Indonesian Journal of Electrical Engineering and Computer Science
Vol. 42, No. 2, May 2026, pp. 609~618
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v42.i2.pp609-618

Natural language processing for report consolidation and matching based on latent semantic analysis and cosine similarity

Jeleen M. Mangubat[1], Ryndel Ventura Amorado[1,3], Lovely Rose T. Hernandez[1], Jennifer L. Marasigan[2]
[1]College of Informatics and Computing Sciences (CICS), Batangas State University the National Engineering University, Batangas, Philippines
[2]Computer Engineering Department, College of Engineering (CoE), Batangas State University the National Engineering University, Batangas, Philippines
[3]National Research Council of the Philippines, Taguig City, Philippines

Article Info

Article history:

Received Feb 5, 2025
Revised Mar 5, 2026
Accepted May 1, 2026

Keywords:

Cosine similarity
Intelligent system
Latent semantic indexing
Natural language processing
Web system

ABSTRACT

Consolidation of reports and matching of documents pose several challenges especially when dealing with large amounts of textual data. Thus, organizations are in need of intelligent systems that are capable of automating these processes, ensuring faster, more accurate analysis and retrieval of relevant information. This study applies Latent Semantic Indexing (LSI) and Cosine Similarity to automate the matching of gender-related issues, activities, and programs submitted by university offices. An intelligent web-based system was developed using Python and Django to implement these algorithms for report consolidation. Performance evaluation using accuracy, precision, recall, and F1-score demonstrated that the model correctly classified 90% of entries. A threshold sweep experiment further revealed that a similarity value of 0.51 provides the optimal decision boundary for identifying semantically similar instances. The findings confirm that LSI remains effective for low-resource institutional text analysis, enabling more efficient and accurate report consolidation.

Corresponding Author:

Ryndel Ventura Amorado
College of Informatics and Computing Sciences
Batangas State University, The National Engineering University
Batangas, Philippines
Email: ryndel.amorado@g.batstate-u.edu.ph

1. INTRODUCTION

Information is one of the most valuable resources for businesses and other organizations in the modern era. Correct processing and interpretation of this data can give the company a competitive edge through data-driven decision-making. However, numerous problems are making it difficult for the majority of enterprises to manage and analyze their data [1]-[3]. Some key factors include large textual datasets, limited tools, and unstructured properties. Additionally, information extraction is frequently difficult and time-consuming, which makes it difficult to combine and match related reports [4].

Dealing with various formats, diverse sources, and data quality presents additional difficulties [5], [6]. For instance, the organization frequently struggles to distinguish between duplicate and semantically identical content when required to submit and consolidate reports. Also, a major challenge lies in understanding the semantic relationships among various documents, as the meaning of the text is not always clearly articulated or straightforward, leading to differing interpretations. Moreover, studies suggested

different methodologies to address these challenges. Some solutions are proposed using Natural Language Processing (NLP) techniques for textual data [7]-[9].

It is proven that natural language processing is effective at deciphering and analyzing human language [6]. Information extraction, speech recognition, and text analysis are just a few of the uses for natural language processing. Latent semantic analysis (LSA) is one of the information retrieval methods used in NLP [10]. LSA detects patterns and relationships in datasets by analyzing the co-occurrence of words within a set of documents. This enhances the ability to capture the semantic structure of data, even when different terms are used. Latent semantic indexing (LSI) also reduces dimensionality, assisting in identifying latent concepts and patterns that would be challenging to detect through traditional keyword-based methods [11]-[15]. Additionally, Cosine Similarity is utilized alongside LSI to evaluate the similarity between two text or document vectors by comparing their angular distance [16], [17].

These technologies can help to intelligently consolidate reports and match documents by identifying and comparing similar content across extensive datasets. This can effectively assist organizations in accessing the pertinent information and may better utilize their data resources, in addition to supporting data-driven decision-making. However, existing institutional workflows heavily rely on manual inspection and keyword-based matching. These approaches often fail to identify semantically equivalent entries expressed using different terminologies, leading to redundancy and inconsistencies, resulting in more time to review. Current systems cannot capture latent relationships within heterogeneous administrative text or documents.

This study explores the application of LSI and Cosine Similarity in accurately matching semantically similar text. An intelligent system was developed for reviewing the submissions and reports for gender issues, activities, and programs by various offices. Thus, the performance of the model was calculated to determine its overall accuracy in document matching.

This paper provides the following key contributions:

1. Use LSI and Cosine Similarity to institutional report consolidation, validating the need beyond traditional information retrieval.
2. Develops an NLP pipeline for low-resource, heterogeneous administrative texts and documents, enabling accurate semantic similarity detection.
3. Introduces an empirically derived similarity threshold based on expert-validated evaluation, providing a quantitative decision boundary for classifying semantically similar entries.

2. RELATED WORKS

Recent research continues to demonstrate the effectiveness of LSI, which is one of the classical semantic approaches similar to TF-IDF and cosine similarity in document retrieval and semantic matching across diverse domains. The study of Hoque *et al.* [18] applied LSI with Singular Value Decomposition (SVD) and cosine similarity for Bangla document ranking, showing that LSI can capture hidden semantic relationships in low-resource languages effectively. Goyal and Sharma [19] concluded that cosine similarity consistently provides a good performance in comparing articles, showing the reliability in vector-space document matching. The study also compared various vectorization methods, including Bag of Words, TF-IDF, BERT, and the Universal Sentence Encoder.

Moreover, LSI remains widely used in text summarization. Sagum *et al.* [20] developed an LSA-based system to easily summarize Philippine Supreme Court documents and decisions, while [21] compared various methods in summarizing legal documents validated by domain experts and confirmed the competitiveness of cosine similarity. Rinjeni *et al.* [22] found that cosine similarity outperforms other distance measures, such as Jaccard similarity, in identifying semantically related academic texts. Collectively, existing research highlights that both classical vector-space models (LSI, TF-IDF, cosine similarity) and modern neural architectures remain valuable tools for document similarity, classification, and summarization, supporting their application in domains requiring semantic alignment and automated text matching.

Despite the extensive use of various semantic models in retrieval and summarization tasks, their application in institutional reporting, particularly for consolidating gender-related issues, activities, and programs, remains limited. Existing workflows still depend heavily on manual review, which cannot reliably detect semantic equivalence across heterogeneous administrative texts. This study addresses this gap by applying LSI and cosine similarity to automate semantic matching within institutional report consolidation.

3. RESEARCH METHOD

The study proposes methods shown in Figure 1 for calculating the similarity between the reports using the vector space model.

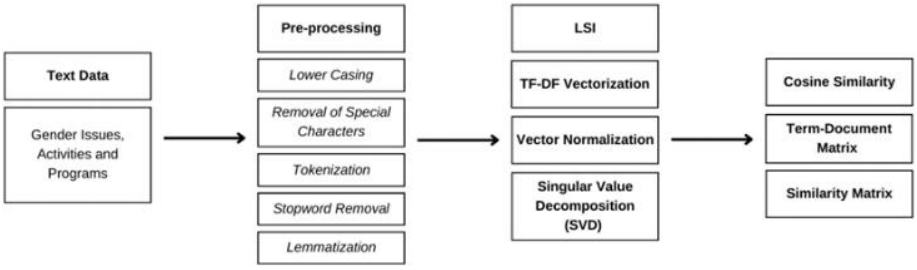


Figure 1. Process diagram

3.1. Data collection

Data from various assessment tools of the gender and development office (GAD) in one of the well-known universities in the Philippines were collected. The collection contains different attributes, as shown in Table 1, like gender issues, activities, and programs.

Table 1. Dataset description

Attributes	Description
Gender Issue and/or GAD mandate	Description of gender issues or GAD mandates
Cause of the gender Issue	Provided additional contextual text related to the gender issues
GAD result statement/ GAD objective	Describes the desired outcomes or objectives related to GAD.
Relevant agency MFO/PAP	Program activities and projects category
GAD activity	Descriptions of the activities.
Output performance indicators and target	Performance indicators and targets of the activity
GAD budget	Allocated budget
Source of budget	Fund source
Responsible unit/office	Assigned office to conduct the activity
Campus	Campus of the responsible unit/office

3.2. Pre–processing

The experiment was conducted using Python in Google Colab. As shown in Figure 2, the raw data were filtered to eliminate the irrelevant and duplicate datasets. This process enables the system to manipulate datasets into their proper forms. Included in the pre-processing stage is the lemmatization, which reduces words to their dictionary form. To filter words that are not significant to the context, stop word is used words, like are a, an, is, are, the, etc, are removed. During the tokenization process, patterns will emerge in raw datasets. Then, it will be broken down into a stream of terms to manage them easily. After the pre-processing stage, the output will be passed to the Matrix Parser for further processing.

```
# Stopwords
stop_words = set(stopwords.words('english'))
custom_stopwords = {
    "gad", "gender", "development", "plan", "budget",
    "activity", "activities", "program", "programs",
    "office", "unit", "campus", "university",
    "statement", "result", "mandate", "issue", "cause",
    "based", "conduct", "conducted",
    "number", "form", "forms",
    "table", "year", "target", "targets",
    "responsible"
}
stop_words.update(custom_stopwords)

# Vocabulary tracking
raw_vocab = set()
processed_vocab = set()

lemmatizer = WordNetLemmatizer()

def preprocess(text):
    tokens = word_tokenize(text.lower())
    raw_vocab.update([t for t in tokens if t.isalpha()])
    tokens = [t for t in tokens if t.isalpha() and t not in stop_words]
    tokens = [lemmatizer.lemmatize(t) for t in tokens]
    processed_vocab.update(tokens)
    return " ".join(tokens)

processed_docs = [preprocess(doc) for doc in documents]

print("RAW VOCAB SIZE:", len(raw_vocab))
print("PROCESSED VOCAB SIZE:", len(processed_vocab))
print(
    "VOCABULARY REDUCTION:",
    (1 - len(processed_vocab) / len(raw_vocab)) * 100,
    "%"
)
```

Figure 2. Pre-processing of GAD PAPs and activities

3.3. Latent semantic indexing (LSI)

After pre-processing, the text will be converted into a numerical vector, such as TF-IDF, which is based on the frequency and importance relative to the corpus, as shown in Figure 3. To reduce the dimensionality and identify patterns of the TF-IDF matrix, singular value decomposition (SVD) was used.

Latent Semantic Indexing was applied using SVD to project the TF-IDF matrix into a lower-dimensional latent space. Several values of the number of components were evaluated to determine the optimal configuration. To determine the optimal number of latent dimensions for the LSI model, the singular values obtained from the SVD decomposition of the term-document matrix were analyzed. The variance for a given number of components *k* was computed by dividing the sum of the squared singular values of the first *k* components by the sum of all squared singular values. This ratio indicates how much semantic information is preserved as dimensionality increases.

```
import matplotlib.pyplot as plt
from sklearn.decomposition import TruncatedSVD
from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = TfidfVectorizer(max_features=5000)
tfidf_matrix = vectorizer.fit_transform(processed_docs)

print("TF-IDF Matrix Shape:", tfidf_matrix.shape)

k_values = [50, 100, 150, 200, 250, 300]
explained_variances = []

for k in k_values:
    svd = TruncatedSVD(n_components=k)
    svd.fit(tfidf_matrix)
    explained_variances.append(svd.explained_variance_ratio_.sum())
```

Figure 3. Latent semantic indexing

3.4. Cosine similarity

The Cosine Similarity function shown in Figure 4 was applied to determine the similarity between two vectors of an inner product space. It is calculated by taking the cosine of the angle between two vectors and determining if this is pointing in nearly the same direction. It is commonly used to measure document similarity in text analysis [23]. The cosine similarity then measures the similarity between the two vectors using in (1):

$$\cos\theta = \frac{\vec{A}\cdot\vec{B}}{\|\vec{A}\|\|\vec{B}\|} \tag{1}$$

Where

$$\vec{A}\cdot\vec{B} = \sum_{i=1}^{n} A_i \cdot B_i \quad \|\vec{A}\| = \sqrt{\sum_{i=1}^{n} A_i^2} \quad \|\vec{B}\| = \sqrt{\sum_{i=1}^{n} B_i^2} \tag{2}$$

```
from sklearn.metrics.pairwise import cosine_similarity
terms = vectorizer.get_feature_names_out()

def top_terms(component, n=10):
    idx = component.argsort()[::-1][:n]
    return [terms[i] for i in idx]

for i in range(10):
    print(f"Topic {i+1}: ", top_terms(svd.components_[i]))

similarity_matrix = cosine_similarity(lsi_matrix)
similarity_matrix[:5, :5]
```

Figure 4. Cosine similarity index

3.5. Development of the system

To develop the intelligent system, various software tools were utilized. The system was developed using Python and Django to easily implement LSI and Cosine Similarity. SQLite was used for data collection and retrieval due to its lightweight and serverless architecture. Bootstrap, jQuery, and JavaScript were used for front-end development to enhance responsiveness.

3.6. Performance evaluation

Performance metrics such as accuracy, precision, recall, and F1-score were used to evaluate the performance of the model. These metrics are computed by comparing the number of gender issues, activities, and programs that are accurately classified as similar (1) and dissimilar (0) to the judgment made by the office expert [24], [25]. Accuracy, Precision, Recall, and F1-score will be computed based on the given in (3)-(6) [26].

$$Accuracy = \frac{Number\ of\ Correct\ Prediction}{Total\ Number\ of\ Prediction} \quad (3)$$

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positive} \quad (4)$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (5)$$

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (6)$$

4. RESULTS AND DISCUSSION

This section shows the implementation of LSI and Cosine Similarity to the web-based system for activity and program matching and report consolidation.

4.1. Results

4.1.1. Implementation of LSI

Figure 5 shows how text preprocessing affects the vocabulary size of the dataset. The raw corpus had 765 unique words. After applying preprocessing steps like lowercasing, removing stop words, and filtering for specific domains, the count dropped to 697 unique words. This is an 8.9% reduction in vocabulary size. Figure 6 shows the explained variance from different numbers of LSI components. At k = 50, the model captures about 86% of the variance. When the dimensionality increases to k = 100, it shows a significant improvement, capturing almost 100% of the semantic variance. After k = 150, the explained variance levels off, with values consistently nearing 1.0.

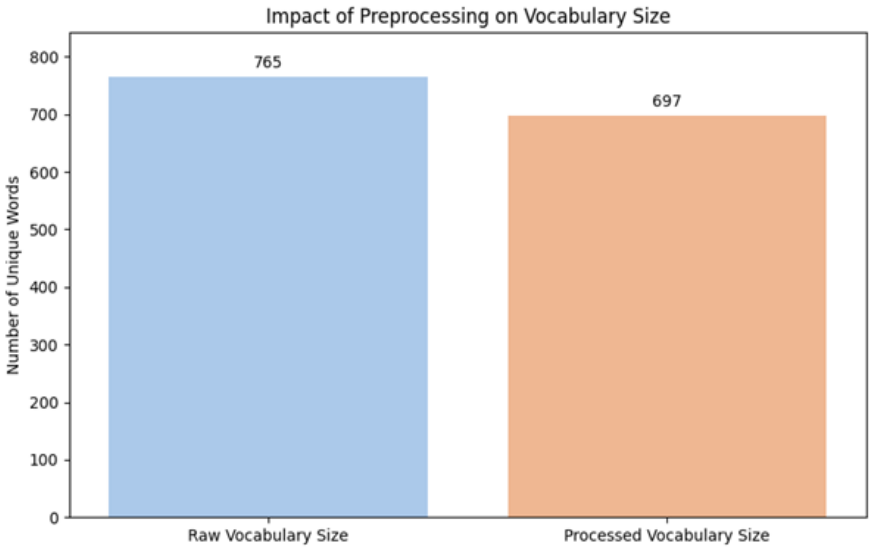


Figure 5. Preprocessing result on vocabulary size

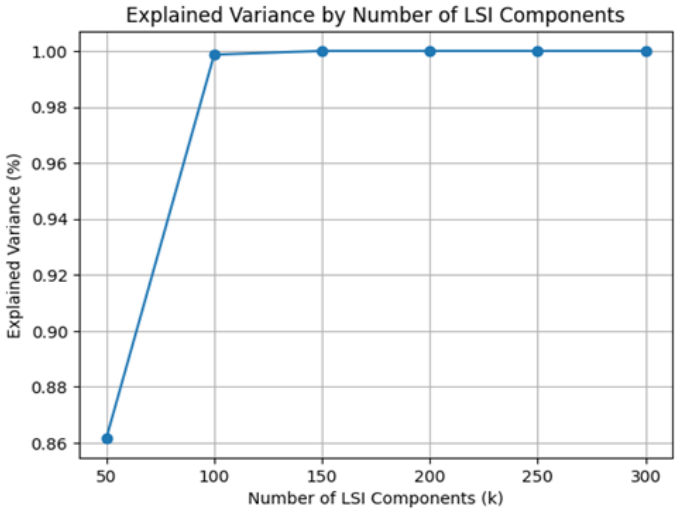


Figure 6. Variance by number og LSI components

4.1.2. Utilization of Cosine Similarity

Figure 7 presents the cosine similarity matrix derived from the LSI-transformed vectors. All diagonal values show perfect similarity = 1.0, while off-diagonal values vary across the matrix. Several blocks of moderate similarity from 0.3-0.6 appear in the lower-right portion, indicating natural clusters of related issue–activity entries. Most other pairwise similarities approach zero, suggesting that most entries in the dataset are semantically distinct.

Figure 8 shows the confusion matrix of precision, which measures the proportion of predicted positives that were correct, and was found to be 98% highlighting the model's effectiveness in identifying positive cases. The recall, which evaluates the model's ability to identify all actual positive instances, was 82%. The F1-score has an 89%, providing a single measure that combines the model's precision and recall performance. This score shows that the model performs well in correctly identifying both positive and negative instances while minimizing false positives and false negatives.

To determine an appropriate cutoff for classifying similar entries, a threshold sweep experiment was conducted. Similarity thresholds from 0.50 to 0.95 were evaluated in increments of 0.01, and the model's predictions were compared with expert-annotated labels. For each threshold, accuracy, precision, recall, and F1-score were computed.

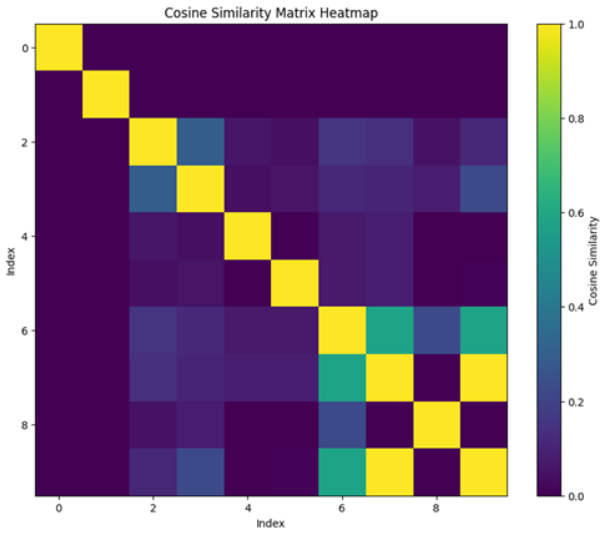


Figure 7. Cosine similarity matrix heatmap

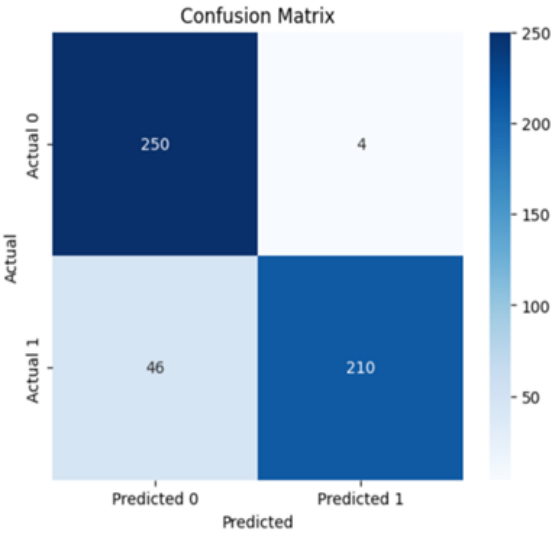


Figure 8. Confusion matrix

As shown in Figure 9, the highest F1-score occurs at a threshold of 0.51, indicating the most balanced trade-off between precision and recall. At this value, the model maintains a high precision while minimizing the loss of true positive matches. Thus, 0.51 was adopted as the optimal decision boundary, replacing the initially assumed value of 0.70.

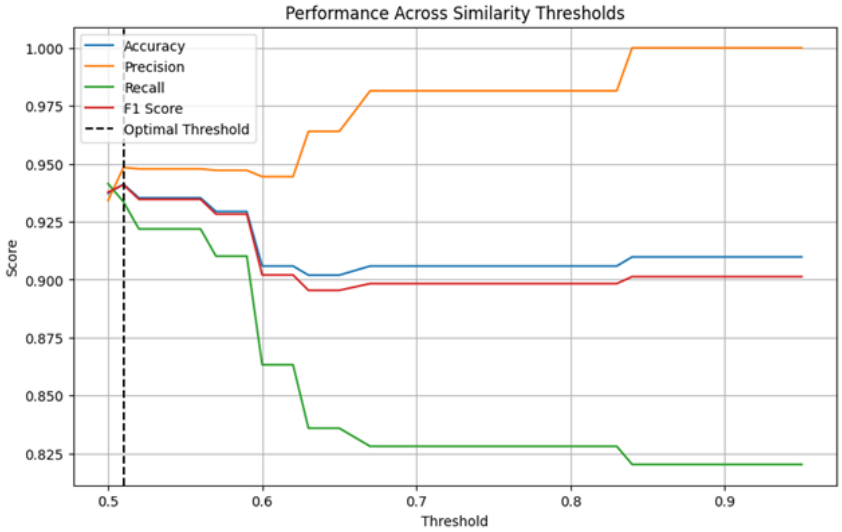
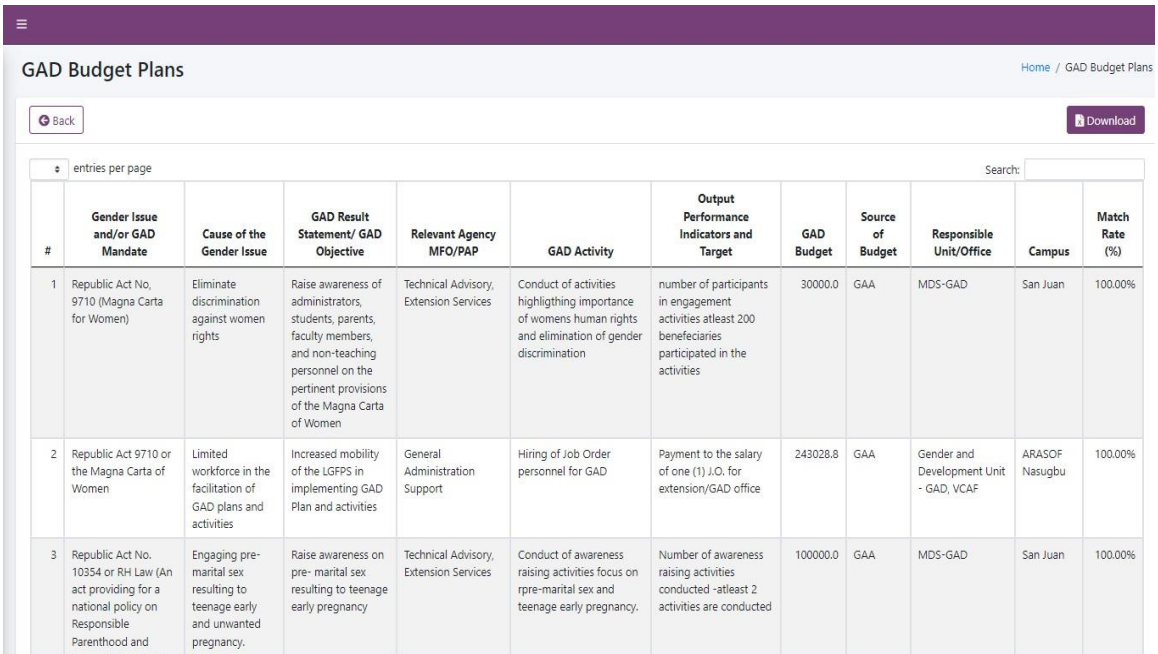


Figure 9. Performance across similarity thresholds

4.1.3. Code and system implementation of LSI and cosine similarity

An intelligent report consolidation and program matching system was also developed. The system includes interfaces for the users to enter their reports and programs for consolidations. Figure 10 shows the actual implementation of LSI and Cosine Similarity in the developed system. The system will sort and display the list of similar activities and their percentage similarity or match rate to other activities available in the database. This information will be displayed on the office administrator's side for easy consultation and download.



#	Gender Issue and/or GAD Mandate	Cause of the Gender Issue	GAD Result Statement/ GAD Objective	Relevant Agency MFO/PAP	GAD Activity	Output Performance Indicators and Target	GAD Budget	Source of Budget	Responsible Unit/Office	Campus	Match Rate (%)
1	Republic Act No, 9710 (Magna Carta for Women)	Eliminate discrimination against women rights	Raise awareness of administrators, students, parents, faculty members, and non-teaching personnel on the pertinent provisions of the Magna Carta of Women	Technical Advisory, Extension Services	Conduct of activities highligthing importance of womens human rights and elimination of gender discrimination	number of participants in engagement activities atleast 200 benefeciaries participated in the activities	30000.0	GAA	MDS-GAD	San Juan	100.00%
2	Republic Act 9710 or the Magna Carta of Women	Limited workforce in the facilitation of GAD plans and activities	Increased mobility of the LGFPS in implementing GAD Plan and activities	General Administration Support	Hiring of Job Order personnel for GAD	Payment to the salary of one (1) J.O. for extension/GAD office	243028.8	GAA	Gender and Development Unit - GAD, VCAF	ARASOF Nasugbu	100.00%
3	Republic Act No. 10354 or RH Law (An act providing for a national policy on Responsible Parenthood and	Engaging pre-marital sex resulting to teenage early and unwanted pregnancy.	Raise awareness on pre- marital sex resulting to teenage early pregnancy	Technical Advisory, Extension Services	Conduct of awareness raising activities focus on rpre-marital sex and teenage early pregnancy.	Number of awareness raising activities conducted -atleast 2 activities are conducted	100000.0	GAA	MDS-GAD	San Juan	100.00%

Figure 10. System interface for LSI and cosine similarity in GAD system

4.2. Discussions

The study shows that LSI and cosine similarity perform well in identifying semantically related GAD issues, activities, and programs from various institutional reports. With an accuracy of 90%, precision of 98%, recall of 82%, and an F1-score of 89%, the model is able to differentiate similar from non-similar entries even when the wordings and phrasings are different across submissions. This is helpful for administrative offices, which often struggle to manually examine large volumes of text where important similarities are hidden by inconsistent terminology.

The preprocessing stage reduced the vocabulary from 765 to 697 terms or an 8.9% decrease, showing that the corpus retained its semantic depth while minimizing noise. This modification allows latent relationships to show more clearly without sacrificing domain-specific language. The explained-variance analysis reinforces that 50 components already captured 86% of the semantic variance, while 100 components provided near-complete coverage before the curve flattened. This pattern confirms that k = 100 strikes an effective balance between dimensional compactness and semantic richness.

Findings from the cosine similarity heatmap further support this conclusion. Distinct clusters appeared, indicating that some reports contain redundant content, while entries showing near-zero similarity reflect genuinely unique submissions. Pairs with moderate similarity reveal partial thematic overlaps, illustrating how automated semantic grouping can assist in consolidating GAD reports more coherently.

The system evaluation also demonstrated strong practical value. The prototype interface processed user queries efficiently, generated similarity scores, and ranked related entries, enabling administrators to identify redundancies and improve consistency across reports. Notably, the optimization process found that a threshold of 0.51, rather than the commonly used 0.70, offered the best balance between precision and recall. This emphasizes the importance of empirical tuning rather than relying on conventional defaults.

These results strengthen the findings of Hoque *et al.* [18] and Goyal and Sharma [19], who highlighted the effectiveness of LSI and cosine similarity in low-resource and specialized text environments. This study extends their work by demonstrating how classical semantic techniques can be applied to institutional reporting, an area where automation remains relatively underdeveloped.

Overall, the findings underscore that classical semantic models, when properly optimized, still serve as powerful tools for analyzing low-resource institutional texts. With cosine similarity and an empirically validated threshold of 0.51, LSI offers a dependable and scalable approach for automating the consolidation of GAD reports and shows strong potential for broader applications, including accreditation documentation, ISO compliance reports, and other administrative datasets.

5. CONCLUSION

The results of this study demonstrate the strong capability of LSI combined with cosine similarity in automating document matching and consolidation. With an accuracy of 90%, the model shows considerable promise in identifying semantically related documents, even when different offices use varied or inconsistent terminology. Its high precision (98%) and solid recall (82%) indicate that the system can reliably classify relevant instances while keeping both false positives and false negatives to a minimum. The resulting F1-score of 89% reflects a well-balanced performance, underscoring the model's suitability for organizations that must manage large volumes of textual data. Overall, the findings affirm that classical semantic models remain highly effective for low-resource administrative corpora and can significantly enhance institutional workflows.

The study also confirms the usefulness of LSI in addressing the challenges faced by different offices when identifying gender-related concerns embedded in narrative reports. LSI's dimensionality reduction preserves meaningful semantic patterns, improving the accuracy and dependability of the sorting process. When paired with cosine similarity, the method performs particularly well in measuring relationships among GAD topics. By evaluating the cosine distance between vector representations, the system enables scalable comparison and effective grouping of related concerns, helping uncover underlying connections across varied submissions.

Despite these strong results, some limitations remain. The comparatively lower recall suggests that a portion of valid matches, especially those expressed in more abstract or nuanced language, were not fully captured. Because LSI has difficulty modeling deeper contextual relationships, more advanced approaches such as BERT or other transformer-based embeddings may help address this gap. For future work, the study recommends: (1) conducting comparative experiments between LSI and transformer-based semantic models, (2) expanding the dataset to include additional campuses or reporting cycles, and (3) integrating domain-specific ontologies to better represent terminology related to Gender and Development.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the support given by Batangas State University, the National Engineering University, and the College of Informatics and Computing Sciences.

REFERENCES

[1] A. Saidi, S. Ben Othman, M. Dhouibi, and S. Ben Saoud, "FPGA-based implementation of classification techniques: A survey," *Integration*, vol. 81, pp. 280–299, Nov. 2021, doi: 10.1016/j.vlsi.2021.08.004.

[2] T. L. Chasupa and W. Paireekreng, "The framework of extracting unstructured usage for big data platform," in *2021 2nd International Conference on Big Data Analytics and Practices (IBDAP)*, IEEE, Aug. 2021, pp. 90–94. doi: 10.1109/IBDAP52511.2021.9552131.

[3] L. Guo, F. Shi, and J. Tu, "Textual analysis and machine leaning: crack unstructured data in finance and accounting," *Journal of Finance and Data Science*, vol. 2, no. 3, pp. 153–170, Sep. 2016, doi: 10.1016/j.jfds.2017.02.001.

[4] S. Alias, M. S. Sainin, T. S. Fun, N. Daut, and T. L. Sheng, "Unsupervised text feature extraction for academic chatbot using constrained FP-Growth," *ASM Science Journal*, vol. 14, pp. 1–11, Apr. 2021, doi: 10.32802/asmscj.2020.576.

[5] I. Bifulco, S. Cirillo, C. Esposito, R. Guadagni, and G. Polese, "An intelligent system for focused crawling from big data sources," *Expert Systems with Applications*, vol. 184, p. 115560, Dec. 2021, doi: 10.1016/j.eswa.2021.115560.

[6] Supriyono, A. P. Wibawa, Suyono, and F. Kurniawan, "Advancements in natural language processing: Implications, challenges, and future directions," *Telematics and Informatics Reports*, vol. 16, p. 100173, Dec. 2024, doi: 10.1016/j.teler.2024.100173.

[7] N. H. Baqer, A. T. Sadiq, and Z. H. Ali, "Enhancement of sentiment analysis in hotel reviews through latent semantic indexing and convolutional neural networks," *Ingenierie des Systemes d'Information*, vol. 28, no. 6, pp. 1613–1618, Dec. 2023, doi: 10.18280/isi.280618.

[8] P. Figuera and P. García Bringas, "Revisiting probabilistic latent semantic analysis: extensions, challenges and insights," *Technologies*, vol. 12, no. 1, p. 5, Jan. 2024, doi: 10.3390/technologies12010005.

[9] M. Voggenreiter, P. Schneider, and A. Gulraiz, "Aggregating industrial security findings with semantic similarity-based techniques," in *Signals and Communication Technology*, vol. Part F2085, 2024, pp. 121–139. doi: 10.1007/978-3-031-44260-5_7.

[10] A. Subasi, "Introduction," in *Practical Machine Learning for Data Analysis Using Python*, Elsevier, 2020, pp. 1–26. doi: 10.1016/b978-0-12-821379-7.00001-1.

[11] A. Kontostathis and W. M. Pottenger, "A framework for understanding latent semantic indexing (LSI) performance," *Information Processing and Management*, vol. 42, no. 1 SPEC. ISS, pp. 56–73, Jan. 2006, doi: 10.1016/j.ipm.2004.11.007.

[12] Sukri, N. A. Samsudin, E. Fadzrin, S. K. A. Khalid, and L. Trisnawati, "Job matching analysis by latent semantic indexing enhanced on multilingual word meanings," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 37, no. 1, pp. 434–442, Jan. 2025, doi: 10.11591/ijeecs.v37.i1.pp434-442.

[13] F. Horasan, "Latent semantic indexing-based hybrid collaborative filtering for recommender systems," *Arabian Journal for Science and Engineering*, vol. 47, no. 8, pp. 10639–10653, Aug. 2022, doi: 10.1007/s13369-022-06704-w.

[14] R. Kurniawan, I. Daqiqil Id, and Z. Indra, "Analyzing student perspectives on learning experience using latent semantic indexing algorithm," in *2023 6th International Conference of Computer and Informatics Engineering (IC2IE)*, IEEE, Sep. 2023, pp. 287–291. doi: 10.1109/IC2IE60547.2023.10331300.

[15] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391–407, Sep. 1990, doi: 10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9.

[16] V. Kotu and B. Deshpande, "Classification," in *Data Science*, Elsevier, 2019, pp. 65-163. doi: 10.1016/B978-0-12-814761-0.00004-6.

[17] V. Kotu and B. Deshpande, "Recommendation engines," in *Data Science*, Elsevier, 2019, pp. 343-394. doi: 10.1016/b978-0-12-814761-0.00011-3.

[18] M. N. Hoque, R. Islam, and M. S. Karim, "Information retrieval system in bangla document ranking using latent semantic indexing," in *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, IEEE, May 2019, pp. 1–5. doi: 10.1109/ICASERT.2019.8934837.

[19] K. Goyal and M. Sharma, "Comparative analysis of different vectorizing techniques for document similarity using cosine similarity," in *2022 Second International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE)*, IEEE, Dec. 2022, pp. 1–5. doi: 10.1109/ICATIECE56365.2022.10046766.

[20] R. A. Sagum, P. A. C. Clacio, R. E. R. Cayetano, and A. D. F. Lobrio, "Philippine court case summarizer using latent semantic analysis," *Procedia Computer Science*, vol. 227, pp. 474–481, 2023, doi: 10.1016/j.procs.2023.10.548.

[21] M. Akter, E. Çano, E. Weber, D. Dobler, and I. Habernal, "A comprehensive survey on legal summarization: challenges and future directions," *ACM Computing Surveys*, vol. 58, no. 7, pp. 1–32, May 2026, doi: 10.1145/3776586.

[22] T. P. Rinjeni, A. Indriawan, and N. A. Rakhmawati, "Matching scientific article titles using cosine similarity and jaccard similarity algorithm," *Procedia Computer Science*, vol. 234, pp. 553–560, 2024, doi: 10.1016/j.procs.2024.03.039.

[23] F. Al-Anzi and D. Abuzeina, "Enhanced latent semantic indexing using cosine similarity measures for medical application," *International Arab Journal of Information Technology*, vol. 17, no. 5, pp. 742–749, Sep. 2020, doi: 10.34028/iajit/17/5/7.

[24] S. S. Kamaruddin, Y. Yusof, N. A. A. Bakar, M. A. Tayie, and G. A. A. J. Alkubaisi, "Graph-based representation for sentence similarity measure: A comparative analysis," *International Journal of Engineering and Technology*, vol. 7, no. 2, pp. 32–35, Apr. 2018, doi: 10.14419/ijet.v7i2.14.11149.

[25] R. Singh and S. Singh, "Text similarity measures in news articles by vector space model using NLP," *Journal of The Institution of Engineers (India): Series B*, vol. 102, no. 2, pp. 329–338, Apr. 2021, doi: 10.1007/s40031-020-00501-5.

[26] K. N. Singh, S. D. Devi, H. M. Devi, and A. K. Mahanta, "A novel approach for dimension reduction using word embedding: an enhanced text classification approach," *International Journal of Information Management Data Insights*, vol. 2, no. 1, p. 100061, Apr. 2022, doi: 10.1016/j.jjimei.2022.100061.

BIOGRAPHIES OF AUTHORS

Jeleen M. Mangubat earned a Master of Science in Information Technology at Batangas State University, the National Engineering University, and a Bachelor of Science in Information Technology at Lyceum of the Philippines—Batangas. She is a full-time faculty member in the BS Information Technology Department and a faculty researcher in the field of web development and programming. She has demonstrated a strong commitment to student engagement and leadership development. She was the organization adviser for the Integrated Information Technology Students Society (IINTESS) in 2022 and is currently the adviser for the Junior Philippine Computer Society (JPCS) – Alangilan Chapter. She can be contacted at email: jeleen.mangubat@g.batstate-u.edu.ph.

Dr. Ryndel Ventura Amorado is a faculty member of the College of Informatics and Computing Sciences of Batangas State University, the National Engineering Technology, Lipa Campus. He serves as College Dean on the same college and concurrently a Faculty Researcher. He obtained his Doctorate Degree in Information Technology in 2020 at the Technological Institute of the Philippines and Master's Degree in Information Technology in 2012 from Batangas State University. He has several publications in Machine Learning, Information Security and Cryptography, which are his primary research interest. In addition, he also counts network design, data mining, and e-learning as research areas. He can be contacted at email: ryndel.amorado@g.batstate-u.edu.ph.

Lovely Rose T. Hernandez is the Campus Head of the Gender and Development Office at Batangas State University, the National Engineering Technology, Alangilan Campus, and a Computer Science faculty member at the College of Informatics and Computing Sciences at the same university. She holds a Master of Science in Computer Science and has published a paper titled Performance Analysis of Lightweight Vision Transformers and Deep Convolutional Neural Networks in Detecting Brain Tumors in MRI Scans: An Empirical Approach. Her research interests include Artificial Intelligence, Natural Language Processing, and Data Science. She can be contacted at email: lovelyrose.hernandez@g.batstate-u.edu.ph.

Engr. Jeniffer L. Marasigan is a Professional Computer Engineering and currently serving as Computer Engineering Faculty Member at College of Engineering, Batangas State University the National Engineering University. She holds Master of Science in Computer Engineering and BS Computer Engineering at the same university. Her research interest are Smart Technologies, IoT Systems and Networking. She can be contacted at email: jennifer.marasigan@g.batstate-u.edu.ph.