Journal of Applied Informatics and Computing (JAIC)
Vol.10, No.4, August 2026, pp. 3884~3890
e-ISSN: 2548-6861

Comparative Analysis of Siamese BiLSTM and IndoBERT for Semantic Textual Similarity Detection in Functional Scientific Papers of Meteorology, Climatology, and Geophysics Officers at BMKG

Dhedy Listyawan[1*], Ahmad Musyafa[2], Choirul Basir[3]
[1,2]Master of Informatics Engineering Program, Universitas Pamulang
[3]Department of Mathematics, Faculty of Mathematics and Natural Science, Universitas Pamulang
dhedy.listyawan@bmkg.go.id [1,*], ahmad.musyafa@unpam.ac.id [2], dosen02278@unpam.ac.id [3]

Article Info

Article history:
Received 2026-06-30
Revised 2026-08-03
Accepted 2026-08-12

Keywords:
Semantic Textual Similarity,
Siamese BiLSTM,
IndoBERT,
Document Similarity,
PMG BMKG

ABSTRACT

The competency assessment process for Meteorology, Climatology, and Geophysics (PMG) functional officers at BMKG requires manual review of Scientific Paper (KTI) similarity, a task that is time-consuming, prone to subjectivity, and increasingly burdensome as submission volume grows. While Semantic Textual Similarity (STS) research has advanced considerably for high-resource languages, empirical evidence for Indonesian-language technical documents in specialized scientific domains remains limited, and prior work has not established which neural architecture is preferable under such data-constrained conditions. This study addresses that gap by empirically comparing two Siamese Network architectures, Siamese BiLSTM and IndoBERT, for automatic STS detection on 87 PMG KTI documents, yielding 3,741 document pairs automatically labeled using the 90th percentile of TF-IDF cosine similarity scores and validated against manual annotation (Cohen's Kappa κ=0.82). Siamese BiLSTM employs Word2Vec embeddings with Focal Loss, while IndoBERT fine-tunes the pretrained indobert-base-p1 model with Contrastive Loss; both apply class weighting to address the 90:10 class imbalance, with classification thresholds independently calibrated via grid search. Evaluated on 1,123 held-out test pairs, Siamese BiLSTM achieves an F1-Score of 64.84% (Accuracy 91.99%, Precision 58.04%, Recall 73.45%) at threshold 0.60, outperforming IndoBERT's F1-Score of 57.73% (Accuracy 89.05%, Precision 47.19%, Recall 74.34%) at threshold 0.935, a difference confirmed statistically significant by McNemar's test (χ^2=7.6992; p=0.0055). This result runs counter to the common assumption that Transformer-based models universally outperform recurrent architectures, suggesting that smaller, domain-tuned embeddings can be more effective under limited-data, domain-specific conditions. The better-performing Siamese BiLSTM model, requiring only 16.77 MB with no GPU dependency, was deployed as a Streamlit web application, enabling the BMKG PMG Assessment Team to perform similarity detection quickly and consistently within their existing workflow.

I. INTRODUCTION

The Badan Meteorologi, Klimatologi, dan Geofisika (BMKG, Indonesian Agency for Meteorology, Climatology, and Geophysics) conducts a Competency Assessment for its Meteorology, Climatology, and Geophysics (PMG) functional officers as part of the promotion process. One key assessment component is the Scientific Paper (*Karya Tulis Ilmiah*, KTI), which must be checked for similarity against other papers to prevent duplication and plagiarism. Manual review by the Assessment Team is time-consuming and prone to subjectivity, especially as the number of submitted papers increases each assessment period while the number of reviewers remains limited.

Conventional keyword-based or string-matching approaches often fail to detect similarity at the meaning level

when two documents discuss the same topic but use different sentence structures and word choices (paraphrasing). *Semantic Textual Similarity* (STS) based on machine learning offers a solution by measuring similarity of meaning through high-dimensional vector representations (*embeddings*), rather than literal word matching. *Siamese Network* architectures are a popular approach for STS, consisting of two identical sub-networks that share weights to learn comparable representations of two inputs simultaneously [1]. Two widely studied variants are Siamese Long Short-Term Memory (LSTM) [1], [2] and Siamese architectures built on pretrained Transformer models such as BERT [3], [4].

A comprehensive survey by Li *et al.* [4] documents that Siamese networks have been successfully applied across classification, regression, and few-shot learning tasks, while Transformer-based sentence embeddings such as Sentence-BERT [5] have generally outperformed recurrent architectures on major English-language STS benchmarks. However, this body of evidence is overwhelmingly drawn from high-resource languages and general-domain corpora; empirical comparisons for *Indonesian-language* technical documents in specialized scientific domains remain scarce, and existing Indonesian STS studies have notable limitations. Krisnawati *et al.* [6] applied Universal Sentence Encoder with FAISS for Indonesian-English bilingual similarity detection but evaluated on only 116 documents without a domain-specific technical corpus. Viji and Revathy [8] combined BERT with graph convolutional networks for document-level near-duplicate detection, but their approach was designed for general English text deduplication rather than domain-specific semantic assessment. Neither study directly addresses the combination of factors present in the BMKG PMG setting: a technical, terminology-dense Indonesian corpus, a severely limited number of source documents, and a need for models deployable without GPU infrastructure. Consequently, it remains an open empirical question whether Transformer-based architectures retain their advantage over recurrent Siamese architectures under such data-constrained, domain-specific conditions — a gap this study directly addresses.

This study aims to: (1) compare the performance of Siamese BiLSTM and IndoBERT for STS detection on BMKG PMG scientific papers based on Accuracy, Precision, Recall, and F1-Score; (2) test the statistical significance of the performance difference using McNemar's test; and (3) deploy the best-performing model as a web application usable directly by the BMKG PMG Assessment Team.

II. METHOD

This research follows five stages: (1) data collection and preprocessing, (2) pair formation and automatic labeling, (3) separate training of both model architectures, (4) evaluation and statistical significance testing, and (5) deployment of the best-performing model as a web application.

A. Dataset and Labeling

The dataset consists of 87 BMKG PMG KTI documents in PDF format, spanning topics across meteorology, climatology, geophysics, and instrumentation, with an average length of approximately 4,200 words per document after preprocessing. Text was extracted using the *pdfplumber* library. Preprocessing includes lowercasing, non-alphabet character removal, whitespace normalization, and Indonesian stopword removal using *PySastrawi*. All documents were combinatorially paired without repetition, yielding C(87,2)=3,741 unique document pairs.

The relatively small number of source documents (87) is a direct consequence of the operational setting: it reflects the actual number of KTI submissions in a single BMKG PMG competency assessment period, rather than a sampling choice. While this scale is modest relative to typical deep-learning corpora, the combinatorial pairing strategy expands it to 3,741 pairs, and the results in Section III indicate it was sufficient to train Siamese BiLSTM (4.4M parameters) to convergence, whereas the larger IndoBERT (124M parameters) showed comparatively weaker generalization — itself evidence, discussed further in Section III-D, that dataset scale interacts with model capacity in this setting.

Each pair was automatically labeled based on the *cosine similarity* score of the Term Frequency-Inverse Document Frequency (TF-IDF) [7] representation, where the weight of term *t* in document *d* is:

$$w(t,d) = tf(t,d) \times \log(N / df(t)) \quad (1)$$

with *tf(t,d)* the frequency of term *t* in document *d*, *N* the total number of documents, and *df(t)* the number of documents containing term *t*. Pairs scoring at or above the 90th percentile (P90=0.1753) were labeled *Similar*, yielding 375 Similar pairs (10.0%) and 3,366 Not Similar pairs (90.0%). TF-IDF was chosen for its computational simplicity and interpretability as a labeling heuristic, but it is important to note this approach captures primarily *lexical* overlap rather than deeper *semantic* relatedness, and may introduce label noise for pairs that share vocabulary without sharing meaning, or vice versa — a known limitation discussed further in Section III-E.

Labeling validity was verified against manual annotation on a randomly drawn 200-pair sample (approximately 5.3% of the full pair set), stratified to include both TF-IDF-Similar and TF-IDF-Not-Similar cases. Two independent annotators with domain familiarity in BMKG scientific writing rated each pair as Similar or Not Similar based on topical and content overlap, without knowledge of the corresponding TF-IDF label. Agreement between the automatic TF-IDF label and the manual consensus label was then computed using Cohen's Kappa [9]:

$$\kappa = (p^0 - p^e) / (1 - p^e) \quad (2)$$

with p^0 the observed agreement proportion and p^e the expected chance agreement. The result, κ=0.82, indicates almost perfect agreement, supporting the use of the TF-IDF-based label as a practical proxy for semantic similarity in this setting, while the residual disagreement is consistent with the lexical-versus-semantic gap noted above. The dataset was split using stratified 70:30 sampling (random_state=42) into 2,618 training pairs (262 Similar, 2,356 Not Similar) and 1,123 test pairs (113 Similar, 1,010 Not Similar).

B. Siamese BiLSTM Architecture

Siamese BiLSTM consists of two identical weight-sharing sub-networks [1], [2]. Each branch comprises a 300-dimension Word2Vec embedding layer [11], a 128-unit Bidirectional LSTM (BiLSTM) layer, Batch Normalization, 0.30 Dropout, and a 128-unit Dense layer with ReLU activation. The resulting vectors *u* and *v* from both branches are compared using *cosine similarity*:

$$\cos(u,v) = (u \cdot v) / (\|u\| \|v\|) \quad (3)$$

The model is trained with Focal Loss [13] to address extreme class imbalance by down-weighting well-classified samples:

$$FL(p^t) = -\alpha(1-p^t)^\gamma \log(p^t) \quad (4)$$

with p^t the predicted probability for the true class, γ=2.0 the focusing parameter, and α=0.75 the class-weighting factor. In addition, *balanced* class weighting [15] is applied:

$$w^i = N / (K \times n^i) \quad (5)$$

with *N* the total sample count, *K* the number of classes, and n^i the number of samples of class *i*. On the 90:10 dataset this yields class weights of ≈0.56 (Not Similar) and ≈4.91 (Similar). Training uses the Adam optimizer [16] (learning rate=0.001), gradient clipping (clipnorm=1.0), and early stopping (patience=8) based on validation AUC.

The necessity of this Focal Loss and class-weighting combination was established empirically rather than assumed: an initial training run using standard Contrastive Loss *without* class weighting resulted in complete model collapse, with the network predicting every pair as Similar regardless of input (Recall=100%, Precision=10.1%, F1=18.28%). This single-condition ablation — comparing the unweighted baseline against the Focal-Loss-plus-class-weighting configuration (F1=64.84%) — demonstrates that class-imbalance handling is not an incremental refinement but a prerequisite for the model to learn a non-trivial decision boundary at all on this 90:10 dataset. A more fine-grained ablation isolating the individual contribution of Focal Loss versus class weighting alone was beyond the scope of this study and is noted in Section III-F as an avenue for future work.

TABLE I.
SIAMESE BiLSTM HYPERPARAMETER CONFIGURATION

Parameter	Value
Embedding	Word2Vec, 300 dimensions
BiLSTM Units	128 (256 output, bidirectional)
Dropout	0.30
Dense	128 units, ReLU activation
Loss Function	Focal Loss (γ=2.0; α=0.75)
Class Weight	Not Similar=0.56; Similar=4.91
Optimizer	Adam (lr=0.001)
Batch Size	32
Max. Epoch	50 (early stop, patience=8)
Optimal Threshold	0.60 (grid search)

C. IndoBERT Architecture

IndoBERT uses the pretrained *indobert-base-p1* model [10] (12 Transformer encoder layers, 768 hidden dimensions, 124 million parameters) as a shared encoder. Sentence representations are obtained via mean pooling over final-layer token embeddings, followed by a projection head (768→256 dimensions, ReLU) before *cosine similarity* computation using Eq. (3). The model is trained with Contrastive Loss [12], which minimizes the representation distance for similar pairs while penalizing dissimilar pairs only when their distance falls below a margin:

$$L = y \cdot D^2 + (1-y) \cdot \max(0, m-D)^2 \quad (6)$$

with *y* the pair label (1 if similar, 0 otherwise), *D* the distance (1 minus cosine similarity) between the two embeddings, and *m*=0.5 the minimum distance margin for dissimilar pairs. Class weighting equal to Siamese BiLSTM is applied. Fine-tuning uses the AdamW optimizer [14], which decouples weight decay from gradient adaptation.

TABLE II.
INDOBERT HYPERPARAMETER CONFIGURATION

Parameter	Value
Base Model	indobert-base-p1 (124M params)
Pooling	Mean Pooling
Projection Head	768→256, ReLU activation
Dropout	0.20
Loss Function	Contrastive Loss (margin=0.5)
Class Weight	Not Similar=0.56; Similar=4.91
Optimizer	AdamW
Batch Size	16
Max. Epoch	5 (early stop, patience=3)
Optimal Threshold	0.935 (grid search)

D. Evaluation Scheme

Both models were evaluated on 1,123 test pairs using four standard binary classification metrics [15]:

$$\text{Accuracy} = (TP+TN) / (TP+TN+FP+FN) \quad (7)$$

$$\text{Precision} = TP / (TP+FP) \quad (8)$$

$$\text{Recall} = TP / (TP+FN) \quad (9)$$

$$\text{F1-Score} = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (10)$$

with *TP*, *TN*, *FP*, and *FN* the counts of True Positive, True Negative, False Positive, and False Negative, respectively. Classification threshold was calibrated separately per model via grid search (0.10–0.90, step 0.01) maximizing F1-Score. The two models converge to markedly different optimal thresholds (0.60 for Siamese BiLSTM, 0.935 for IndoBERT) because their loss functions shape output score distributions differently: Focal Loss drives Siamese BiLSTM's cosine-similarity outputs toward two tight clusters near 0.50 and 1.00, so a mid-range cut-off around 0.60 already separates most classes cleanly, whereas Contrastive Loss allows IndoBERT's outputs to spread continuously across the upper range, requiring a threshold much closer to 1.0 before Precision and Recall balance at their optimum. This threshold behavior is visualized directly from the score distributions underlying Fig. 2 and is consistent with the loss-function properties described in Section II-B and II-C.

Statistical significance of the performance difference was tested using McNemar's test [17] with Yates' continuity correction, suitable for two models evaluated on an identical paired test set:

$$\chi^2 = (|b-c|-1)^2 / (b+c) \quad (11)$$

with *b* the number of cases where Siamese BiLSTM is correct but IndoBERT is wrong, and *c* the reverse. As additional validation, the 95% confidence interval of F1-Score was estimated using the Bootstrap method [18] with 1,000 resampling iterations:

$$CI^{95\%} = [P^{2.5}(\theta^*), P^{97.5}(\theta^*)] \quad (12)$$

with θ^* the set of F1-Score values from the 1,000 resampling iterations, and $P^{2.5}$ and $P^{97.5}$ the 2.5th and 97.5th percentiles of that distribution.

E. Implementation and Deployment

The best-performing model was deployed as a *Streamlit* web application, with the forward-pass inference reimplemented using pure NumPy [19] without deep-learning library dependencies, making it compatible with GPU-less cloud deployment environments. Model weights are stored in 13 *.npy* files totaling 16.77 MB. The application offers two analysis modes: one-to-one comparison, which returns a similarity score in approximately 0.11–0.15 seconds per pair on standard cloud CPU resources, and batch evaluation via CSV and ZIP upload for processing multiple document pairs in a single run. Systematic measurement of throughput at scale and end-user experience with the BMKG PMG Assessment Team was outside the scope of this study and is identified as a direction for future evaluation in Section III-F.

III. RESULTS AND DISCUSSION

A. Training Results

Siamese BiLSTM training on all 2,618 training pairs converged at epoch 7 of a scheduled maximum of 50, with a best validation AUC of 0.8285. An initial attempt using standard Contrastive Loss without class weighting resulted in model collapse, predicting all pairs as similar (Recall=100%, Precision=10%); the combination of Focal Loss and class weighting was necessary to overcome this. IndoBERT converged faster, at epoch 2 of a maximum of 5, consistent with fine-tuning behavior of large pretrained models, which typically require fewer iterations than training from scratch.

TABLE III.
TRAINING COMPARISON OF BOTH MODELS

Aspect	Siamese BiLSTM	IndoBERT
Best Epoch	7 of 15 (early stop)	2 of 5 (early stop)
Val. AUC / Val. Loss	AUC=0.8285	Loss=0.0286
Trainable Parameters	4,395,716 (4.4M)	124,443,649 (124M)
Model Size	16.77 MB	≈498 MB

B. Evaluation Metrics Comparison

Table IV summarizes evaluation results on the 1,123 test pairs, computed using Eq. (7)–(10). Siamese BiLSTM achieves an F1-Score of 64.84% at threshold 0.60, outperforming IndoBERT's F1-Score of 57.73% at threshold 0.935. Siamese BiLSTM leads on three of four metrics (Accuracy, Precision, F1-Score), while IndoBERT is slightly higher on Recall (74.34% versus 73.45%). The most notable gap is in Precision (10.85 percentage points), indicating Siamese BiLSTM produces proportionally fewer false positives. Fig. 1 visualizes the four metrics side by side.

TABLE IV.
EVALUATION METRICS COMPARISON

Metric	Siamese BiLSTM	IndoBERT	Diff.
Accuracy	91.99%	89.05%	+2.94 pp
Precision	58.04%	47.19%	+10.85 pp
Recall	73.45%	74.34%	-0.89 pp
F1-Score	64.84%	57.73%	+7.11 pp

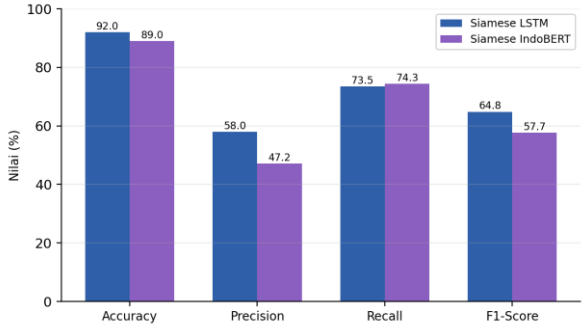

Fig. 1. Evaluation Metrics Comparison of Siamese BiLSTM and IndoBERT

Confusion matrix analysis (Fig. 2) shows Siamese BiLSTM produced 950 True Negatives, 60 False Positives, 30 False Negatives, and 83 True Positives, compared to IndoBERT's 916 True Negatives, 94 False Positives, 29 False Negatives, and 84 True Positives. The gap in False Positives (94 versus 60) is the main contributor to IndoBERT's lower Precision.

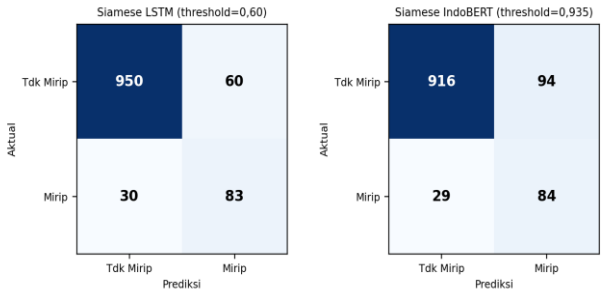

Fig. 2. Confusion Matrix of Siamese BiLSTM and IndoBERT

C. Statistical Significance Test

McNemar's test using Eq. (11) with b=83 (cases where BiLSTM was correct but IndoBERT was wrong) and c=50 (the reverse) yields $\chi^2=(|83-50|-1)^2/(83+50)=7.6992$, $p=0.0055$ ($<\alpha=0.05$), rejecting H^0 (equal performance). This confirms the performance difference between Siamese BiLSTM and IndoBERT is statistically significant, not merely random variation. As additional validation, the 95% Bootstrap Confidence Interval computed via Eq. (12) from 1,000 iterations (Fig. 3) shows an F1-Score range of [0.5811–0.7104] (mean 0.6473) for Siamese BiLSTM and [0.5082–0.6415] (mean 0.5763) for IndoBERT. Although the two intervals show partial overlap, this does not contradict the significant McNemar result: McNemar's test is paired and compares predictions on identical test pairs, whereas the confidence interval estimate is independent and disregards prediction correlation on the same pairs — the two results are complementary rather than contradictory.

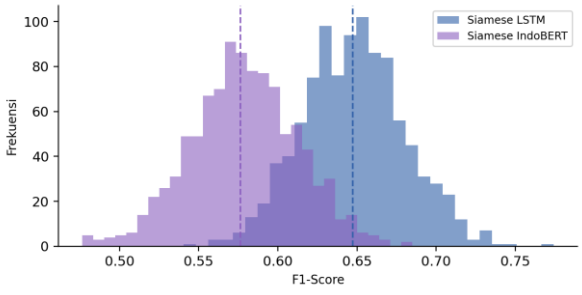

Fig. 3. Bootstrap F1-Score Distribution (1,000 Iterations)

D. Discussion

The advantage of Siamese BiLSTM in this study contrasts with the general literature trend reporting Transformer-based superiority for STS tasks [3], [4]. At least three factors may explain this. First, dataset scale: Krisnawati *et al.* [6], using Universal Sentence Encoder on a small 116-document bilingual set, similarly reported limited effectiveness on domain-specific text, consistent with large pretrained models requiring more fine-tuning data to perform optimally — a pattern this study's own comparison reproduces at a larger parameter scale (Section II-A). Second, domain characteristics: BMKG PMG papers use highly specific meteorological, climatological, and geophysical terminology likely underrepresented in IndoBERT's general-domain pre-training corpus, whereas the Word2Vec embedding was retrained specifically on this study's technical corpus and therefore captures domain vocabulary more directly; this is a plausible source of embedding bias rather than a demonstrated one, since the pre-training corpora of both models were not directly audited in this study. Third, parameter scale: IndoBERT's 124 million parameters are more prone to overfitting on a 2,618-pair training set than BiLSTM's 4.4 million parameters.

Loss function characteristics also contribute to the performance gap, as detailed in Section III-C: Focal Loss on Siamese BiLSTM produces a more discriminative, tightly clustered score distribution, whereas Contrastive Loss on IndoBERT produces a more continuously spread distribution requiring a much higher decision threshold.

It is important to situate these findings relative to other STS approaches not directly benchmarked in this study. Sentence-BERT [5] and SimCSE [20] represent widely adopted alternatives that produce sentence embeddings through contrastive or Siamese fine-tuning objectives similar in spirit to the architectures compared here, and have demonstrated strong results on general-domain, high-resource benchmarks. This study does not include a direct empirical comparison against these methods, and the reported advantage of Siamese BiLSTM should therefore be interpreted as relative to IndoBERT with Contrastive Loss specifically, not as a claim of superiority over the broader family of Transformer-based sentence-embedding techniques. A direct comparison, discussed further in Section III-F, would strengthen the positioning of this work's contribution within the wider STS literature.

E. Error Analysis

Beyond the aggregate metrics in Table IV, qualitative inspection of misclassified pairs during model development revealed two recurring failure patterns common to both architectures. The first is *vocabulary-topic conflation*: pairs of KTI documents from different scientific subfields (e.g., a seismology paper and a climatology paper) that happen to share domain-general BMKG terminology — such as station names, measurement units, or procedural language — were occasionally scored as more similar than their actual topical content warranted. This pattern is consistent with the TF-IDF-based labeling limitation noted in Section II-A, since lexical overlap does not guarantee semantic relatedness, and both models were trained on labels inheriting that same lexical bias. The second pattern is *paraphrase underestimation*: pairs discussing closely related findings using substantially

different sentence structure and vocabulary were sometimes scored below the classification threshold, particularly by IndoBERT at its high 0.935 cut-off, where small shifts in embedding position near the boundary have an outsized effect on the final label.

These observations are drawn from qualitative review during development rather than a systematic, pre-registered error-annotation protocol over the full 1,123-pair test set, and should be read as motivating hypotheses for future targeted analysis rather than quantified findings.

F. Limitations and Future Work

This study has several limitations that constrain the scope of its conclusions and point to concrete directions for future work. Dataset scale: the source corpus of 87 documents reflects a single BMKG PMG assessment period; while combinatorial pairing yields 3,741 training pairs, this remains modest for training a 124-million-parameter model such as IndoBERT, and results should not be assumed to hold at larger corpus scales without re-evaluation. Labeling method: ground-truth labels were derived from an automatic TF-IDF-based heuristic rather than full manual expert annotation; although validated against manual annotation on a 200-pair sample (κ=0.82), a full manual gold-standard label set, or comparison against SBERT- or SimCSE-derived pseudo-labels, would allow a more rigorous assessment of label quality. Baseline coverage: this study compares two architectures under matched Siamese training but does not benchmark against off-the-shelf Sentence-BERT [5] or SimCSE [20] embeddings, nor against a Precision-Recall / AUC-PR analysis, which is generally more informative than ROC-style metrics under the 90:10 class imbalance present here; both comparisons are natural next steps. Ablation depth: the necessity of class-imbalance handling was demonstrated (Section II-B), but the individual contribution of Focal Loss versus class weighting was not isolated through a full factorial ablation. Real-world validation: the claim that the deployed application can improve assessment objectivity and efficiency is grounded in its quantitative performance and sub-second inference time (Section II-E), but has not yet been validated through structured evaluation with the BMKG PMG Assessment Team in live usage, including inter-rater comparison against assessor judgments. Addressing these points — particularly a Sentence-BERT/SimCSE baseline, AUC-PR reporting, and a field evaluation with assessors — forms the primary agenda for continued work on this system.

IV. CONCLUSION

Within the scope of the BMKG PMG scientific-paper dataset evaluated here, Siamese BiLSTM with Focal Loss outperforms IndoBERT with Contrastive Loss for STS detection, with a statistically significant advantage (F1-Score 64.84% versus 57.73%; χ^2=7.6992; p=0.0055). This advantage is accompanied by substantially better computational efficiency — 16.77 MB versus ≈498 MB model size, with no GPU required for inference — making Siamese BiLSTM the more practical choice for resource-constrained deployment in this setting. These findings should be interpreted as specific to domain-specific, limited-size Indonesian technical corpora of the kind examined here, rather than as a general claim that recurrent architectures outperform Transformer-based models for Indonesian STS at large; broader generalization across document types, institutions, or larger corpora requires further validation. The Siamese BiLSTM model has been deployed as a *Streamlit* web application accessible to the BMKG PMG Assessment Team for real-time KTI similarity detection, subject to the field-validation limitations noted in Section III-F. Future work should prioritize benchmarking against Sentence-BERT and SimCSE, AUC-PR evaluation under class imbalance, a fine-grained ablation of loss and weighting components, dataset expansion through full manual annotation, and structured evaluation with BMKG assessors in live usage.

ACKNOWLEDGMENT

The authors thank the Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) for providing access to PMG Scientific Paper data used in this research, and the Master of Informatics Engineering Program, Universitas Pamulang, for guidance and support throughout this study.

REFERENCES

[1] J. Mueller and A. Thyagarajan, "Siamese recurrent architectures for learning sentence similarity," in Proc. 30th AAAI Conf. Artif. Intell., 2016, pp. 2786–2792.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," arXiv:1810.04805, 2019.

[4] Y. Li, C. L. P. Chen, and T. Zhang, "A survey on Siamese network: Methodologies, applications, and opportunities," IEEE Transactions on Artificial Intelligence, vol. 3, no. 6, pp. 994–1014, 2022.

[5] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in Proc. 2019 Conf. Empirical Methods in Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 3982–3992.

[6] L. D. Krisnawati, A. W. Mahastama, S.-C. Haw, K.-W. Ng, and P. Naveen, "Indonesian-English textual similarity detection using Universal Sentence Encoder (USE) and Facebook AI Similarity Search (FAISS)," CommIT Journal, vol. 18, no. 2, pp. 183–195, 2024.

[7] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," Information Processing & Management, vol. 24, no. 5, pp. 513–523, 1988.

[8] D. Viji and S. Revathy, "Analyzing semantic similarity amongst textual documents to suggest near duplicates," Indonesian Journal of Electrical Engineering and Computer Science, vol. 25, no. 3, pp. 1703–1711, 2022.

[9] J. Cohen, "A coefficient of agreement for nominal scales," Educational and Psychological Measurement, vol. 20, no. 1, pp. 37–46, 1960.

[10] B. Wilie et al., "IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding," arXiv:2009.05387, 2020.

[11] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv:1301.3781, 2013.

[12] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2, 2006, pp. 1735–1742.
[13] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 2980–2988.
[14] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in Proc. Int. Conf. Learn. Represent., 2019.
[15] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," Information Processing & Management, vol. 45, no. 4, pp. 427–437, 2009.
[16] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv:1412.6980, 2015.
[17] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," Psychometrika, vol. 12, no. 2, pp. 153–157, 1947.
[18] B. Efron and R. J. Tibshirani, An Introduction to the Bootstrap. New York: Chapman & Hall/CRC, 1993, pp. 168–177.
[19] C. R. Harris et al., "Array programming with NumPy," Nature, vol. 585, no. 7825, pp. 357–362, 2020.
[20] T. Gao, X. Yao, and D. Chen, "SimCSE: Simple contrastive learning of sentence embeddings," in Proc. 2021 Conf. Empirical Methods in Natural Language Processing (EMNLP), 2021, pp. 6894–6910.