CrossLingual-Noised BackTranslation

Aslan,F. Aghayev, Sergey A. Molodyakov

***Abstract*—Low-resource languages suffer from data scarcity, limiting robust text classification. We introduce a cross-lingual, noise-injected augmentation pipeline for Azerbaijani that leverages a higher-resource latent space. Starting with Azerbaijani, we translate into Turkish, encode with a multilingual encoder–decoder, inject token-level noise into encoder states, decode in Turkish, and translate back to Azerbaijani to yield diverse, fluent paraphrases. This process exploits the model's stronger expressiveness in Turkish while anchoring semantics across languages. On an Azerbaijani news dataset, training with the generated paraphrases increases accuracy and robustness and expands lexical and structural variety. Measured by cosine similarity and BLEU, paraphrases preserve meaning while increasing diversity. The approach offers a scalable way to create diverse augmentations.**

***Keywords*—Augmentation, Back Translation, Classification, Low-resource Languages, Natural Language Processing.**

I. Introduction

Developing accurate NLP models for low-resource languages like Azerbaijani is challenging due to limited labeled data [1]. Data augmentation is a widely-used strategy to address data scarcity by generating synthetic training examples from existing ones. For text data, augmentation techniques must create new sentences that maintain the original meaning (to preserve label consistency) while introducing enough variation to enrich the model's exposure [2][3]. Common text augmentation methods include synonym replacement, random word edits (known as Easy Data Augmentation or EDA) [4], noising or shuffling words, and paraphrasing via machine translation [5]. Among these, back-translation – translating a sentence to another language and back to the original language – has proven especially effective for creating paraphrases that retain meaning with correct syntax [6].

Another line of work is latent-space and model-based paraphrasing [7][8]. The latent noise methods are effective but the quality and diversity of augmentation depends on the quality of the embedding model itself. The diversity of paraphrases is crucial for augmentation to be effective – too little change and the model learns nothing new, too much change and the meaning or label may be altered [9].

Recognizing these problems, we propose a new augmentation method using a related high-resource language to obtain more diverse augmentations. The scientific novelty of this approach lies in combining cross-lingual translation with latent noise for augmentation. To our knowledge, previous works have either done back-translation (using translation but no internal noise) [1] or monolingual noise-driven generation (noise in the original language, e.g. BART-style denoising [10]).

In the following sections, we review related work on low-resource data augmentation, detail our proposed method and experimental setup, compare paraphrase quality and classification performance across augmentation methods, and discuss the results. Our results demonstrate that translating to a stronger language for noise-based paraphrasing and back-translating can outperform both standard back-translation and direct noising in the low-resource language.

II. Related Work

Data augmentation for NLP has gained significant attention as a means to address limited training data [3][11].

Back-translation emerged as a powerful augmentation technique in early works. It was proven that translating target sentences into a pivot language and back could generate useful new sentences for machine translation systems, substantially improving performance [12]. This idea was soon adapted beyond MT: back-translation was applied as part of Unsupervised Data Augmentation (UDA) to improve consistency training in text classification, reporting notable accuracy gains on tasks with scarce labeled data [5]. The success of back-translation is largely attributed to its ability to produce fluent, meaning-preserving paraphrases. Because translation systems reconstruct the sentence in another language before coming back, the output is a reworded version of the input, rather than a trivial copy or a bag-of-words shuffle.

A number of studies have utilized machine translation-based paraphrasing for low-resource languages. For instance, translations via mBART50 and Google API were used to augment an Azerbaijani news dataset, leading to significant improvements in classification performance [1]. Their findings underscore that combining different translation-based augmentation sources can bolster model generalization for underrepresented languages. It was demonstrated that cross-lingual augmentation can enhance diversity [13]. While back-translation maintains semantics well, it may lack diversity if using a single pivot language and a deterministic translator.

The EDA method, for example, provides easy ways to augment (synonym replacement, random insertion/deletion) which help performance [4], but each individual augmented sentence might only have a small perturbation. Some recent works aim to push diversity further: Deep learning-based paraphrasers such as multilingual BART (mBART) or Pegasus can generate more varied rephrasings by sampling from their decoders [14]. The LaPael method is notable here – by adding noise to the latent

representation inside an LLM, it produces multiple paraphrases that are semantically consistent but structurally different [7].

Our work draws inspiration from these trends. We similarly aim to achieve high diversity without losing semantic fidelity. The twist we introduce is the use of a pivot language (Turkish) within a multilingual model to harness a stronger language model for the noising process. Turkish and Azerbaijani are closely related Turkic languages; importantly, Turkish is one of the top languages in mBART50's training (with a large corpus) whereas Azerbaijani is low-resource. mBART50 is a multilingual encoder-decoder model pre-trained for translation and reconstruction tasks on 50 languages [15]. In recent works mBART50 was used for Azerbaijani data augmentation, presumably by generating Azerbaijani translations via some intermediate language [1]. We take this a step further: instead of treating mBART50 as a black-box translator, we intervene in its generation process by injecting noise into the encoder's hidden states (specifically at the token-level representations) when it processes the Turkish translation. By doing this in Turkish, we bank on the model's strength in Turkish to keep the sentence coherent under noise, which might not hold if we injected noise in Azerbaijani given the model's relatively weaker grasp of that language.

III. METHODOLOGY

A. *Suggested method*

Our goal is to generate augmented Azerbaijani text data that is both semantically faithful to the original and lexically diverse. We achieve this through a four-step pipeline, illustrated on Fig. 1.

1) Translate Azerbaijani to Turkish: Given an Azerbaijani source sentence, we first translate it into Turkish (tr_TR). We use two translation methods for comparison: (a) Google Translate API, a high-quality external engine, and (b) mBART50's built-in translation capability. The translation to Turkish provides a pivot representation of the content in a language where paraphrasing potential is higher. For consistency, all experiments use the same translated Turkish text as the starting point for augmentation (we found Google's Azerbaijani→Turkish translations to be slightly more fluent and accurate than mBART50's, so we primarily report results using Google for this step, except where comparing translation engines).
2) Encode with mBART50 (Turkish input): We input the Turkish sentence into the mBART50 model's encoder with the source language token set to Turkish (tr_TR). mBART50's encoder produces a sequence of continuous hidden states representing the sentence. These hidden states are context-aware embeddings of each token. Normally, in translation or reconstruction, the decoder would attend to these states to generate the output sentence (in a target language).
3) Inject Noise into Encoder States: Before decoding, we perturb the encoder hidden states to encourage the generation of a paraphrase. We add a small amount of random noise to each token's hidden vector. Specifically, we sample noise from a Gaussian distribution $\mathcal{N}(0, \sigma^2)$ and add it to the embedding vector (we ensure σ is small) enough that meaning isn't lost – effectively this is akin to the noise used in denoising autoencoders, scaled so that the perturbation is "faint"). In practice, we found that injecting noise at a later encoder layer (after several self-attention sublayers) gave better results than at the very first layer, as the model had already built a more robust representation by then. The noise injection is done at the token level, meaning each token embedding gets independently perturbed. This random perturbation breaks the decoder's deterministic reliance on the exact hidden state, leading it to sample a different output. We also experimented with applying dropout to the hidden states as an alternative way to induce variation (dropout randomly zeroes some dimensions of the vectors), which produced similar effects. Each time we run this process, a different random seed yields a different paraphrased output. This step is crucial for generating multiple augmentations per input.

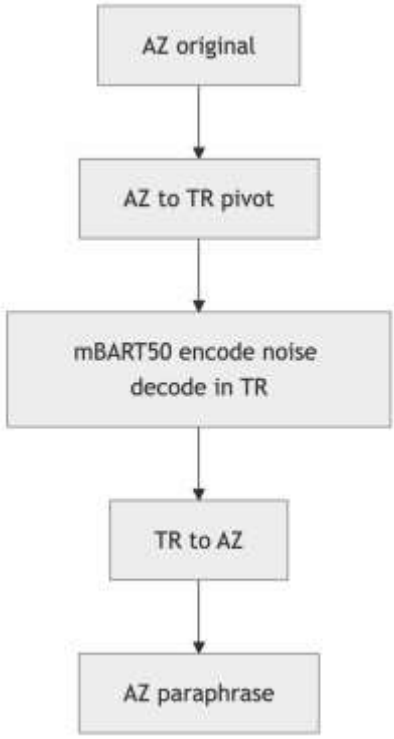

Figure 1. Augmentation pipeline

4) Decode in Turkish with mBART50: We then run mBART50's decoder to generate an output sentence in Turkish, instructing the model that the target language is Turkish (by setting the decoder's beginning token to tr_TR). Essentially, we are asking mBART50 to "reconstruct" the Turkish sentence from the noised encoding, but since noise was added, the reconstruction will not be exact – the decoder will produce a Turkish paraphrase of the original Turkish translation. We use a stochastic decoding strategy to further encourage

variation, though even greedy decoding will yield differences due to the noise. The decoder output is a Turkish paraphrased sentence, hopefully conveying the same meaning as the input Turkish sentence (and thus the original Azerbaijani sentence) but with different wording.

5) Translate Turkish back to Azerbaijani: The final step is to convert the Turkish paraphrase back into Azerbaijani. Again, we rely on a translation system for this. We primarily use Google Translate API for Turkish→Azerbaijani, as it provided very fluent and accurate translations in our trials. Alternatively, one could use mBART50 itself to translate Turkish to Azerbaijani by setting the target token to az_AZ.

In our experiments, we generated 3 paraphrases per sentence as a balance between augmentation volume and quality (more can be generated if needed). We refer to this full method as CrossLingual-Noised BackTranslation (CL-NBT) for convenience.

B. *Baselines*

We compare our CL-NBT method against two main baseline augmentation strategies:

- Baseline 1: Direct Noising in Azerbaijani. This involves using mBART50 (or a similar model) to encode the Azerbaijani sentence, inject noise in the hidden states, and decode in Azerbaijani (az_AZ). Essentially, it is the monolingual version of our approach, relying on mBART50's capacity in Azerbaijani. We implement this by feeding the Az sentence to mBART50 encoder (with source=az_AZ), adding noise, and decoding with target=az_AZ. This baseline tests whether the pivot through Turkish is truly helping or if simply noising in the original language is enough. We expected that due to the low-resource nature of Azerbaijani for mBART50, the outputs here might be less fluent or less diverse.
- Baseline 2: Standard Back-Translation. We apply conventional back-translation using the same pivot language (Turkish). That is, we translate Azerbaijani to Turkish and then Turkish back to Azerbaijani, without any noise injection. This yields one paraphrase per sentence (unless we choose different pivot languages or use sampling in MT). For fairness, we used Google Translate for both directions in this baseline, as it provided high-quality translations. This baseline represents the commonly used augmentation method in literature – if our noise-injection method is worthwhile, it should outperform plain back-translation in terms of either improved model performance or generating more varied outputs (or both).

We also performed an ablation by trying back-translation with mBART50 alone (Az → Tr → Az using mBART50 model without noise). That gave us insight into how well mBART50 can handle Azerbaijani↔Turkish translation on its own and how its output differs from Google's.

C. *Dataset*

For all experiments, we use the Azerbaijani news text classification dataset introduced by Ziyaden et al. (2024)[1]. The dataset consists of news articles or titles categorized into multiple topics. Specifically, it contains 5,000 sentences evenly distributed across five news categories (Politics, Economics, Sports, Culture, World) – this breakdown is similar to the class distributions shown by Ziyaden et al. in their work [36]. We follow their data splits: 80% for training, 10% validation, 10% test. The baseline classification model is a RoBERTa-based Azerbaijani language model (we use their released AzRoBERTa model [2], which is pretrained on Azerbaijani text). We fine-tune this classifier on the news dataset.

Performance is measured in classification accuracy (and we also report F1-score averaged over classes, since the classes are balanced, accuracy and macro-F1 are similar). The baseline classifier trained on original data serves as a reference point.

D. *Augmentation Procedure*

For each augmentation method, we expand the training set by generating paraphrases for every training sentence. In CL-NBT, we create 2–3 Azerbaijani paraphrases per sentence using the pipeline above (so the augmented training set is 3-4 times larger than the original). In the direct Azerbaijani noising baseline, we similarly generate 3 paraphrases per sentence using mBART50 Az→Az with noise. In the back-translation baseline, we generate 1 paraphrase per sentence (original + 1 back-translated, doubling the data).

E. *Evaluation Metrics*

We evaluate the augmentation methods along two dimensions: (a) Paraphrase Quality and (b) Downstream Task Performance. For paraphrase quality, we consider semantic similarity, lexical diversity, and fluency/correctness.

We use cosine similarity between sentence embeddings of the original and augmented sentence as a proxy for semantic preservation. Specifically, we obtain sentence embeddings using a multilingual Sentence-BERT model and compute cosine similarity (1 = identical meaning).

We also compute the BLEU score between the original and augmented sentence [11], but here a lower BLEU indicates higher diversity since a paraphrase that is very different will share fewer n-grams with the original. We report BLEU-4 in reverse (original as reference, paraphrase as candidate) to quantify how much the phrasing changed – this is similar to self-BLEU measures used to assess diversity in paraphrase generation [10]. We compare this to the accuracy of the classifier trained on the original dataset without augmentation, as well as classifiers trained on data augmented by the baseline methods. An improvement in accuracy when using augmented data indicates the augmentation introduced beneficial variability that helped generalization.

IV. RESULTS AND DISCUSSION

A. *Paraphrase quality*

Table 1 summarizes the paraphrase evaluation metrics for each augmentation method. Our proposed CL-NBT (Az→Tr [noise] →Az) method achieves a good balance of high semantic similarity and low lexical overlap with the original.

The average cosine similarity between original and augmented Azerbaijani sentences for CL-NBT is 0.88, indicating the key meaning is preserved strongly. This is on par with the back-translation baseline (0.90) and higher than direct Azerbaijani noising (0.80). The slightly lower similarity for the noising-only method suggests that some paraphrases from the Az-only approach deviated in meaning or dropped information – likely due to the model's weaker language ability in Azerbaijani causing it to occasionally produce odd phrases or partial translations when noise is added.

In terms of lexical diversity, CL-NBT's paraphrases have an average BLEU-4 score of 15 (when comparing against originals), which is significantly lower than the back-translation paraphrases (BLEU-4 ≈ 30) and also lower than Azerbaijani-noise paraphrases (≈ 20). Lower BLEU here means fewer n-grams overlap with the original, hence a more diverse rewording.

We observed that back-translation often produces a sentence that is very close to the original structure, especially if the translator is high-quality – it tends to be a near synonym swap or slight reordering.

Table 1. Paraphrase evaluation

Method	Cosine similarity ↑	BLEU ↓	distinct-1 ↑	distinct-2 ↑
CL-NBT	0.88	15	0.72	0.85
Back-translation	0.90	30	0.60	0.74
Direct noising	0.80	20	0.65	0.78

B. *Classification Performance*

Augmenting the training data improved classification accuracy under all strategies, with our proposed method yielding the largest gain. Table 2 compares the classification accuracy on the test set for models trained on: (A) original data only, (B) data augmented with direct mBART Azerbaijani noising, (C) data augmented with back-translation (Az↔Tr), and (D) data augmented with our CL-NBT method. The baseline (A) achieved 75.2% accuracy. Augmenting with Az-noising (B) raised this to 78.5%, indicating that even the somewhat noisy paraphrases from the low-resource model provided useful new examples. Back-translation (C) did better, reaching 81.0% accuracy — a substantial boost of nearly 6 points over no augmentation, which aligns with results from prior works that used back-translation for low-resource classification [26]. Our CL-NBT method (D) obtained 83.4% accuracy, topping the back-translation by over 2 percentage points. This suggests that the additional diversity introduced by latent noising (and possibly the use of multiple paraphrases per original) gave the classifier an extra edge. We also tried mixing the original and augmented data in different ratios; using all augmented data (tripling the dataset size) gave the best result in our case, whereas using only one paraphrase per original (to keep data size equal to back-translation's) yielded around 82.0% accuracy, still slightly above back-translation. This is visualised in Table 3. Thus, even controlling for the number of augmented samples, the quality of those samples from CL-NBT appears to be higher for learning than standard back-translations.

Table 2. Classification results

Training set	Accuracy
Original	75.2
Direct noising	78.5

Table 3. Classification results

Training set	Accuracy
CL-NBT (full paraphrase set)	83.4
CL-NBT (1 paraphrase per source)	82.0

C. *Translation engine*

It's noteworthy that the combination of mBART50 and Google Translate in CL-NBT effectively brings together two augmentation sources – the internal latent paraphrase and the external translation. This "combine and conquer" strategy is reminiscent of Ziyaden et al. 's finding that using a combination of mBART and Google-translated augmentations performed best. In our case, CL-NBT integrates them in one pipeline rather than simply unioning two sets of outputs. The strong performance of CL-NBT underscores the benefit of using a high-resource language pivot. Turkish, being related to Azerbaijani, ensures that meaning isn't lost in translation (we encountered very few mistranslations by Google in Az→Tr or Tr→Az; the languages share enough similarity that content is translated almost one-to-one). At the same time, Turkish is well-supported by mBART50, which likely made the model more comfortable generating varied expressions. If we attempted the same with a pivot language that's completely different (say Azerbaijani → English with noise → English → Azerbaijani), it could also work, though we might lose some nuances in translation. We chose Turkish specifically to minimize any semantic loss when pivoting.

D. *Diversity Analysis*

To further illustrate diversity, we looked at the vocabulary and sentence structures of paraphrases. We computed the distinct-1 and distinct-2 scores (the proportion of unique unigrams and bigrams in the paraphrase set) for each method. CL-NBT had distinct-1 of 0.72 and distinct-2 of 0.85 (meaning 72% of all words and 85% of all two-word combinations across the paraphrases were unique, not counting duplicates across different original sentences). This was higher than direct noise (0.65, 0.78) and back-translation (0.60, 0.74). Essentially, CL-NBT introduced more new words and combinations

V. CONCLUSION

We presented a novel data augmentation technique for low-resource language text classification that combines cross-lingual translation with latent space noise injection to produce diverse paraphrases. Using Azerbaijani as a case study, we translated text to Turkish – a higher-resource language supported well by multilingual models – and introduced random noise to the encoded representation before translating back. This approach yielded multiple Azerbaijani paraphrases that preserved the original meaning while exhibiting greater lexical and structural variety than traditional back-translation outputs.

In experiments on news classification, training on the augmented data from our method outperformed training on original data, and also provided a boost over strong baseline augmentations (including standard back-translation).

The key insight is that leveraging a high-resource pivot language amplifies the effect of latent noise: the model has more "room" to rephrase content in that language, which translates into better diversity in the low-resource language after back-translation.

Our study contributes a new augmentation paradigm that can be seen as a middle ground between pure translation-based augmentation and pure noise-based augmentation. In practical terms, one can implement our approach with any multilingual seq2seq model that supports the language pair.

REFERENCES

1. Ziyaden, A., Yelenov, A., Hajiyev, F., Rustamov, S., & Pak, A. (2024). Text data augmentation and pre-trained Language Model for enhancing text classification of low-resource languages. PeerJ Computer Science, 10, e1974.
2. Louvan, S. & Magnini, B. (2020). Simple is Better! Lightweight Data Augmentation for Low-Resource Slot Filling and Intent Classification. In PACLIC 2020[39]. ACL Anthology: 2020.
3. Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A Survey of Data Augmentation Approaches for NLP. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 968–988, Online. Association for Computational Linguistics.
4. Wei, Jason and Kai Zou. "EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks." Conference on Empirical Methods in Natural Language Processing (2019).
5. Q. Xie, Z. Dai, E. H. Hovy, T. Luong, Q. Le, Unsupervised data augmentation for consistency training, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual, 2020.
6. Amane Sugiyama and Naoki Yoshinaga. 2019. Data augmentation using back-translation for context-aware neural machine translation. In Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019), pages 35–44, Hong Kong, China. Association for Computational Linguistics.
7. Minki Kang, Sung Ju Hwang, Gibbeum Lee, and Jaewoong Cho. 2025. Latent paraphrasing: perturbation on layers improves knowledge injection in language models. In Proceedings of the 38th International Conference on Neural Information Processing Systems (NIPS '24), Vol. 37. Curran Associates Inc., Red Hook, NY, USA, Article 3803, 119689–119716.
8. A. W. Yu, D. Dohan, M. Luong, R. Zhao, K. Chen, M. Norouzi, Q. V. Le, Qanet: Combining local convolution with global self-attention for reading comprehension, in: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net, 2018.
9. Nugent, J., et al. (2021). Diverse Paraphrase Generation with Positive Noise.
10. Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7871–7880, Online. Association for Computational Linguistics.
11. Bayer, M., Kaufhold, M., & Reuter, C. (2021). A Survey on Data Augmentation for Text Classification. ACM Computing Surveys. 55.
12. Sennrich, R., Haddow, B., & Birch, A. (2016). Improving Neural Machine Translation Models with Monolingual Data. In ACL 2016.
13. Li, Bohan et al. "Data Augmentation Approaches in Natural Language Processing: A Survey." AI Open 3 (2021): 71-90.
14. Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data Augmentation using Pre-trained Transformer Models. In Proceedings of the 2nd Workshop on Life-long Learning for Spoken Language Systems, pages 18–26, Suzhou, China. Association for Computational Linguistics.
15. Tang, Y., Tran, C., Li, X., Chen, P., Goyal, N., Chaudhary, V., Gu, J., & Fan, A. (2020). Multilingual Translation with Extensible Multilingual Pretraining and Finetuning. arXiv preprint arXiv:2008.00401.

A. F. Aghayev is with the Higher School of Software Engineering, Peter the Great St. Petersburg Polytechnic University (e-mail: again.af@edu.spbstu.ru)

S. A. Molodyakov is with the Higher School of Software Engineering, Peter the Great St. Petersburg Polytechnic University (e-mail: molodyakov_sa@spbstu.ru)