

Minimax-Based Feature Selection for Low-Resource Language Classification Theoretical Research Paper with A Worked Numerical Illustration

Maftukha Sayfullaeva¹, Tatyana Kim², Muyassar Kholova³, Muhayyo Roziyeva⁴,
Dilnoz Ollaberganova⁵, Abdurasul Boltayev⁶, Mastura Allayarova⁷

¹Tashkent University of Information Technologies named after Muhammad al-Khorazmi, Tashkent, Email: Uzbekistan. maftuha87@mail.ru, <https://orcid.org/0009-0008-4337-1692>

²Associate Professor, Tashkent State University of Oriental Studies, Tashkent, Uzbekistan. Email: kimtancha1976@gmail.com, <https://orcid.org/0000-0001-6145-0724>

³Associate Professor, Department of Linguistics, Termez State University, Termez, Uzbekistan. Email: xolovam@tersu.uz, <https://orcid.org/0000-0002-7604-2958>

⁴Researcher, Mamun University, Khiva, Uzbekistan. Email: ruzieva_m@gmail.com, <https://orcid.org/0009-0003-1739-742X>

⁵Senior Lecturer, Department of World Literature, Uzbekistan State World Languages University, Tashkent, Uzbekistan. Email: dilnozollaberganova@uzswlu.uz, <https://orcid.org/0009-0003-8734-0963>

⁶Jizzakh State Pedagogical University, Jizzakh, Uzbekistan. Email: a.boltayev@jspu.uz, <https://orcid.org/0000-0002-5611-239X>

⁷Senior Lecturer, Department of Economics Gulistan State University, Gulistan, Uzbekistan. Email: allayarovamastura@gmail.com, <https://orcid.org/0009-0000-4164-1013>

Abstract

Character and word features are commonly large sets that are used in language classification systems. If the resources are restricted in languages that have limited amount of labeled data, however, a feature set that works well for the overall corpus may work poorly for a small language subset or an unfamiliar domain of text. A minimax approach is proposed in this paper to choose a small set of features for the identification of a target language. Large validation loss is computed for a group of languages and domains, as opposed to the average loss. The framework is based on a feature budget constraint, regularized classifier training and group-specific evaluation. An equivalent adversarial weighting interpretation is responsible for the method's attention to poorly served groups. A finite numerical illustration compares six candidate feature subsets and illustrates how a compromise between the average performance and worst group protection can be achieved. Also included is a practical selection procedure and an empirical evaluation protocol. The paper is methodological: There is no report of any corpus experiment, and all the numbers lost in the illustration are assumed ones. The framework presents a testable method for compact language classification in limited resources, and highlights the importance of the robustness of group estimates and the design of validation.

Keywords: Minimax optimization, feature selection, low-resource languages, language identification, group robustness, text classification.

1. The classification problem under unequal resources

Language identification is the process of labelling an input text with a language. It does facilitate the organization of the corpus, multilingual search and routing of text to language-specific processing tools. In this paper, “language classification” is not sentiment classification or topic classification in a language, but classification of the language itself. We assume that a text has exactly one reference language label. Beyond the present closed-set model are requirements for extensions of such mixed-language text and for unknown languages.

Language resources are very unequally distributed. Joshi et al. report on disparities in language representation and technological assistance of languages (Joshi et al., 2020). Limited labeled examples, limited domain coverage, few linguistic tools or a combination of these factors could represent resource scarcity. Therefore, low-resource classification should be decided not on a universal threshold, but on clear and explicit data budgets, and collection conditions.

When the number of text features is large compared to the number of labeled samples, then feature selection takes importance. Frequent words might be very common in dominant news corpus and not found in

conversational text. A small language group may be identified by a rare character sequence and not detected if the frequency of the sequence is pooled below a threshold. So, when choosing features by their average contribution, patterns of well-represented groups can be retained while some of the features required elsewhere are removed.

Thus, the central question becomes: What is the criterion that the losing group will have the least classification loss, given the affordable features, for the system that is supposed to classify the groups? This question changes the selection criterion, but keeps the simple classifier and familiar text representations. It is not an algorithm, but an application framework whose foundations are taken from existing feature-selection and robust-optimization principles.

2. From language evidence to a candidate feature pool

2.1. Character, word, and boundary evidence

In the survey of language identification conducted by Jauhiainen et al., there are many representations and modeling options that can be used for the task (Jauhiainen et al., 2019; Blodgett et al., 2020). Language identification models are useful for small-scale language identification systems like `langid.py` (Lui & Baldwin, 2012). The research results argue for a set of features that can be produced without relying heavily on language-specific resources.

Character n-grams are local spelling patterns. Word features can be used as lexical evidence, but will need uniform tokenization and adequate observations. The characteristics of letters that are sensitive to boundaries can help to differentiate sequences within word from sequences at word beginnings and ends. HeLI leverages word-based evidence in combination with the backoff technique (Jauhiainen et al., (2016) while Gomez et al., (2017) examine a mixture of typed and untyped character n-grams and words for similar-language discrimination. These methods offer informative starting points, but lack evidence of the performance of the minimax selector offered here.

The multilingual evaluation also shows that covering a wide range of languages is more important than drawing conclusions from a narrow range of language differences that are easy to distinguish. A candidate vocabulary needs to then preserve features observed in individual languages, in addition to conforming to privacy and data quality checks, and not just be based on a pooled frequency rule in table 1.

Table 1: Candidate representations and their selection concerns.

Representation	Main evidence	Concern under scarce data
Character n-grams	Local orthographic patterns	Rare sequences may be unstable
Word unigrams	Lexical differences	Vocabulary may depend on topic
Boundary character features	Word beginnings and endings	Tokenization choices affect counts
Script indicators	Writing-system differences	Transliteration can remove the cue

2.2. What existing selection methods contribute

Filters, wrappers and embedded procedures are the three types of feature selection procedures (Jauhiainen et al., 2017; Guyon & Elisseeff, 2003). Filters are used before classifiers are fitted and Wrappers are used to evaluate subsets using a given learning algorithm. Study of text classification compares the frequency-based and statistical selection criteria. Other approaches to reducing active coefficients include mutual-information criteria, which can balance the relevance of the coefficients with the redundancy between them; and sparse regularization (Peng et al., 2005; Al-Omari & Al-Haija, 2024).

These are useful tools that can still be incorporated. For instance, a language-aware filter could narrow down the number of candidates prior to the wrapper search. The change proposed is to use the worst group loss for the remaining subsets. This distinction is significant, because in a small language group two subsets may have the same number of features, and similar mean losses, but quite different losses.

2.3. Separating linguistic evidence from collection artifacts

Names of websites, headers of files, and boilerplate text may be linked to language, but with no transferable linguistic information. Language-dependent differences in sources of collection can lead to keeping these correlations in both training and validation sets when split by random sentences. The documents should then be stored together by source, author or collection batch where possible based on the available metadata. Duplicate and near duplicate material should be eliminated prior to splitting.

Ensure that distinctions important to target languages are maintained in the normalization process. Normalization and removal of diacritics or merging of separate letters might be considered, but this can eliminate valuable evidence. All preprocessing decisions should be recorded and validated via an ablation if it changes the representation in a way that matters.

3. A budget-constrained minimax selector

3.1. Samples, masks, and group definitions

Let the labeled training data be (equation 1)

$$\mathcal{D}_{\text{tr}} = \{(x_i, y_i)\}_{i=1}^n, \quad y_i \in \{1, \dots, L\} \quad (1)$$

Here, x_i is a text and L is the number of target languages. A feature map $\phi(x) \in \mathbb{R}^d$ contains normalized counts derived from a vocabulary fitted on training data only. The binary mask (equation 2)

$$z \in \{0,1\}^d, \quad \phi_z(x) = z \odot \phi(x) \quad (2)$$

indicates which features remain active. The symbol $\odot$ denotes elementwise multiplication. Each feature has a nonnegative cost c_j , and feasible masks satisfy (equation 3)

$$\mathcal{Z}_B = \{z \in \{0,1\}^d : \sum_{j=1}^d c_j z_j \leq B\} \quad (3)$$

With $c_j = 1$, the budget B limits the number of selected features. More detailed costs can approximate stored coefficients or extraction effort. Such costs must be measured or justified; feature count alone does not fully describe deployment latency.

Validation examples are partitioned into nonempty groups $\mathcal{V}_1, \dots, \mathcal{V}_G$. A group may represent a language or a language–domain combination. Group definitions are fixed before examining selection outcomes. If a combination has no validation observations, the method cannot estimate its loss and must report that coverage gap.

3.2. Training the classifier for each subset

A regularized multinomial logistic classifier provides a transparent baseline (equation 4):

$$p_{\theta,z}(y = \ell \mid x) = \frac{\exp(w_\ell^\top \phi_z(x) + b_\ell)}{\sum_{r=1}^L \exp(w_r^\top \phi_z(x) + b_r)} \quad (4)$$

For a fixed mask, model parameters are estimated from training data (equation 5):

$$\hat{\theta}(z) \in \operatorname{argmin}_{\theta} \left\{ \frac{1}{n} \sum_{i=1}^n -\log p_{\theta,z}(y_i \mid x_i) + \alpha \|W\|_F^2 \right\}, \quad \alpha > 0 \quad (5)$$

The regularization strength is chosen within the model-selection procedure. The same training rules, stopping criterion, and hyperparameter search are used for every subset. Otherwise, changes in fitting effort could be mistaken for benefits of feature selection.

The group validation loss is (equation 6)

$$\hat{R}_g(z) = \frac{1}{|\mathcal{V}_g|} \sum_{(x,y) \in \mathcal{V}_g} -\log p_{\hat{\theta}(z),z}(y \mid x) \quad (6)$$

Cross-entropy evaluates predicted probabilities as well as their correctness. It is not a misclassification percentage, and a reduction in this loss does not imply an identical reduction in error rate.

3.3. Choosing features against the largest loss

The proposed selection rule is (equation 7)

$$z^* \in \operatorname{argmin}_{z \in \mathcal{Z}_B} \max_{1 \leq g \leq G} \hat{R}_g(z) \quad (7)$$

The inner maximum identifies the currently least well-served group. The outer minimization chooses a feasible mask that makes this largest loss as small as possible. This follows the worst-case reasoning of robust optimization and is related to distributionally robust learning (Jade & Bewoor, (2025)) and group robustness (Bertsimas et al., (2011; Saleem et al., 2025; Sagawa et al., 2020)). However, the present formulation selects a feature mask using validation groups; it should not be conflated with training a neural network by group DRO.

An equivalent epigraph formulation is (equation 8)

$$\begin{aligned} \min_{z,t} \quad & t \\ \text{subject to} \quad & \hat{R}_g(z) \leq t, \quad g = 1, \dots, G \\ & z \in \mathcal{Z}_B. \end{aligned} \quad (8)$$

This expression is conceptually simple but does not turn subset search into a linear program. The losses depend on retrained classifier parameters, and the binary search space remains combinatorial.

3.4. An adversarial weighting interpretation

Let $\Delta_G = \{q \in \mathbb{R}^G : q_g \geq 0, \sum_g q_g = 1\}$ be the probability simplex. For any fixed mask (equation 9),

$$\max_g \hat{R}_g(z) = \max_{q \in \Delta_G} \sum_{g=1}^G q_g \hat{R}_g(z) \quad (9)$$

The equality holds because a weighted average cannot exceed its largest component, and placing all weight on a maximizing group attains that component. The selector can therefore be viewed as facing an evaluator that concentrates weight on the most difficult group.

This protects against changes in the mixture weights of the represented groups. It does not guarantee protection against a new language, a new writing convention, or an unseen domain whose conditional distribution differs from every validation group. No interchange of the minimum and maximum is assumed.

4. Practical search when exhaustive selection is too expensive

Exhaustive search is suitable only for a small candidate pool. With an exact budget of k unit-cost features, there are $\binom{d}{k}$ subsets. A practical implementation can first retain a moderate pool through within-language screening and then perform a greedy wrapper search.

4.1. Selection procedure

1. Split documents into training, selection-validation, and final test partitions before constructing the vocabulary.
2. Fit preprocessing and the candidate feature map on training data. Apply a documented, language-aware screening rule if necessary.
3. Start with an empty mask and a bias-only classifier. Compute its group validation losses.
4. For each affordable unselected feature, retrain the classifier with that addition and compute the maximum group loss.
5. Accept the addition with the lowest maximum loss if it improves the current objective. For ties, prefer fewer features and then lower pooled validation loss.
6. Stop when the budget is reached or no addition improves the objective. Optionally examine one-for-one swaps using the same acceptance rule.
7. Lock the selected mask and training configuration, refit on the permitted development data, and evaluate once on the untouched test partition.

Search cost can be reduced by using feature blocks, e.g. by choosing a set of character features to screen before screening the features within the set. A block budget should, therefore, be separated from a budget for each individual feature. Warm starts may decrease fitting time as long as all the candidates have the same optimization tolerance.

The greedy procedure is a heuristic solution. This is because the interaction of features suggests that a feature that is individually unhelpful can become helpful when used along with another feature. There is no sub modularity or global optimality assumption. The final selected mask is locally stable under additions and swaps but this is not a solvability property for the Neighbourhood chosen, merely a property of the selected mask.

The same validation set can also be used repeatedly, leading to the overfitting of feature choices. In case of very small datasets, the entire selection pipeline should be estimated by using nested cross-validation or repeated source-aware outer splits. The feature vocabulary, budget, group definitions and stopping rule are not to be influenced by the final test data.

5. Worked example: average loss versus worst-group loss

5.1. Assumed inputs

Consider four hypothetical feature blocks: A (character trigrams), B (character four-grams), C (word unigrams), and D (boundary character features). Exactly two equally priced blocks are selected. Three validation groups, g_1 , g_2 , and g_3 , represent different language-domain conditions. Their pooled proportions are $\pi = (0.6, 0.3, 0.1)$.

Table 2 shows the assumed cross-entropy losses for classifier fitting some hypothetical classifiers. These values are constructed to illustrate the decision rule. They are not corpus results, predictions for named languages, nor evidence of any superiority of any one particular feature family. The two-block budget doesn't also set that the ensuing representations have an equal number of individual features.

Table 2: Synthetic losses and calculated selection scores.

Selected blocks	Group 1	Group 2	Group 3	Pooled loss	Maximum loss
A + B	0.30	0.42	0.80	0.386	0.80
A + C	0.20	0.35	0.95	0.320	0.95
A + D	0.40	0.45	0.50	0.425	0.50
B + C	0.25	0.38	0.90	0.354	0.90
B + D	0.43	0.46	0.55	0.451	0.55
C + D	0.35	0.50	0.70	0.430	0.70

5.2. Exact comparison of the six choices

A pooled-loss selector minimizes (equation 10)

$$\hat{R}_\pi(z) = \sum_{g=1}^3 \pi_g \hat{R}_g(z) \quad (10)$$

It selects $A + C$, because (equation 11)

$$\hat{R}_\pi(A + C) = 0.6(0.20) + 0.3(0.35) + 0.1(0.95) = 0.320 \quad (11)$$

The minimax selector instead chooses $A + D$, whose largest loss is 0.50, the smallest maximum among all six subsets. Its pooled loss is 0.425. Relative to the pooled-optimal choice, the worst-group loss decreases by (equation 12)

$$0.95 - 0.50 = 0.45, \quad \frac{0.45}{0.95} \times 100\% \approx 47.4\% \quad (12)$$

The pooled loss increases by 0.105. These calculations demonstrate a tradeoff in the assumed table, not an empirical improvement. The example also shows why reporting only an overall score can conceal a substantial group disparity.

5.3. What a changed deployment mixture would mean

If deployment group proportions become equal, the assumed mean loss for $A + C$ becomes 0.500, whereas that for $A + D$ becomes 0.450. Thus, the ranking changes under this particular reweighting. More generally, Equation (9) implies that the largest represented-group loss bounds every mixture of those same group losses. These bound uses fixed conditional losses. It is not necessarily maintained automatically when deployment examples in a group become more or less difficult. Changes in a mixture and changes within language–domain groups therefore must be analyzed during a real deployment evaluation.

6. Designing an empirical test of the framework

6.1. Data budgets and comparison systems

It is important that it should be identified in a subsequent corpus study as to its target languages, collection sources, label guidelines and its limitations for availability. Where data collection, naming or usage conditions impact on language communities, they should be involved. There are some known training budgets that can be simulated, but there is no justification to use a subsample of an existing corpus as a representation of all aspects of the natural low resource scenario.

Comparisons with relevant classifiers include a full-feature logistic classifier, a frequency-filtered classifier, a pooled statistical filter, a pooled-loss wrapper, and the minimax wrapper. The feature-count or measured-cost budget of each compact method should be the same. If the requirements for the resources and the coverage of the languages are documented, an additional comparison can be provided by a multilingual pretrained model; Cross-lingual representation learning is a different modeling trajectory (Ta, 2023; Conneau et al., 2020). It is important to report the costs of pretraining, not as zero-cost training of a small linear model.

6.2. Outcomes that should be reported together

The main result should be the maximum test cross-entropy within the pre-defined groups, as well as per-group losses and group sample sizes. They offer complementary classification measures in the form of macro-F1, balanced accuracy and per-language recall. Practical cost is measured in terms of model size, chosen number of features, selection time, and inference time.

Variation in performance should be reported on repeated outer splits and membership should be reported. If there are multiple documents in the sources, then re-sample the sources instead of assuming that each sentence is independent. Loss estimates may be unreliable for very small groups and should be reported as a range of possible losses rather than with just one minimax value.

6.3. Ablations and falsifiable expectations

Three helpful ablations are made: removal of language-aware screening, replacement of language-aware screening with pooled screening, and replacement of language–domain groups with language-only groups. They establish if there is an observed difference in favour of representation, screening, or group construction.

The hypothesis is testable because it suggests that the minimax wrapper will achieve a reduction in the worst case loss in the largest group compared to the pooled wrapper at the same budget. The other is that the features it has chosen are more resistant to splits in the source. Both hypotheses could be incorrect. The method should be deemed unsuccessful on a data set when its worst-group gains are not present, statistical gains are uncertain, or the computational cost and the overall error increase in an unacceptable way.

7. Limits of worst-group feature selection

The worst-group optimization problem can be sensitive to mislabeled data and tiny groups. The advantage of using cross entropy is that the loss is unbounded if the model gives a very low confidence to the reference class; this means that a single confident error can have a big impact on the average result of a small group. In addition to regularisation and good label review, these are helpful but by themselves do not remove sampling uncertainty. Examples of reporting calibration and errors can be useful to differentiate true difficulty from problems in the reporting of the items or error during pre-processing.

Strong selection will not be able to fill in the gaps in the candidate pool. It is also not possible to identify an unseen language without samples or external knowledge. Character normalization, segment-level labels and explicit rejection options might be necessary to support transliterations, code switching, and shared vocabulary not included in the model.

Language groups do not necessarily constitute social fairness categories. Lower worst group loss is not a distributional result of equity, suitable naming of the language, or acceptable downstream uses. Blodgett et al. highlight the importance of linking technical language notions of bias with social context of language technology (Duchi & Namkoong, 2018). Thus, deployment should be made based on both group performance and community needs and data-governance requirements.

Last but not least, minimax protection is expensive. It can lead to more average loss and to a noisy group estimate. If a specific average-loss-tolerance requirement has been specified for a deployment, it can be included, but the tolerance would have to be substantiated before final test. If the problem is rendered, the resulting problem may be infeasible, and this is a good sign that the representation or resource allocated is not sufficient for the problem.

Conclusion

In this paper, the problem of language identification in low-resource conditions is posed as a constrained feature selection task with a criterion based on worst-group validation. This mathematical formulation involves selecting a binary vector indicating which features are retained, followed by training a regularized classifier on the reduced feature set and optimizing for the maximum loss on predefined groups of languages and domains. The adversarial weighting interpretation clarifies the rationale behind the robustness to protected mixture from represented groups. The numerical illustration reveals that the minimax selection and pooled loss selection may select different sets of features, with an inherent trade-off between average losses. Because the numerical example relies on fictitious numbers, it is able to demonstrate the behavior of the criterion but not the utility thereof. The empirical procedure described above will help in that respect.

References

1. Al-Omari, M., & Al-Haija, Q. A. (2024). Towards Robust IDSs: An Integrated Approach of Hybrid Feature Selection and Machine Learning. *J. Internet Serv. Inf. Secur.*, 14(2), 47-67. <https://doi.org/10.58346/JISIS.2024.12.004>
2. Bertsimas, D., Brown, D. B., & Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM review*, 53(3), 464-501. <https://doi.org/10.1137/080734510>
3. Blodgett, S. L., Barocas, S., Daumé Iii, H., & Wallach, H. (2020, July). Language (technology) is power: A critical survey of “bias” in NLP. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 5454-5476).
4. Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., ... & Stoyanov, V. (2020, July). Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 8440-8451). <https://doi.org/10.18653/v1/2020.acl-main.747>
5. Duchi, J., & Namkoong, H. (2018). *Learning models with uniform performance via distributionally robust optimization*. arXiv. <https://arxiv.org/abs/1810.08750>
6. Gomez, H., Markov, I., Baptista, J., Sidorov, G., & Pinto, D. (2017, April). Discriminating between similar languages using a combination of typed and untyped character n-grams and words. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)* (pp. 137-145). <https://doi.org/10.18653/v1/W17-1217>
7. Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar), 1157-1182.
8. Jade, S. P., & Bewoor, L. A. (2025). Developing a hybrid word embedding model using contextual information and classification algorithms. *International Academic Journal of Science and Engineering*, 12(4), 538-555. <https://doi.org/10.71086/IAJSE/V12I4/IAJSE12116>
9. Jauhiainen, T., Lindén, K., & Jauhiainen, H. (2016, December). HeLI, a word-based backoff method for language identification. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)* (pp. 153-162).
10. Jauhiainen, T., Lindén, K., & Jauhiainen, H. (2017, May). Evaluation of language identification methods using 285 languages. In *Proceedings of the 21st Nordic Conference on Computational Linguistics* (pp. 183-191).
11. Jauhiainen, T., Lui, M., Zampieri, M., Baldwin, T., & Lindén, K. (2019). Automatic language identification in texts: A survey. *Journal of Artificial Intelligence Research*, 65, 675-782. <https://doi.org/10.1613/jair.1.11675>
12. Joshi, P., Santy, S., Budhiraja, A., Bali, K., & Choudhury, M. (2020, July). The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 6282-6293). <https://doi.org/10.18653/v1/2020.acl-main.560>
13. Lui, M., & Baldwin, T. (2012, July). langid. py: An off-the-shelf language identification tool. In *Proceedings*

of the ACL 2012 system demonstrations (pp. 25-30).

14. Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8), 1226-1238. <https://doi.org/10.1109/TPAMI.2005.159>
15. Sagawa, S., Koh, P. W., Hashimoto, T. B., & Liang, P. (2020). *Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization*. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
16. Saleem, H., Javed, M., Hanif, K., Ullah, A., Ghani, M. U., Waqas, M., ... & Hassan, S. A. (2025). a Robust Machine Learning-Based Ensemble Learning Framework for Hate Speech Detection in Low-Resource Social Media Text. *Archives for Technical Sciences*, 2(33), 747-759. <http://dx.doi.org/10.70102/afts.2025.1833.747>
17. Ta, N. T. D. (2023). Explainable Artificial Intelligence for Automated Language Assessment Systems. *International Journal of English and Education*, 12(2), 124-132. <https://doi.org/10.67050/IJEE/V12I2/IJEE232010>
18. Uvarajan, K. P. (2026). Design and Implementation of an Ultra-LowPower VLSI Architecture for Energy-Constrained IoT Communication Systems. *IAECES Journal of Electronics and Communication Engineering*, 1(1), 95-104.
19. Raja, C. K. (2026). Memory-Optimized VLSI Design for Low-Latency Streaming Signal Processing Applications. *Journal of Integrated VLSI and Signal Processing*, 1(3), 16-22.
20. Jing, W. (2026). Routing-Driven Via Density Reduction Techniques for Reliability-Aware Physical Design. *Journal of VLSI and Embedded System Design*, 2(3), 40-51.