p–ISSN: 2723 - 6609 e-ISSN: 2745-5254
Vol. 5, No. 12, Desember 2024 http://jist.publikasiindonesia.id/

Multi-Label Topic Classification on the Qur'an using the K-Nearest Neighbor and Latent Semantic Analysis Methods

Ghina Annisa Shabrina[1*], Kemas Muslim Lhaksmana[2]
Telkom University, Indonesia
Email: shabriina@student.telkomuniversity.ac.id[1*],
kemasmuslim@telkomuniversity.ac.id[2]
*Correspondence

Keywords: Qur'an; KNN; LSA; Hamming Loss.

ABSTRACT

The Qur'an, comprising over 80,000 words, 6,236 verses, and 114 surahs, presents a multifaceted and deeply significant text that demands a nuanced understanding of historical context, classical Arabic, and exegesis. To analyze and classify its content, various methodologies have been employed, including K-nearest neighbor (KNN) and Latent Semantic Analysis (LSA). This research investigates the effectiveness of combining KNN with LSA for multi-label topic classification of Qur'anic verses. The study reveals that KNN alone achieved a micro average F1-score of 0.49, demonstrating reliable performance, particularly for topics such as "aqidah" (creed) and "worldly matters." When LSA was applied with 100 components, there was a decrease in performance, reflected by a drop in the micro average F1-score to 0.43 and an increase in Hamming loss to 0.1657. However, as the number of LSA components increased to 200 and 300, performance improved, with micro average F1-scores rising to 0.45 and 0.47, and Hamming loss values decreasing to 0.1507 and 0.1466, respectively. This indicates that while LSA can enhance KNN performance, optimal results are achieved with a higher number of components.

Introduction

The Qur'an, as the holy book of Islam, serves as a guide for daily life. It consists of less than 80,000 words, 6,236 verses divided into 114 surahs, and is partitioned into 30 parts or juz. (Ta'a, Abdullah, Ali, & Ahmad, 2014). Although there is no universally accepted classification among Muslims, various approaches have been developed to understand and categorize the contents of the Qur'an.

The classification of Qur'anic verses has a unique characteristic where each verse can belong to more than one class, known as multi-label classification (Pane, Mubarok, & Adiwijaya, 2018). This differs from traditional classification, where each piece of data or document typically belongs to only one class. This uniqueness reflects the complexity of the Qur'an, which has been the subject of research across various disciplines such as

theology, linguistics, history (Wirastri, Nurhaeni, & Syahreni, 2017), and social sciences. The Qur'an influences over 1.5 billion Muslims worldwide (Dary, Bijaksana, & Sa'adah, 2015), and requires a deep understanding of its content. However, the process of understanding and interpreting its verses and the relationships between them is not simple. The deep and often layered semantic complexity within the Qur'an necessitates a careful and structured approach (Rivki & Bachtiar, 2017).

Therefore, rigorous and accurate testing methods are essential in classifying Qur'anic verse data. This involves using techniques such as multi-label classification, which allows for a deeper analysis of the relationships between verses and aids in providing more precise results in class determination. Among the many testing methods, KNN (K-Nearest Neighbor) is a widely used classification method. (Saputra & Yadi, 2020), (Ningrum, Hukom, & Adiwijaya, 2020), and is often combined with other methods to enhance classification outcomes.

This study aims to perform multi-label topic classification on pre-labeled Qur'anic verses. (Muflihah, Lhaksmana, & Bijaksana, 2024) By combining it with Latent Semantic Analysis (LSA). LSA can assist in dimensionality reduction, noise reduction, and improving performance in multi-label classification by better grouping data based on key concepts, thereby enabling KNN to work more effectively in detecting similarities among complex samples (Syahraeni, 2017).

Method

The research methodology employed in this study is a classification method using KNN (K-Nearest Neighbor) combined with LSA (Latent Semantic Analysis) in the hope of improving classification outcomes. The dataset used consists of the Qur'an, where each verse has been multi-labeled (Muflihah et al., 2024). The dataset includes seven labels: aqidah (creed), akhlak (morality), Syariah (Islamic law), ilmu (knowledge), kisah (stories), alam gaib (unseen world), alam dunia (worldly matters), and alam akhirat (the hereafter).

A total of 6,236 data points are used, with the dataset split into training and test data in a 7:3 ratio. This results in 4,365 data points used for training and 1,871 data points used for testing.

Preprocessing

Preprocessing is a necessary step in handling raw data that may not be compatible with the system, potentially affecting the results during data processing. In text classification, several common processes are involved:

a. Case Folding: This process converts all characters in the text to lowercase, ensuring uniformity.
b. Remove Punctuation: This step removes any characters other than letters and necessary punctuation marks.
c. Tokenization: This process splits the text into individual words or tokens (Putrisannim T. H., Adiwijaya, A., and Faraby, S. A. 2019).

After preprocessing, each verse in the dataset is assigned a value of 1 for each label it contains and a value of 0 if the label is not present.

Table 1 Labeled Qur'an Dataset

Al-Baqarah (2:25)	Aqidah	Akhlak	Syariah	Ilmu	Alam Ghaib	Alam Dunia	Alam Akhirat
بشر الذي ءامن عمل صلحت أن هم جنة جري من تحت نهر كلما رزق من من ثمرة رزق قال هذا الذي رزق من قبل أتى ه متشبه هم في زوج مطهرة هم في خلد	1	1	0	0	0	1	1

Feature Extraction

The feature extraction method used in this study is TF-IDF (Term Frequency-Inverse Document Frequency). TF-IDF measures how frequently a term appears in a document (term frequency) and also considers the term's frequency across all documents in the corpus (inverse document frequency). Specifically, term frequency (TF) refers to how often a term appears in a document, while inverse document frequency (IDF) is the logarithm of the ratio of the total number of documents in the corpus to the number of documents containing the term.

This method is a statistical approach to assess the importance of a term within a document, helping to determine the document's characteristics by focusing on terms with high significance (Hidayati, D.C., Al Faraby, S., and Adiwijaya, A. 2020).

Classification

After the preprocessing and feature extraction processes, the data is now ready to be used as a classification dataset.

Latent Semantic Analysis

Latent Semantic Analysis (LSA) is a technique used to uncover hidden relationships between words in a document and to identify patterns in language usage. In the context of analyzing the Qur'an, LSA can help reveal hidden meanings or relationships between topics that are not immediately apparent. LSA operates by decomposing the term-document matrix into a simpler form, allowing us to identify latent dimensions that represent the underlying concepts in the text (Hidayati, Dian Chusnul, et al. 2013).

In this study, LSA is utilized to reduce the dimensionality of the Qur'anic text data before classification is performed using the KNN method. LSA is implemented through the use of Truncated SVD (Singular Value Decomposition), which extracts the principal components from the term-document matrix that represents the verses of the Qur'an. After the text data is reduced to a simpler dimensional form, normalization is conducted to ensure that the data is on a uniform scale.

K-Nearest Neighbor Method

The k-nearest Neighbor (KNN) algorithm is a method used to classify an object based on its proximity to existing training data (Rivki & Bachtiar, 2017). This algorithm

works by finding the *k* nearest training data points to the object to be classified and determining the class of that object based on the majority class of its *k* nearest neighbors. KNN is a type of supervised learning algorithm, where new objects are classified based on labels already present in the training data (Prakasawati, Abdillah, & Hadiana, 2017). The primary goal of the KNN algorithm is to classify new objects by leveraging existing attributes and training samples. For instance, if we want to classify a verse from the Qur'an into one of several categories, KNN will search for the most similar verses from the training dataset and then classify the new verse based on the categories of those closest verses. This process involves calculating the distance between the new object and every object in the training dataset, typically using metrics such as Euclidean distance.

In this study, several distance metrics, including Euclidean, Manhattan, and Cosine, are examined to determine the most effective metric for the multi-label topic classification of Qur'anic verses. The selection of the appropriate metric is crucial, as each metric measures similarity between verses in different ways. For example:

1. Euclidean distance: well-suited for data with uniform dimensions.
2. Manhattan distance is more sensitive to linear differences.
3. Cosine distance is more effective in handling sparse text data, where the orientation of the vector is more important than its magnitude.

By testing various metrics, this research aims to identify the most suitable approach for uncovering latent relationships between verses, thereby enhancing the accuracy of classification and the understanding of the Qur'an's complex structure.

The advantages of KNN include its simplicity and its ability to perform well across various types of data. However, KNN also has some drawbacks, such as its sensitivity to data scaling (Hidayati, Dian Chusnul, et al. 2013) and its decreased performance on large datasets, as each classification decision requires searching through the entire training dataset. (Fatiara, Agustian, & Afrianty, 2024). Despite these challenges, with the proper selection of the *k* parameter and the use of appropriate data preprocessing techniques, KNN can be a highly effective tool for classification tasks, including text analysis like the classification of Qur'anic verses.

Implementation of KNN dan LSA with GridSearchCV

Grid Search is a method used to find the optimal parameters to improve model performance by testing every combination of available hyperparameters. (Müller & Guido, 2016). This process includes a feature for performing cross-validation, which aims to evaluate the model's performance more comprehensively, considering that the model's performance can be affected by data splitting. The number of folds to be used can be determined through the cv parameter when initializing GridSearchCV from *sklearn*. If this parameter is not set, a default 5-fold cross-validation will be performed.

The Grid Search process starts by providing a 'Parameter Grid', which contains various parameter combinations to be tested. The dataset is then divided into two parts: 'Training Data' and 'Test Data'. 'Training Data' is used in the cross-validation process,

where the model is trained and evaluated on different subsets of data for each parameter combination in the 'Parameter Grid'.

After the cross-validation process is completed, the best combination of parameters, or 'Best Parameters', is selected based on the performance measured, usually using metrics such as the average F1-score. The model is then retrained using the 'Training Data' with the best parameters, resulting in the 'Retrained Model'. This model is finally evaluated on the 'Test Data' in the 'Final Evaluation' process to assess its overall performance.

The number of trials conducted in Grid Search is determined by the product of the number of folds in cross-validation and the number of values for each parameter being tested. For example, if the number of folds is 5, and the number of values for the n_components parameter in LSA is 3, k in KNN is 8, weights have 2 values, and metric has 3 values, then Grid Search will perform 720 trials. This process can take a considerable amount of time, but it can be accelerated by running trials in parallel using the n_jobs parameter in GridSearchCV. The default value of n_jobs is 1, but if you want to use all available processors, it can be set to -1.

In the end, Grid Search determines the 'Best Parameters' as the combination of parameters that provides the highest average F1 score across all folds for each parameter combination tested. The model, which has been retrained with the best parameters, is then evaluated on the 'Test Data' to ensure that its performance remains strong on data that it has not seen before.

Result and Discussion

In this test, a classification model combining LSA with KNN was implemented and then optimized using GridSearchCV. The process began with the initialization of the main components: A truncated SVD for performing LSA, which reduces the dimensionality of text features, a Normalizer to normalize the data, and a K Neighbors Classifier to perform the classification. All these components were combined into a pipeline.

Next, a parameter grid was defined for GridSearchCV, covering the number of LSA components (n components), the number of KNN neighbors (k-Neighbor), the type of weight (weights), and the distance metric (metric). This GridSearchCV was used to find the best parameter combination based on the F1 score (with average='micro') through 5-fold cross-validation on a multi-label model.

After the grid search was completed, the time taken to find the best parameters was calculated. The best model was then tested on the test data, with prediction time and F1 score also calculated and displayed. The goal was to find and evaluate the optimal parameter combination for a model that uses LSA as a preprocessing step before classification with KNN in a multi-label classification task.

Table 2 KNN without LSA

Topic	Precision	Recall	F1-Score

Aqidah	0.57	0.59	0.58
Akhlak	0.55	0.31	0.39
Syariah	0.55	0.31	0.40
Ilmu	0.57	0.23	0.33
Kisah	0.58	0.42	0.49
Alam Ghaib	0.59	0.15	0.24
Alam Dunia	0.62	0.47	0.54
Alam Akhira	0.53	0.33	0.40

Table 3 KNN with LSA, n components 100

Topic	Precision	Recall	F1-Score
Aqidah	0.53	0.52	0.53
Akhlak	0.46	0.21	0.29
Syariah	0.41	0.29	0.34
Ilmu	0.36	0.15	0.21
Kisah	0.52	0.41	0.46
Alam Ghaib	0.28	0.06	0.10
Alam Dunia	0.55	0.41	0.47
Alam Akhira	0.51	0.32	0.40

Table 4 KNN with LSA, n components 200

Topic	Precision	Recall	F1-Score
Aqidah	0.54	0.54	0.54
Akhlak	0.51	0.27	0.35
Syariah	0.43	0.29	0.35
Ilmu	0.41	0.21	0.28
Kisah	0.54	0.43	0.48
Alam Ghaib	0.58	0.08	0.14
Alam Dunia	0.56	0.45	0.50

Alam Akhira	0.53	0.32	0.40

Table 3 KNN with LSA, n components 300

Topic	Precision	Recall	F1-Score
Aqidah	0.58	0.52	0.55
Akhlak	0.57	0.26	0.36
Syariah	0.55	0.29	0.38
Ilmu	0.60	0.18	0.27
Kisah	0.62	0.42	0.50
Alam Ghaib	0.88	0.08	0.15
Alam Dunia	0.64	0.44	0.52
Alam Akhira	0.58	0.27	0.37

Table 2: Hamming Loss from each experiment

KNN with LSA n *components*	*Hamming Loss*
100	0.1657
200	0.1507
300	0.1466

The classification results indicate that the use of LSA in combination with KNN has an inconsistent impact on the model's performance. Without LSA, KNN produced a relatively stable and higher micro F1-score, suggesting that the model could effectively utilize the original text features for classification. However, when LSA was applied with a lower number of components (100 and 200), there was a decrease in the micro F1-score, indicating that some important information might have been lost during the dimensionality reduction process. For example, with 100 components, the Hamming loss was 0.1657, and with 200 components, the Hamming loss decreased to 0.1507. However, when the number of components increased to 300, the Hamming loss further decreased to 0.1466, indicating a continued improvement in information retention as the dimensionality was increased. While the model's performance improved with 300 LSA components, it still did not surpass the results of KNN without LSA. This suggests that while LSA is useful for capturing latent patterns in the text, there is a trade-off between

dimensionality reduction and information retention necessary for accurate classification. Therefore, in the context of Al-Qur'an text classification, selecting the optimal number of LSA components is crucial to achieving a balance between model complexity and classification accuracy.

Conclusion

From the comparison of classification using K-Nearest Neighbor (KNN) with and without Latent Semantic Analysis (LSA), it was found that the addition of LSA had varying impacts on the model's performance. KNN without LSA produced an average micro F1-score of 0.49, which was quite stable. However, the application of LSA with 100 components reduced the micro F1-score to 0.44, with a Hamming loss of 0.1657. Although increasing the number of LSA components to 300 improved the performance with a Hamming loss of 0.1466, the results still did not match KNN without LSA. While LSA helps identify latent patterns in the text, its effect on classification does not always enhance accuracy. Therefore, it is important to select the optimal number of LSA components or consider alternative methods such as Topic Modeling or Word Embeddings to achieve better results.

Bibliography

Dary, Mochamad Irfan, Bijaksana, Moc Arif, & Sa'adah, Siti. (2015). Analisis dan Implementasi Short Text Similarity dengan Metode Latent Semantic Analysis Untuk Mengetahui Kesamaan Ayat al-Quran. *EProceedings of Engineering*, *2*(3).

Fatiara, Nurul, Agustian, Surya, & Afrianty, Iis. (2024). Komparasi Metode K-Nearest Neighbors Dan Long Short Term Memory Pada Klasifikasi Terjemahan Al-Qur'an. *ZONAsi: Jurnal Sistem Informasi*, *6*(2), 332–345.

Muflihah, Filza Rahma, Lhaksmana, Kemas Muslim, & Bijaksana, Moch Arif. (2024). Centrality-Based Multilabel Neural Networks Classification of Qur'an Verse Topics. *2024 International Conference on Data Science and Its Applications (ICoDSA)*, 469–474. IEEE.

Müller, Andreas C., & Guido, Sarah. (2016). *Introduction to machine learning with Python: a guide for data scientists*. " O'Reilly Media, Inc."

Ningrum, Pipit Anggriati, Hukom, Alexandra, & Adiwijaya, Saputra. (2020). The Potential of Poverty in the City of Palangka Raya: Study SMIs Affected Pandemic Covid 19. *Budapest International Research and Critics Institute-Journal (BIRCI-Journal) Volume*, *3*, 1626–1634.

Pane, Reynaldi Ananda, Mubarok, Mohamad Syahrul, & Adiwijaya, Adiwijaya. (2018). Klasifikasi Multi Label Pada Topik Ayat Al-qur'an Terjemahan Bahasa Inggris Menggunakan Multinomial Naive Bayes. *EProceedings of Engineering*, *5*(1).

Prakasawati, Putri Eka, Abdillah, Gunawan, & Hadiana, Asep Id. (2017). Pola Kemampuan Anak Berdasarkan Rapor Menggunakan Text Mining Dan Klasifikasi Nearest Neighbor. *Semnasteknomedia Online*, *5*(1), 1–2.

Rivki, Muhammad, & Bachtiar, Adam Mukharil. (2017). Implementasi algoritma K-Nearest Neighbor dalam pengklasifikasian follower twitter yang menggunakan Bahasa Indonesia. *Jurnal Sistem Informasi*, *13*(1), 31–37.

Saputra, A., & Yadi, I. Z. (2020). Klasifikasi Ayat Al-Quran Terjemahan Menggunakan Metode Support Vector Machine Dan K-Nearest Neighbors. *Bina Darma Conference on Computer Science*, *2*(4), 449–466.

Syahraeni, Andi Syahraeni. (2017). Sejarah dalam Perspektif al-Qur'an. *Rihlah: Jurnal Sejarah Dan Kebudayaan*, *5*(1), 29–40.

Ta'a, Azman, Abdullah, Mohd Syazwan, Ali, Abdul Bashah Mat, & Ahmad, Muhammad. (2014). Themes-based classification for Al-Quran knowledge ontology. *2014 International Conference on Information and Communication Technology Convergence (ICTC)*, 89–94. IEEE.

Wirastri, Unang, Nurhaeni, Nani, & Syahreni, Elfi. (2017). Aplikasi Teori Comfort Kolcaba Dalam Asuhan Keperawatan Pada Anak Dengan Demam Di Ruang Infeksi Anak RSUPN Dr. Cipto Mangunkusumo. *Jurnal Kesehatan*, *6*(1), 27–32.