JQSR

http://publisher.uthm.edu.my/ojs/index.php/jqsr
e-ISSN : 2773-5532

Journal of Quranic Sciences and Research

Mini-Batch *k*-Means versus *k*-Means to Cluster English *Tafseer* Text: View of Al-Baqarah Chapter

Mohammed A. Ahmed[1,2*], Hanif Baharin[1], Puteri N. E. Nohuddin[1]

[1]Institute of IR 4.0,
Universiti Kebangsaan Malaysia, Bangi, Selangor, 43600, MALAYSIA

[2]Network Engineering Department, College of Engineering,
Al-Iraqia University, 10053, Baghdad, IRAQ

*Corresponding Author

DOI: https://doi.org/10.30880/jqsr.2021.02.02.006
Received 14 October 2021; Accepted 23 October 2021; Available online 19 December 2021

Abstract: Al-Quran is the primary text of Muslims' religion and practise. Millions of Muslims around the world use al-Quran as their reference guide, and so knowledge can be obtained from it by Muslims and Islamic scholars in general. Al-Quran has been reinterpreted to various languages in the world, for example, English and has been written by several translators. Each translator has ideas, comments and statements to translate the verses from which he has obtained (*Tafseer*). Therefore, this paper tries to cluster the translation of the *Tafseer* using text clustering. Text clustering is the text mining method that needs to be clustered in the same section of related documents. The study adapted (mini-batch *k*-means and *k*-means) algorithms of clustering techniques to explain and to define the link between keywords known as features or concepts for Al-Baqarah chapter of 286 verses. For this dataset, data preprocessing and extraction of features using Term Frequency-Inverse Document Frequency (TF-IDF) and Principal Component Analysis (PCA) applied. Results showed that two/three-dimensional clustering plotting assigning seven cluster categories (k = 7) for the *Tafseer*. The implementation time of the mini-batch *k*-means algorithm (0.05485s) outperformed the time of the *k*-means algorithm (0.23334s). Finally, the features 'god', 'people', and 'believe' was the most frequent features.

Keywords: Text mining, text clustering, *k*-means algorithm, mini-batch *k*-means algorithm, *tafseer* translation, Al-Baqarah chapter

1. Introduction

Text mining is a molecule for processing information, data interpretation, data machine learning, and computer linguistics. Many information is collected, including news, research papers, books, digital libraries, reviews, and web pages. Consequently, work in text mining was very involved. A significant aim is to obtain high-quality data from this huge text. Usually, text mining tasks include text analysis, text classification, concept/entity extraction, granular taxonomics processing, sentiment analysis, clustering and summary documentation (Gregorio, 2019; Han *et al*., 2012).

Text clustering is the text mining method that needs to be clustered into the same group of related documents. This is usually achieved by discovering patterns and trends through statistical pattern analysis, topic modelling and statistical language modelling. Text mining typically needs to format the input text (preprocessing) before using text mining techniques (Han *et al*., 2012) (Karthikeyan *et al*., 2019). Text clustering is like data mining documents loaded in the weight vector term are clustering objects. The standard techniques of clustering including partitioning clustering (*k*-

**Corresponding author: p103761@siswa.ukm.edu.my*

means) and (mini-batch *k*-means) (Feizollah *et al*., 2014), density-based (DBSCAN) (Indah *et al*., 2019), hierarchical (network analysis map) (Chua & Nohuddin, 2017), and grid-based (STING) (Han *et al*., 2012).

Al-Quran is text data that requires further research. Muslims believe that al-Quran was revealed from GOD (Allah) to Prophet Mohammed (SAW). Al-Quran is written in Arabic and is translated into numerous world languages such as English that several translators have written. The longest Quran's (Sura) chapter is Al-Baqarah; its verses cover different aspects. These aspects are not sequentially written but are equipped for asbabunnuzul ayat (verses). Al-Baqarah is predictable as a cluster since the verses will group according to the similarity of the text, which is represented the aspects (Huda *et al*., 2019).

Al-Quran has been translated into numerous world languages such as English, which several translators have written. Each translator has his or her commentary and statements describing the verses from which (*Tafseer*) was acquired. This study aims to find relationships and to cluster keywords of Al-Baqarah chapter features or concepts using mini-batch *k*-means and *k*-means clustering algorithms.

The rest of this paper is arranged as follows. Section 2 explores the previous researches. Section 3 discusses the methodology of research. In Section 4, an experimental procedure is given. The results and findings are discussed in Section 5. Finally, the conclusions are provided in Section 6.

2. Related Work

This review includes several papers. In this study (Huda *et al*., 2019), several models for identifying the Quranic verses in the Al-Baqarah chapter were developed into three cluster techniques: *k*-means, *k*-medoids, and bisecting. Every verse in the Al-Baqarah chapter, with 286 lines, was interpreted as an English translation text from the Qur'an. Three similarity tests are also used. Finally, the Sura Al-Baqarah type chapters, correlating with each other, were obtained. *k*-medoid with cosine similarity was the optimal finding.

A paper by Chua & Nohuddin (2017) introduces a combination of network analysis (map) text-mining techniques and TF-IDF (Term Frequency-Inverse Document Frequency) to gain keywords and to cluster relationships between *Tafseer*'s chapters and keywords. This experiment chooses 130 keywords of six short chapters (Sura) from the translated *Tafseer* Al-Quran. The method suggested was known as the KCRA framework. Slamet *et al*. (2016) study leading to the first practical process in the learning of the Holy Qur'an verse concepts. The algorithm is used to cluster a total of 6,236 verses by using partitioning (*k*-means) for stemmed/unstemmed terms, forming three clusters.

Hamoud & Atwell (2016) created an integrated Quran Question and Response Corpus, which is then clustered and displayed with available WEKA free Java software. The probability clustering process clusters the corpus into four clusters, with data obtained from four portals on the website. Meanwhile, an information retrieval web-based verses search system for Al-Qur'an has been developed and integrated with the clustering algorithm (SPC), which enables Muslims to detect relevant data in the Quran verse by dividing it into their group (Indra *et al*., 2019).

An Indonesian Quran Translation Semantic answering System (QAS) was developed by Putra, Gusmita, *et al*. (2016). The method asks users three questions, and a weighted vector (TF-IDF) will be generated for each term belonging to each type of response (also called the entity group).To feed user questions to semantic interpreters. The authors grouped 222 ontology concepts into 6, 24 and 77 on time, place and person concepts.

Putra, Hulliyah, *et al*. (2016) is a work that produces a weighted vector for each term of the Indonesian Quran Translation (ITQ) and uses the same semantic QAS as (Putra, Gusmita, *et al*., 2016). However, the author here shows more information on the effects of TF-IDF and has various work procedures.

Finally, Feizollah *et al*. (2014) examine the efficiency in the detection of Android malware of two clustering algorithms, namely mini-batch *k*-means and *k*-means. Results show that in detecting Android malware, the mini-batch *k*-means algorithm performs better than *k*-means.

3. Research Methodology

3.1. Pre-processing Operations

Data pre-processing is an essential step in the build-up of the system to make the performance more efficient. Data here has been translated into a more usable format (Harjanta, 2015). The preprocessing involves tokenising, POS tagging, case folding, stemming and deletion of stopping words. All the words in the documents are then changed to standardisation (Huda *et al*., 2019).

3.2. Weighting Processing (Feature Extraction)

Features in text mining tasks are usually known as the basic elements to be used for a document. A collection of term weighting functions, commonly used in text mining, may decide the significance of a candidate feature. One of the common functions is the Term Frequency-Inverse Document Frequency (TF-IDF) statistics to measure how important a term is to a document. Term Frequency (TF) in the document (d) is the number of times a word appears (t). On the other hand, a text's Inverse Document Frequency (IDF) measures a word's commonness across all documents. If the word

rarely appears in the document, the IDF value increases (Chua & Nohuddin, 2017; Harjanta, 2015). TF-IDF can be formulated as Eq. 1 (Karthikeyan *et al*., 2019).

$$TF - IDF(t, d, D) = tf(t, d) * \log(\frac{|D|}{d(t)}) \quad (1)$$

where d and D is a collection of documents, t is a term in the document (Sebastiani, 2002).

3.3. Clustering Algorithm (*k*-means & mini-batch *k*-means)

In comparison to the classification, the clustering of data into groups relies on the characteristics of the clustering where data does not have a prior label (unsupervised learning) (Han *et al*., 2012). Clustering methods including partitioning clustering (*k*-means and mini-batch *k*-means) (Feizollah *et al*., 2014), density-Based (DBSCAN) (Indah *et al*., 2019), hierarchical (network analysis map) (Chua & Nohuddin, 2017), and grid-based (STING) (Han *et al*., 2012). This paper adopted two partitioning algorithms (mini-batch *k*-means) algorithm technique (Sculley, 2010), which is a variation of the (*k*-means) algorithm to achieve the goal.

k-means is one of the best methods for partial clustering. In the clustering process, this approach uses a partitioning technique that iteratively minimises the space between the data from each node. The approach starts with the foundation of a random starting point for the cluster which converges iteratively (Han *et al*., 2012).

The mini-batch *k*-mean is the updated variant of the *k*-mean algorithm. Mini-batches are used in large datasets to reduce the calculation time. It also tries to optimise the outcome of the clustering. The mini-batch *k*-means requires mini-batches as an input that are random subsets of the entire dataset. The mini-batch *k*-means is considered faster than *k*-means and is typically used for large data sets. Hence this research has adapted it with its experiments.

The problem in *k*-means optimisation is finding the set E of cluster centres $e \in R^m$ with $|E| = k$, to minimise over a set XD of examples $xd \in R^m$ the following objective function in Eq. 2.

$$\text{Min} \sum_{xd \in XD} \|f(E, xd) - xd\| \quad (2)$$

$f(E, xd)$ returns the Euclidean distance the nearest cluster centre $e \in E$ to xd. Although the problem is NP-hard in general, it is well known that gradient descent methods converge to the local optimum when seeded with an original set of k examples drawn randomly from XD (Bottou & Bengio, 1995) (Sculley, 2010). The following is the algorithm of mini-batch *k*-means:

Algorithm: Mini-batch *k*-means.

Input: *k*, mini-batch size *s*, iterations it, data set XD
Initialise each $e \in E$ with an xd picked randomly from XD
Output: the set of clusters.
Steps:
Step 1: $z \leftarrow 0$
Step 2: **for** i = 1 **to** it **do**
Step 3: $M \leftarrow s$ examples picked randomly from XD
Step 4: **for** $xd \in M$ **do**
Step 5: $d[xd] \leftarrow f(E, xd)$ // Cache the centre nearest to xd
Step 6: **end for**
Step 7: **for** $xd \in M$ **do**
Step 8: $e \leftarrow d[xd]$ // Get cached centre for this xd
Step 9: $z[e] \leftarrow z[e] + 1$ // Update per-centre counts
Step 10: $\eta \leftarrow \frac{1}{z[e]}$ // Get per-center learning rate
Step 11: $e \leftarrow (1 - \eta)e + \eta xd$ // Take gradient step
Step 12: **end for**
Step 13: **end for**

3.4. Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is an ideal method for obtaining a pattern from a huge dataset that leads to the extraction of the range of eigenvectors related to the input distribution's significant eigenvalue (Wold *et al*., 1987). In this analysis, PCA is used to decrease the data column that TF-IDF weighted to create a cluster virtualising.

4. Experiments

Python version 3.7.7 is the software platform used for this study experiment. For the experiments in this article, there are four main phases. The flowchart of the experiment is shown in Fig.1.

Fig. 1 - Experimental scenario flowchart

4.1. Dataset

http:/tanzil.net/trans/ is a portal that offers several translation documents to various *Tafseer* translators in different languages, including English. Several papers use datasets obtained from this website, such as (Chua & Nohuddin, 2017; Chua & Nohuddin, 2014; Husin *et al*., 2017; Putra *et al*., 2016; Slamet *et al*., 2016; Sukmana *et al*., 2016; Zeroual & Lakhouaja, 2016). The purpose of this analysis is to discover the relationships and to cluster the characteristics of the chapter Al-Baqarah English *Tafseer* document by (Muhammad Sarwar) translator comprising 286 verses.

4.2. Preprocessing Operations

Preprocessing requires several stages of reading data, such as tokenising, POS tagging, stemming, case folding, and deletion of stopwords. This process is a necessary step to improve the precision of the final results. Table 1 gives information for the number of the total term for the *Tafseer* before and after this phase. This table shows a reduction in terms of each preprocessing stage.

Table 1 - The total number of terms during the preprocessing

Total terms	Total terms After (stopwords)	Total terms after (stopwords+stemming)
1659	1438	1126

4.3. Weighting Terms

The dictionary or corpus is assembled or compiled after processing the document by TF-IDF to assign a weight for each feature or term. Both are combined to make the corpus ready to be clustered. Fig. 2 displays *Tafseer*'s first fifteen features.

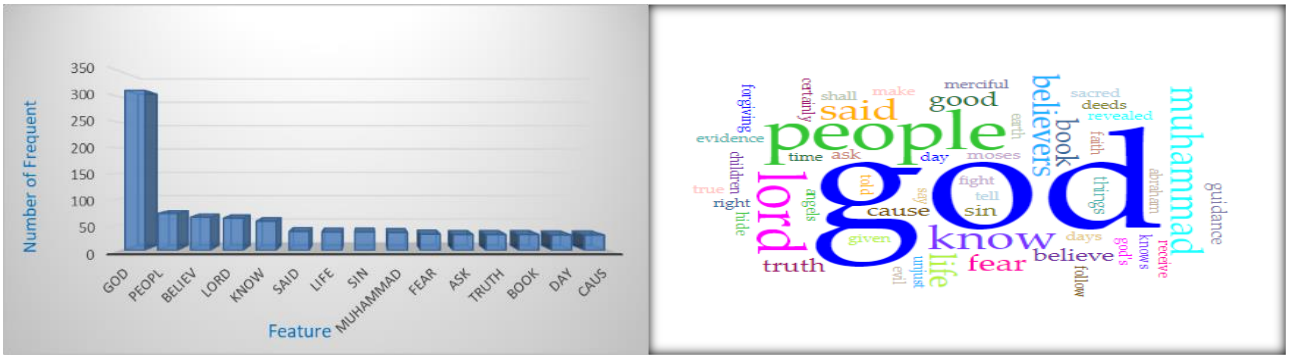

Fig. 2 - The *Tafseer*'s most frequent terms

4.4. Clustering Algorithm Parameters

The *k*-means and mini-batch *k*-means algorithms used for the clustering operation. The value *k* of this algorithm should be more than one (Gregorio, 2019; Huda *et al*., 2019). In this experiment, k is equal to seven based on studies of (Ahmed *et al*., 2020; Choiruddin, 2005; Huda *et al*., 2019) for the Al-Baqarah chapter. This chapter includes 53 subjects, where some subjects are the same as other subjects. The verses with the same subject are then grouped to form seven key themes with the number of verses in each theme.

5. Results and Discussions

The realistic and overview results of the analysis for this study are as follows:

- The document (*Tafseer*) used by the experiments is translated by Muhammad Sarwar. The document has 286 lines each line represent a verse from chapter Al-Baqarah.
- Table 1 shows the total number of terms obtained from the Al-Baqarah chapter after implementing the preprocessing operation.
- $k = 7$ for both *k*-means and mini-batch *k*-means algorithms.
- Fig. 2 shows the first fifteen features obtained after implementing a weighting term process using TF-IDF to produce the corpus.
- The corpus has a large number of columns (big dataset); hence PCA is used to reduce these columns to implement the visualisation process after the clustering operation. Fig. 3 shows 2D/3D visualisation for seven clusters of the *Tafseer*'s features.
- The implementation time for the mini-batch *k*-means algorithm is (0.05485s), and it is faster than (0.23334s) of the *k*-means algorithm.
- The most first three features for the Al-Baqarah chapter was ('god', 'people', and 'believe') respectively.

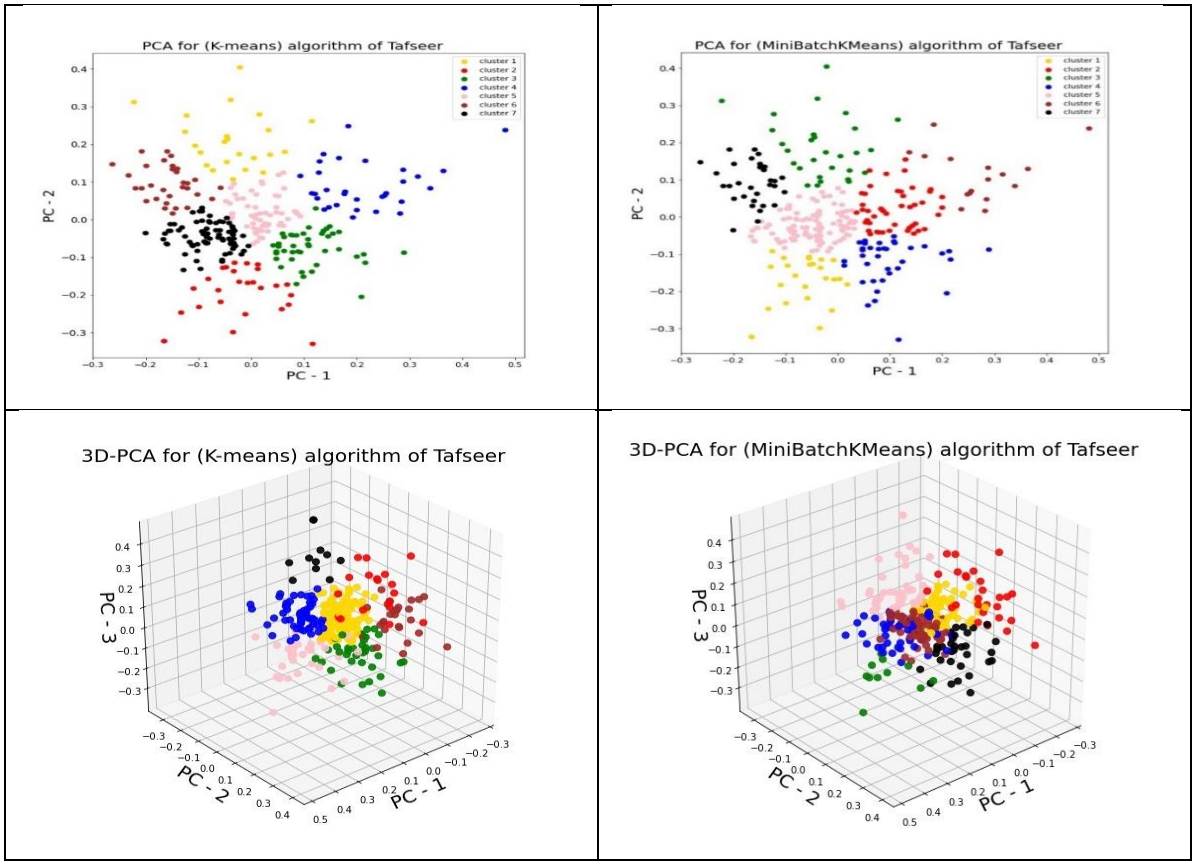

Fig. 3 - Two/three-dimensional plotting for features clustering of *Tafseer*

6. Conclusions

Mini-batch *k*-means and *k*-means cluster algorithms have been implemented with the experiments of this study. This study cluster the verses of the Al-Baqarah chapter into seven clusters. The document of the Al-Baqarah chapter was English text of *Tafseer* done by the translator (Muhammad Sarwar) consisted of 286 lines (verses). Before the clustering algorithms were implemented, the preprocessing operation (such as tokenising, POS tagging, stemming, case folding,

and deletion of stopwords) was applied. TF-IDF is used to create the corpus, and PCA is used to reduce the dimension of the corpus matrix for virtualisation purposes. The most three features were ('god', 'people', and 'believe') respectively, and mini-batch *k*-means algorithm implementation time was faster than the *k*-means algorithm.

This study benefited Muslims and researchers because Al-Quran is the primary book for them. For further work, the authors wanted to increase the other chapters of Al-Quran and to increase the number of translators regardless of language.

References

Ahmed, M.A., Baharin, H., & Nohuddin, P.N.E. (2020). Analysis of K-means, DBSCAN and OPTICS Cluster algorithms on Al-Quran verses. International Journal of Advanced Computer Science and Applications, 11(8), 248–254

Bottou, L., & Bengio, Y. (1995). Convergence properties of the k-means algorithms. Advances in Neural Information Processing Systems, 585–592

Choiruddin, H. (2005). Klasifikasi Kandungan Al-Qur'an. Jakarta: Gema Insani

Chua, S, & Nohuddin, P. N. E. (2017). Relationship Analysis of Keyword and Chapter in Malay-Translated Tafseer of Al-Quran. Journal of Telecommunication, Electronic and Computer Engineering (JTEC), 9(2–10), 185–189

Chua, Stephanie, & binti Nohuddin, P. N. E. (2014). Frequent pattern extraction in the Tafseer of Al-Quran. The 5th International Conference on Information and Communication Technology for The Muslim World (ICT4M), 1–5

Feizollah, A., Anuar, N. B., Salleh, R., & Amalina, F. (2014). Comparative study of k-means and mini batch k-means clustering algorithms in android malware detection using network traffic analysis. 2014 International Symposium on Biometrics and Security Technologies (ISBAST), 193–197

Gregorio, Z. (2019). Clustering Scholarship Programs Using Educational Data Mining Techniques. International Journal of Advanced Trends in Computer Science and Engineering, 8(3), 658–662

Hamoud, B., & Atwell, E. (2016). Quran question and answer corpus for data mining with WEKA. 2016 Conference of Basic Sciences and Engineering Studies (SGCAC), 211–216

Han, J., Pei, J., & Kamber, M. (2012). Data mining: concepts and techniques (3rd ed.). Elsevier

Harjanta, A. T. J. (2015). Preprocessing Text untuk Meminimalisir Kata yang Tidak Berarti dalam Proses Text Mining. Jurnal Informatika Upgris, 1(1 Juni)

Huda, A. F., Deyana, M. R., Safitri, Q. U., Darmalaksana, W., Rahmani, U., & others. (2019). Analysis Partition Clustering and Similarity Measure on Al-Quran Verses. 2019 IEEE 5th International Conference on Wireless and Telematics (ICWT), 1–5

Husin, M. Z., Saad, S., & Noah, S. A. M. (2017). Syntactic rule-based approach for extracting concepts from quranic translation text. 2017 6th International Conference on Electrical Engineering and Informatics (ICEEI), 1–6

Indah, R. N. G., Novita, R., Kharisma, O. B., Vebrianto, R., Sanjaya, S., Andriani, T., Sari, W. P., Novita, Y., Rahim, R., & others. (2019). DBSCAN algorithm: twitter text clustering of trend topic pilkada pekanbaru. Journal of Physics: Conference Series, 1363(1), 12001

Indra, Z., Adnan, A., & Salambue, R. (2019). A Hybrid Information Retrieval for Indonesian Translation of Quran by Using Single Pass Clustering Algorithm. 2019 Fourth International Conference on Informatics and Computing (ICIC), 1–5

Karthikeyan, M., Arivarasan, A., & Kumaresan, D. (2019). Performance Assessment of Various Text Document Features through K-Means Document Clustering Approach. International Journal of Advanced Trends in Computer Science and Engineering, 8(5), 1969–1977

Putra, S. J., Gusmita, R. H., Hulliyah, K., & Sukmana, H. T. (2016). A semantic-based question answering system for indonesian translation of Quran. Proceedings of the 18th International Conference on Information Integration and Web-Based Applications and Services, 504–507.

Putra, S. J., Hulliyah, K., Hakiem, N., Iswara, R. P., & Firmansyah, A. F. (2016). Generating weighted vector for concepts in indonesian translation of Quran. Proceedings of the 18th International Conference on Information Integration and Web-Based Applications and Services, 293–297

Sculley, D. (2010). Web-scale k-means clustering. Proceedings of the 19th International Conference on World Wide Web, 1177–1178

Sebastiani, F. (2002). Machine Learning in Automated Text Categorisation. ACM Computing Surveys, 34(1), 1–47

Slamet, C., Rahman, A., Ramdhani, M. A., & Darmalaksana, W. (2016). Clustering the verses of the Holy Qur'an using K-means algorithm. Asian Journal of Information Technology, 15(24), 5159–5162

Sukmana, H. T., Gusminta, R. H., Durachman, Y., & Firmansyah, A. F. (2016). Semantically annotated corpus model of Indonesian Translation of Quran: An effort in increasing question answering system performance. 2016 4th International Conference on Cyber and IT Service Management, 1–5

Wold, S., Esbensen, K., & Geladi, P. (1987). Principal component analysis. Chemometrics and Intelligent Laboratory Systems, 2(1–3), 37–52

Zeroual, I., & Lakhouaja, A. (2016). A new Quranic corpus rich in morphosyntactical information. International Journal of Speech Technology, 19(2), 339–346