport, and food collected from https://www.ukessays.com/. The composition of corpus material is the following: (five politics texts), (five science texts), (five prose texts), (five sport texts), (five food texts), and (five history texts). These texts were saved in text.doc format. There was considerable variation in the lengths of these texts available for a given article. To equalize the lengths among the different texts, I sampled the texts from 850 up to a maximum of 1000 words to make them comparable to each other or about the same size accordingly. Here I built a corpus of thirty texts that are truly representative of each topic, and the size of each text is in harmony with each other prior to analyzing it. The text corpus of the study is shown in the Table below.

Table 1. Corpus of thirty texts

food1.txt	food2.txt	food3.txt
food4.txt	food5.txt	his1.txt
his2.txt	his3.txt	his4.txt
his5.txt	pol1.txt	pol2.txt
pol3.txt	pol4.txt	pol5.txt
pro1.txt	pro2.txt	pro3.txt
pro4.txt	pro5.txt	sci1.txt
sci2.txt	sci3txt	sci4.txt
sci5.txt	spo1.txt	spo2.txt
spo3.txt	spo4.txt	spo5.txt

2. *Pre-processing*

To prepare the text data for text classification, pre-processing was performed before transforming it into numerical features. Four text pre-processing steps were involved in the current application. These are: cleaning raw texts, data generation, feature extraction and selection, and stemming.

A. Data cleaning

To enhance the quality of the text data, the raw text documents were cleaned from words or characters that do not add any value to the meaning of the whole text data, including punctuation marks, page numbers, non-standard formatting, titles, URLs, HTML tags, extra spaces, and so on.

B. Data generation

The thirty texts were broken into lexical tokens. These lexical tokens were used with their co-occurrence frequencies. The co-occurrence frequencies were converted into numerical vectors, where every lexical feature corresponds to the words in the corpus and every value to their respective frequencies, in which every dimension represents a word and every row vector represents a text.

C. Stemming

In Natural Language Processing (NLP) use applications such as text classification or clustering, thematic analysis, sentiment analysis, language translation, etc., getting word stem, base, or root form is important to help in the preprocessing of text and can also be used for query expansion. This is where stemming comes into play. Stemming is a process that removes prefixes and suffixes from words, reducing them into their stem, base or root form, generally known as a written word form. The general aim is to transform the

morphological variants of the words like chooses, choosing, chose, chosen to get linked to the word 'choose', or the words cats, catlike, cattery, catty to get linked to the word 'cat'. In stemming, transformation of morphological variants of a word to its stem can be created based on the assumption that each variant is semantically related for the purpose of textual analysis and information retrieval. The stem need not be an existing word in the lexicon but all its variants should link to this form after the stemming has been completed. However, the results of stemming are not always morphologically right forms of words. Some stemmers may produce nonsense or incomplete words (Jivani, 2011; Jayanthi and Jeevitha, 2015). For example, a stemmer may reduce the words argue, argued, argues, arguing, and argus to just the stem 'argu' or the words' introduction, introducing, introduces to get linked to just the word 'introduc', and that is because stemmer does not check on grammatical rules during the stemming process. For this reason, a lot of different stemmers or stemming programs have been developed to produce morphological variants of a root/base word. UEAstemmer (2005) is one of the most widely used stemmer. Other types of stemmers include, for example, Lovins Stemmer (1968), Porter (1980), Paice-Husk Stemmer (1980), Lancaster Stemmer (1990), Snowball Stemmer (2000), and Regular expression Stemmer, Each of these stemmers comes with its own set of strengths and weaknesses in a way that each of which differs in respect to performance and accuracy. Some stemmers may remove recognized suffixes based on the assumption that most suffixes in English are considered to be potentially removable. Other stemmers may remove some affixes and alter the meaning of a word so greatly and thus throw vital information away. However, elimination of prefixes doesn't basically appear to be useful in text data processing, except in certain domains, and thus is not relevant to the present application. In general, the stemming algorithms used have exhibited two shortcomings or limitations generally known as stemming errors (Paice:1994):

1. Under-stemming errors, happen when words which denote to the same concept or meaning are not converted to the same stem, as in data and datum.
2. Over-stemming errors, happen when words are reduced to the same stem even though they denote to different concept, as in author and authoritarian.

Where stemming errors can result in a loss of information and hinder text analysis, as here, a stemmer's limitation and advantage must be taken into account. More specifically,

the author suggests that cluster analysis methods are considered distance-based measures, since the inclusion of nonsense or incomplete words may change the clusters, thus, making it essential to choose a stemmer that meets a specific need or answers a particular question. While there are many stemmers to execute word stemming, the present study is oriented towards the following two rules while selecting a stemming algorithm to best suit it:

- ✓ Morphological variants of a word are presumed to have the same concept and therefore must be linked to the same stem. The implication is that if the base meaning or concept is the same but the word form is different it is essential to distinguish each word form with its root form.
- ✓ Morphological variants of a word that do not have the same concept must be kept apart.

These two assumptions are practical enough as long as stemming or the stemming program stems and groups words according to the same semantic root or concept. As the interest lied in text classification by topic using lexical features, for this experiment, only simple or light stemming was performed, i.e., only suffixes that added or attached to the end of a base word were removed. For this reason, UEA stemmer was used since it is intended to stem conservatively to orthographically correct word forms and recognizing words which do not need to be stemmed, such as proper nouns. So it is more probable to produce much fewer classes and that the stemmed words will still share the same meaning. It is available for research use in Perl and Java implementations at http://www.cmp.uea.ac.uk/research/stemmer. Figure 3 shows the stemmed corpus of thirty texts used for this study.

Figure 3. Stemmed corpus of thirty texts

D. Feature extraction/selection

Feature extraction is the task of detecting and selecting the most significant features from a dataset. Where feature selection removes redundant lexical features, in essence it removes text data matrix dimensions and therefore allow clustering methods to make more accurate classifications. Topic text clustering depends on there being variability in the frequency of the occurrences of values from each of the thirty texts in the dataset; similar texts having the same lexical features can't be logically clustered. Where the thirty text documents to be classified are defined by lexical features, then the lexical features are only helpful for the purpose if there is significant variation in the weight values that they take. In the early stages of the extraction process, all function words were removed as they coincide with content words. In the present application, thus, we looked for content words with considerable variation in their word count frequency weights, and removed content words with small or no variation. Content words with no or small variation were eliminated from the text data as they included small semantic details and would sophisticate cluster analysis by creating more multi-dimensional data than it should be. The simplest approach of determining the weight value is variance. The weight

value of each lexical feature was calculated by its variance. This means the variance of lexical feature weight values is the average deviation of those weights from their mean.

$$v = (\sum_{i=1..n} (x_i - \mu)^2) / n$$

If we have dataset of n weight values $\{x_1, x_2...x_n\}$ allocated to a lexical feature x. The mean of these weight values μ is $(x_1 + x_2 + ... + x_n) / n$. The magnitude by which any given value x_i differs from μ is then $x_i - \mu$. The average difference from μ across all values is thus $\Sigma_{i=1..n} (x_i - \mu) / n$.

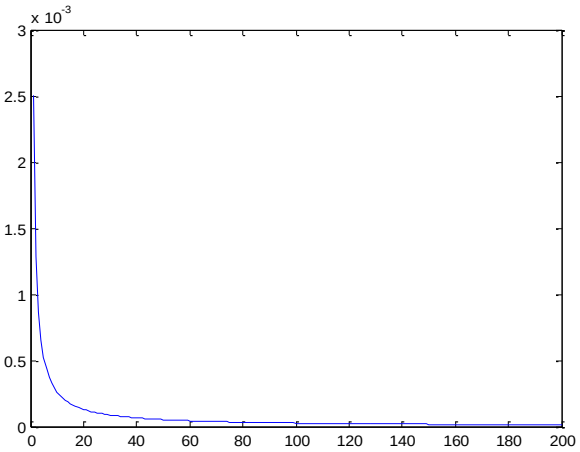

Figure 4.Variance lexical frequency

Here a conservative dimensionality reduction was made by keeping the 40 highest-frequency content words and removing the rest on the basis that they account for only a small fraction of the differences among the thirty texts. The selected 40 highest-frequency content words are shown in Table (2):

Table 2. Forty content-based words extracted by variance

crucial	believe	changed	causes
ago	makes	increase	gone
require	remained	able	become
end	human	present	come
give	taken	allow	large
available	clear	far	always
evidence	grow	including	beauty
alike	especially	feel	according
basic	important	done	same
used	perform	concern	improve

Table 2. Forty content-based words extracted by variance

Feature selection and extraction through applying variance and UEA stemming were able to reduce dimensionality by trimming down the number of features from 200 to 40.

3. Methods and analysis

In this study, we applied two clustering methods using a bag of words. The motivation of using two different methods is to compute similarity between any two clusters in different ways: one is linear hierarchical, the other is non-linear clustering SOM. The most common use of cluster analysis is text classification (Fielding: 2006). The aim of cluster analysis is to classify written texts together into groups that share similar characteristics as determined by some measure of similarity. Euclidian distance was used to measure the distance between each pair of texts, where distance between each pair of texts means some similarity/proximity measure over the whole set of characteristics. This is used most commonly. It simply measures the distances between the pairs of text data points by calculating the square root of the sum of the squared differences between the measurements for each lexical feature. This is expressed by the function:

$$length(z) = \sqrt{(length(x))^2 + (length(y))^2}$$

When performing cluster analysis, we assign the characteristics of written texts to each group to compute and the semantic distance/similarity in-between the text data. Written texts are separated into groups (called clusters) based on the basis of how closely associated they are so that each written text is more similar to other texts in its group than to texts outside the group. As the hierarchical cluster method, Ward linkage clustering (known as minimal increase of sum-of-squares) is used because it is most frequently used in studies where clusters are expected to be solid, compact, and even-sized. It was used along with the clustering tendency to determine whether the resulted clusters have a grouping structure. Ward linkage method specifies the amount of proximity between any couples of texts data in the matter of decreasing of changeability based on a criterion which takes two estimates: relative to a cluster A, (1) the error sum of squares (ESS) is the sum of squared differences of the data vectors in A from their group's mean, and (2) the total error sum of squares (TESS) of a set of *p* clusters is the sum of the ESS of the *p* clusters. At each successive clustering step, the ESS of the *p*

clusters available for fusing at that step is computed. For each original amalgamation of cluster couples the increase in TESS is noted, and the couple which leads to the smallest increase in TESS is fused into a single cluster (Romesburg, 2004). The analysis was done in two stages: the first was conducted on the original text documents (i.e. no word stemming done) and the second was conducted on the stemmed texts. The results of the hierarchical clustering and the clustering tendency are shown in Figure(5).

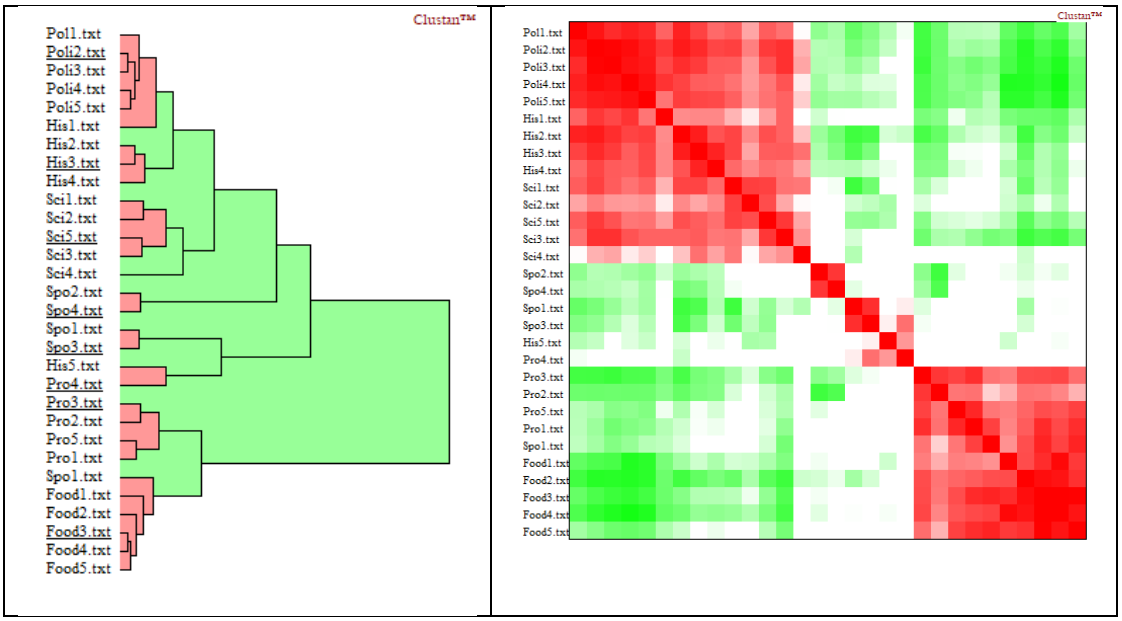

Figure 5. Ward linkage clustering and clustering tendency

The text membership of the hierarchical analysis is shown in the table that follows.

Table 3. Text membership of Ward linkage clustering

Cluster	Membership					
1	Poli1	Poli2	Poli3	Poli4	Poli5	His1
2	His2	His3	His4			
3	Sci1	Sci2	Sci5	Sci3	Sci4	
4	Spo2	Spo4	Spo1	Spo3		
5	His5	Pro4	Pro3	Pro2	Pro5	Pro1
6	Spo5	Food1	Food2	Food3	Food4	Food5

As for the non-linear cluster method, SOM is used to build a topology preserving map from a high-dimensional input area onto a map unit so that relative distances between text data points are preserved. SOM computes the nonlinear distances between text data points and is introduced with various coloring charts. It preserves the topological information about a set of data in a two-dimensional visual image. Given a relevant measure of similarity, text data points which are located closely to one another in multi-dimensional space are located close to one another in their two-dimensional map, and text data points which are located relatively away from one another in multi-dimensional space are clearly set apart leading to a well-built pattern (Kohonen, 2001; Oja, et al., Vesanto, 1999). The analysis was a two-level procedure. The first was the teaching SOM by uploading all the text data into the input area. The second was the creation of the two-dimensional pattern on the plane. For each text data, the weight values in the input area were generated through all the connections to the units in the lattice. Because of the difference in connection weight, a given text data started up one unit more robustly than any of the others, whereby linking each text data with a specific unit in the lattice. Once all the text data had been plotted in that manner, the outcome was a pattern of activation across the lattice. The SOM output used the relative distance between connection text data to search for group borderlines. The Euclidean distances between the connection text data related to each map unit and the connection text data of the directly adjoining units were calculated and figured, and the outcome for each was arranged in a new matrix model, having the same dimensions as the original. The text data was uploaded using a color chart to constitute the relative amounts of the weight values in which a dark chart between the text data correlates to a big distance and, therefore, constitutes a gap between the weight values in the input area. A bright chart is

the borderlines between groups, showing that the texts data are close to each other in the input area. Bright regions correspond to groups and dark regions group dividers. Any outstanding group borderlines will be detectable. The colour range is shown close to the right side of the map, which includes numbers indicating the values of texts data and that of the distances between adjacent texts data (Moisl, 2015). As above, the analysis was done in two stages: the first was conducted on the original text documents (i.e. no word stemming done) and the second was conducted on the stemmed texts. The result of SOM is shown in Figure (6):

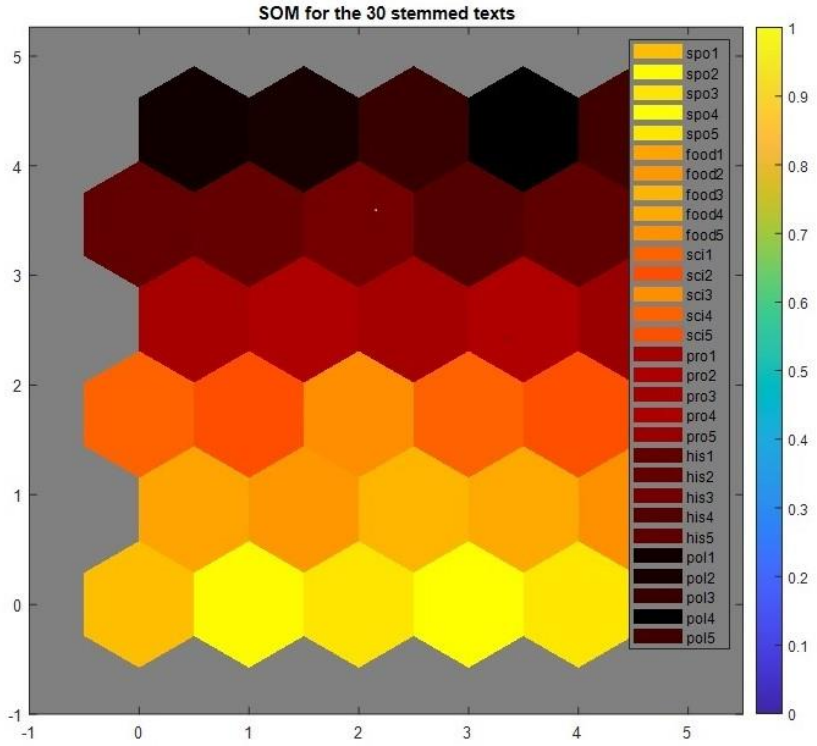

Figure 6. SOM of the thirty texts

The text membership of the SOM analysis is shown in the table that follows.

Table 4. Text membership of SOM clustering

Cluster	Membership						
1	Spo2	Spo3	Spo4	Spo5			
2	Spo1	Food1	Food2	Food3	Food4	Food5	
3	Sci1	Sci2	Sci3	Sci4	Sci5		
4	Pro1	Pro2	Pro3	Pro4	Pro5		
5	His1	His2	His3	His4	His5	Poli3	Poli5
6	Poli1	Poli2	Poli4				

RESULTS & DISCUSSION

To explore whether executing word stemming can improve classification's accuracy, two sets of analyses were carried out at two stages. The study first attempted to examine the original text documents with no word stemming executed using hierarchical clustering and SOM. In the second test, the study attempted to examine the thirty texts with word stemming executed. These texts were analysed for all the possible combinations of text pairs in which the order of texts in each analysis did not matter. This is because various clustering algorithms may produce different clusters on the same dataset and the principal is that texts in the same cluster are more similar to each other than they are to texts in another cluster. In Figure (5) and Figure (6), the analyses enabled to identify six main groups of texts according to the content of their work: one group for politics, the second group for history, the third group for food, the fourth group for sport, the fifth group for prose, and the sixth group for science. These six clusters of texts were identified as having similarity in each cluster and it can therefore be assumed that their topic is related. Table 5 shows a good degree of correspondence between the six main clusters of hierarchical analysis and the clusters in the six main SOM clusters:

Table 5. Tabulation of texts in hierarchical clustering and SOM

Hierachical analysis	SOM
Cluster 6	Cluster 2
Cluster 3	Cluster 3
Cluster 4	Cluster 1
Cluster 5	Cluster 4
Cluster 1	Cluster 6
Cluster 2	Cluster 5

The results obtained from the hierarchical match those identified in the SOM despite the low marginal differences, but the whole picture is clear. The agreement between the methods ensures the accuracy of experimental results to support the validity of classification results. Based on these results, the researcher was able to notice that execution of word stemming had a negligible impact on the classification of the thirty text documents; the results were the same for the stemmed and unstemmed texts since the stemmer used here is a light stemmer which has been developed to stem words to root forms which are lexically full words. Table (6) below shows the result of stemming 24 words by UEA stemmer:

Table 6. 24 sample words stemmed by means of UEAStemmer

UEAStemmer Results

Original Word	**Stemmed Word**	**Original Word**	**Stemmed Word**	**Original Word**	**Stemmed Word**
sings	se	loved	love	foolish	foole
played	played	unhelpful	unhelpful	fearsome	fearsome
playing	playing	seeing	seeing	manly	manly
unreal	unreal	sung	sung	friendlies	friendlies
helpful	helpful	performance	performance	indefinitely	indefinitely
unmanly	unmanly	runly	runly	experiment	experiment
axes	axes	dries	dries	manliness	manline
ran	ran	yond	yond	misled	misled

In Table (6), it is evident that UEA stemmer produces smaller categories permit words to keep their meaning by limiting the number of inaccurate stemming results. This is particularly useful when using stemming in text classification. Such a result will provide an opportunity to advance our knowledge of the link between light stemming and text classification.

CONCLUSION

The primary purpose of the present study was to explore the impact of word stemming on topic-based text classification. Thirty text documents were classified using hierarchical clustering and SOM and applying UEA stemming algorithm. The results demonstrated that the word stemming process had little effect on the classification of these texts; that is, since the stemmer being used here is a light stemmer designed to stem words to root forms that are lexically full words, the clustering results for the texts that were stemmed and those that weren't showed no difference in clustering. Because there does not seem to be much difference between stemmed texts and unstemmed texts included in the analyses by means of hierarchical clustering and SOM, word stemming appears to have a negligible impact on the text classification accuracy. This conclusion requires further testing, however, particularly on stemming method in which one might argue that many similarities of text topics may not necessarily be the result of negligible impact of word stemming and may merely represent general text topic similarities that would have existed even in the absence of a stemming algorithm but are subsequently attributed to it. Such testing may suggest the use of other stemming algorithm to determine whether this will impact (increase or decrease or neglect) text classification's accuracy as noticed with UEA stemmer in this study. It seems clear that a systematic test will only be possible by collecting a large balanced dataset for each text type with a larger number of features that doesn't yet exist in current study efforts.

REFERENCES

Asian, J., Williams, H., and Tahaghoghi, S. (2005). *Stemming Indonesian*. In Proceedings of the twenty-eighth Australian Computer Science Conference, ACS, 307-314, Newcastle, Australia. CRPIT, 38. Estivill-Castro, V., ed.

Ardanuy Mariona Coll and Sporleder Caroline. (2014). *Structure-based Clustering of Novels*. Paper presented in the 3rd Workshop on Computational Linguistics for Literature (CLFL). Gothenburg, Sweden, April 27, 2014.

Baayen, R. (2001). *Word frequency distributions*. Dordrecht: Kluwer.

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern information retrieval*. Addison-Wesley.

Dogan, Turgut and Uysal, Alper Kursat. (2020). A novel term weighting scheme for text classification: TF-MONO. *Journal of Informetrics*, Volume 14, Issue 4.

Fielding Alan. (2006). *Cluster and Classification Techniques for the Biosciences*. Cambridge University Press.

Frakes, W. (1992). Stemming Algorithms. In W. Frakes & R. Baeza-Yates (eds.), *Information Retrieval*, 131-60. NJ: Prentice Hall.

Fuller, M. and Zobel, J. (1998). *Conflation-based comparison of stemming algorithms*. In Proceedings of the third Australian Document Computing Symposium, Sydney, Australia.

Goweder, A. (2004). *Stemming and Arabic information retrieval: the case of broken plurals*. PhD thesis, Department of Computer Science, University of Essex.

HaCohen-Kerner Yaakov, Miller Daniel, and Yigal Yair. (2020). *The influence of preprocessing on text classification using a bag-of-words representation*. Http://doi: 10.1371/journal.pone.0232525. PMID: 32357164; PMCID: PMC7194364.

Hartmann Jochen., Huppertz, Juliana, Schamp Christina, and Heitmann Mark. (2019). Comparing automated text classification methods. *IJRM* Volume 36, Issue 1.

Hull, D. (1996). Stemming algorithms- a case study for detailed evaluation. *Journal of the American Society for Information Science* 47 (1), 70-84.

Jasmeet Singh and Gupta Vishal. (2019). A novel unsupervised corpus-based stemming technique using lexicon and corpus statistics. *Knowledge-Based Systems*, Volume 180: 147-162.

Jayanthi R and Jeevitha C. (2015). An Approach for Effective Text Pre-Processing Using Improved Porters Stemming Algorithm. *IJISET-International Journal of Innovative Science, Engineering & Technology,* Vol. 2 Issue 7: 797-802.

Jivani Anjali Ganesh. (2011). A Comparative Study of Stemming Algorithms. *IJCTA*, Vol 2 (6): 1930-1938.

Khoja, S. and Garside, R. (1999). *Stemming Arabic text*. Computing Department, Lancaster University, Lancaster, U.K. Retrieved February, 9th 2024 from http://www.comp.lancs.ac.uk/computing/users/khoja/stemmer.ps.

Kohonen, Teuvo. (2011). *Self-organizing maps*. 3rd ed. Berlin: Springer-Verlag.

Kraaij, W. and Pohlman, R. (1994). Porter's stemming algorithm for Dutch. In L. G. M. Noordman and W. A. M. de Vroomen (eds.), Informatiewetenschaap1994: Wetenschappelijke bijdragen aan de derde STINFON Conferentie, Tilburg, 167-180.

Kraaij, W. and Pohlman, R. (1996). *Viewing stemming as recall enhancements*. In Proceedings of ACM SIGIR-96, 40-48, Zurich, Switzerland.

Lovins, J. (1968). Development of a stemming algorithm. *Mechanical Translation and Computational Linguistics*, 11 (1), 22-31.

Lüdeling, Anke and Kytö, Merja. (2009). *Corpus Linguistics: An international handbook*. Volume 2. Germany: Walter de Gruyter.

Marie-Claire Jenkins and Smith, Dan. (2005). *Conservative stemming for search and indexing*. Retrieved from http://lemur.cmp.uea.ac.uk/Research/stemmer/stemmer25feb.pdf

Moisl, Hermann. (2015). *Cluster Analysis for Corpus Linguistics*. Berlin: De Gruyter Mouton.

Oja, M., Kaski, S., and Kohonen, T. (2001). Bibliography of self-organizing map (SOM) papers: 1998-2001, *Neural Computing Surveys* 3, 1-156.

Paice D. Christopher. (1994). An evaluation method of stemming algorithms. Paper presented in the SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR 94), 1994.

Romesburg, Charles. (2004). *Cluster Analysis for Researchers:* (Tokyo, Uchida Rokakuho Publishing Co., Ltd.

Savoy, J. (1993). Stemming of French words based on grammatical categories. *Journal of the America Society for Information Science* 44, 1-9.

Senders Youri. (2021). *The impact of stemming and lemmatization applied to word vector based models in sentiment analysis*. The Tilburg University master's thesis.

Vesanto, J. (1999). SOM-based data visualization methods. *Intelligent Data Analysis* 3, 111-126.

Wan Chaun, Wang Yuling, Liu Yaoze, Ji Jinchao, and Feng Guozhong. (2019). Composite Feature Extraction and Selection for Text Classification. *IEEE*, vol.7: 35208-35219.

Worsham Joseph and Kalita Jugal. (2018). *Genre Identification and the Compositional Effect of Genre in Literature*. Paper presented at the 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA, August 20-26, 2018.