Borderless Azerbaijani Processing: Linguistic Resources and a Transformer-based Approach for Azerbaijani Transliteration

Reihaneh Zohrabi[⋆] **Mostafa Masumi**[⋆] **Omid Ghahroodi**[†] **Parham AbedAzad**[†]
Hamid Beigy **Mohammad H. Rohban** **Ehsaneddin Asgari**
Language Processing and Digital Humanities Laboratory
Computer Engineering Department, Sharif University of Technology, IR
AI Innovation Center, Data:Lab, Volkswagen AG, Munich, Germany
{zohrabi,m.masumi,omid.ghahroodi98,parhamabedazad,beigy,rohban}@sharif.edu and asgari@berkeley.edu

Abstract

Recent advancements in neural language models have revolutionized natural language understanding. However, many languages still face the risk of being left behind without the benefits of such advancements, potentially leading to their extinction. One such language is Azerbaijani in Iran, which suffers from limited digital resources and a lack of alignment between spoken and written forms. In contrast, Azerbaijani in the Republic of Azerbaijan has seen more resources and is not considered as low-resource as its Iranian counterpart. In this context, our research focuses on the computational progress made in Iranian Azerbaijani language. We propose a transliteration model that leverages an Azerbaijani parallel dataset, effectively bridging the gap between the Latin and Persian scripts. By enabling seamless communication between these two scripts, our model facilitates cultural exchange and serves as a valuable tool for transfer learning. The effectiveness of our approach surpasses traditional rule-based methods, as evidenced by the significant improvements in performance metrics. We observe a minimum 15% increase in BLEU scores and a reduction of at least 1/3 in edit distance. Furthermore, our model's online demo is accessible at https://azeri.parsi.ai/.

1 Introduction

The Azerbaijani language belongs to the Turkish language family and is spoken in two distinct dialects, primarily in the Republic of Azerbaijan and the Azerbaijani regions of Iran. While these dialects exhibit minor variations, they share a considerable linguistic commonality, making it feasible to transition from one dialect to the other by adapting existing letters and phonetic elements. This linguistic compatibility enables seamless communication between the two dialects, facilitating the utilization of their shared linguistic resources.

Azerbaijani is the official language of the Republic of Azerbaijan. However, according to statistics from the Population and Housing Census conducted by the Statistical Center of Iran[1] in 2015, the population of East and West Azerbaijan, Ardabil, and Zanjan provinces in Iran was approximately 9.5 million. Iranian Azerbaijani, spoken in these regions, is recognized as an ethnic spoken language without an established official writing system, rendering it a low-resource language.

Unlike Iranian Azerbaijani, the Azerbaijani language has recieved substantial attention within the field of natural language processing. For instance, Suleymanov et al. (2019) employs machine learning techniques, including decision trees, support vector machines, and Naive Bayes, to categorize Azerbaijani texts for various applications, such as news classification, sentiment analysis, and recommender systems. Moreover, Akhundova (2021) introduces models that incorporate both rule-based and machine learning approaches for Azerbaijani named entity recognition. Additionally, while Azerbaijani benefits from a repository[2] dedicated to collecting data and computing resources, which greatly simplifies computational tasks related to the language, no similar resources or initiatives had existed for Iranian Azerbaijani until the recent pioneering work (Marzia et al., 2023). This innovative research effort represents the initial and significant step toward establishing essential NLP resources for Iranian Azerbaijani, encompassing the development of standard datasets and starter models for various NLP tasks. It plays a pivotal role in preserving the language and culture of Iranian Azerbaijani. In this research, we have developed a transliteration model bridging the Iranian and Azerbaijani variants, facilitating the processing of Iranian multilingual texts. We refer to the Iranian variant of

⋆, † Equal contribution

[1] https://www.amar.org.ir/

[2] https://github.com/alexeyev/awesome-Azerbaijani-nlp

Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 2: Short Papers), pages 175–183
November 1–4, 2023.

Azerbaijani spoken in Iran and the Azerbaijani variant spoken in Azerbaijan. Due to the scarcity of linguistic resources and prior computational work for the Iranian variant, challenges arise when processing this language. There is a lack of pre-processing tools and pre-trained language models, even within extensive multilingual resources like Facebook's FastText (Bojanowski et al., 2017). For the Azerbaijani variant, there are existing works (Huseynov et al., 2021) that have utilized word2vec (Mikolov et al., 2013) and Glove (Pennington et al., 2014) for vocabulary representation. Furthermore, the existence of different scripts for the Iranian variant has led to data source multiplicity, necessitating script unification through pre-processing. "Suzelrin Vahid Yazilishi: Calligraphy Style and Orthographic Culture of Azerbaijani Turkish (Consistent Spelling of Words)" describes the currently approved Iranian variant of the Azerbaijani script (Perso-Arabic script).

Our primary contributions include:

I. The creation of a parallel dataset encompassing both Iranian and Azerbaijani variants of Azerbaijani.

II. A thorough analysis of existing transliteration tools for Iranian Azerbaijani.

III. The development of a transliteration model, applied in (Marzia et al., 2023), to enhance resources for Iranian Azerbaijani.

We facilitate access to our dataset and code on GitHub and Hugging Face via the provided links [3], [4].

2 Related Works

Considering the significance of transliteration between the two variants of the Azerbaijani language, various efforts have been made to address this challenge, exemplified by tools like **AzConvert**[5] and the **Azalpha plugin**. However, these prior approaches rely on predefined rules and exhibit limitations in terms of accuracy, adherence to standard scripts, and contextual awareness. The rule-based methodologies employed can become overly intricate due to the inherent ambiguities across languages, with one such challenge being homographs. In languages like Azerbaijani, homographs can introduce errors in rule-based transliteration. Furthermore, the lack of adherence to standard scripts in numerous existing Iranian Azerbaijani sources renders rule-based methods unsuitable for handling such content.

There are other works for transliteration between different languages, such as **polyglot**[6] (Chen and Skiena, 2016), a natural language processing tool with various applications, including transliteration between different languages. This tool can support transliteration between 69 different languages, including Azerbaijani and Persian, but there is no such tool for Iranian Azerbaijani. Other studies focus on transliteration between closely related languages, which we categorize into three groups:

2.1 Rule-based methods:

In these methods, transliteration is accomplished through the application of pre-defined rules. Bhalla et al. (2013) employs a rule-based model for syllabification and statistical techniques, specifically for translating English into Punjabi. Similarly, Ahmadi (2019) adopts a rule-based approach for converting the two primary written systems of the Surani Kurdish language (Middle Kurdish) into each other. This is achieved by identifying characters in words, resolving potential ambiguities, and mapping them to the target text. Moreover, Oh and Choi (2002) utilizes pronunciation and content rules for the transliteration process, specifically from English to Korean.

2.2 PGM methods:

These methods typically employ probabilistic modeling techniques, aiming to maximize the probability of parallel word pairs between source and target languages. For instance, in the work of Pingali et al. (2008), a sophisticated statistical transliteration approach is introduced. Remarkably, this technique stands out for its language-independence, making it applicable to a wide range of language pairs. In the initial phase, it leverages hidden Markov model alignment, a powerful tool for capturing linguistic patterns and associations. Subsequently, in the next phase, the approach employs conditional random fields, further enhancing its transliteration capabilities. This multi-phase approach not only contributes to the robustness of the transliteration model but also allows for adaptability across various language

[3] https://github.com/language-ml/Borderless-Azerbaijani-Processing

[4] https://huggingface.co/datasets/language-ml-lab/parallel_azeri_dataset

[5] https://github.com/mousamk/azconvert

[6] https://github.com/aboSamoor/polyglot

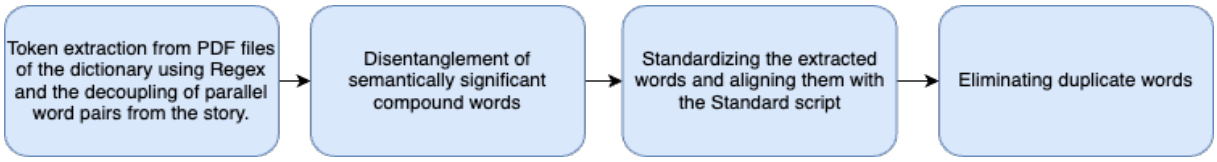

Figure 1: Overview of Preprocessing Steps Applied to the Dataset

contexts, making it a valuable resource in the field of cross-lingual text processing and machine translation research.

2.3 Neural network methods:

These methods include models based on LSTM and transformers. (Shao and Nivre, 2016) is an example of using neural networks for transliteration between English and Chinese. In this paper, the network architecture includes a convolutional layer to extract character-level information and a recurrent layer to process the text. Mahdi Mahsuli and Safabakhsh (2017) uses the encoder-decoder approach with the attention mechanism for transliteration from English to Persian, where instead of randomly initializing the weights of the encoder network, they use the vector representation of the words of the source language as the initial value for the weights. In general, sequence-to-sequence models with an attention mechanism use a recurrent neural network encoder to learn input text representations and a recurrent neural network decoder to generate output sequences from hidden representations created by the encoder. The attention mechanism (Bahdanau et al., 2014) allows the decoder to focus on different input parts for each time step in the output sequence. It can be seen as similar to the alignment mechanism used in traditional statistical translation models. Rosca and Breuel (2016) also uses this type of model with LSTM and GRU.

3 Materials and Methods

3.1 Data collection and preprocessing

Training of the transliteration model necessitates parallel datasets in both the source and target languages. However, due to the unavailability of such a dataset for processing purposes, we collaborated with the Azerbaijani Turkish Language Department of the University of Tabriz (Faculty of Persian and Foreign Languages). Through their cooperation, we were able to gather sources containing parallel texts in both Latin and Persian scripts. Our primary contribution lies in generating resources for this low-resource language.

Latin	Iranian	Example
ü	ۆ	Gün-گۆن
u	و	vurdu-ووردو
o	ۇ	yol-یۇل
ö	ؤ	Göz-گؤز
I	ێ	baxdı-باخدێ
I	ی	bir-بیر
e	ئ	dedi-دئدی

Table 1: Potentially misused vowel characters of Iranian Azerbaijani.

These sources encompass a portion of the Azerbaijani epic 'Koroğlu' written in both languages and utilizing the standard Azerbaijani script. Additionally, for out-of-domain evaluation, we obtained the Azerbaijani folklore tale 'Qurqud Dədə' written in both scripts, with the assistance of the aforementioned group.Additionally, we obtained the 'Azerbaycan sözlüğü' (Esmayil Jafarli, 2013) dictionary, which comprises 120,000 original words, Azerbaijani translations, place names, and the names of renowned individuals and poets, compiled in three PDF volumes.

After collecting the data above, we performed a series of word extraction and pre-processing operations to ensure that the words closely align with the standard script. The pre-processing procedures applied to the data typically adhered to the subsequent steps illustrated in Figure 1.

In the data normalization stage, we carefully assessed specific conditions to ensure the accurate substitution of Iranian Azerbaijani vowel characters with their corresponding Latin counterparts. It is important to note that the Iranian Azerbaijani language has multiple vowel characters that resemble the character "و" (such as "ؤ", "ۆ" and "ۇ"). However, in numerous existing sources, these vowels are often represented by the initial form, leading to inaccuracies in the written Iranian

Azerbaijani words. Therefore, we implemented pre-processing techniques to precisely replace the vowels based on their Latin forms, thereby enhancing the accuracy of the word representations. The table 1 shows vowel characters of the Iranian Azerbaijani language that must be observed in the standard script and may be used incorrectly.

We have summarized the information from the final dataset in Table 2, showcasing the count of parallel word pairs available in both languages. Additionally, you can find sample examples from this dataset in Appendix 1.1. In Iranian variant, the average token length is 7.34 characters, while in Azerbaijan variant, it is 7.73 characters. These word pairs serve as valuable resources for training our transliteration model. It's important to emphasize that the limited availability of linguistic resources for this language significantly hinders the applicability of data augmentation techniques.

	all tokens	**after pre-processing**
Dictionary	120000	72000
Koroğlu	700	500
Total	120700	72500

Table 2: Summary of the number of Parallel Word Pairs in the Final Azerbaijani Dataset.

To train the transliteration model, the data was divided into 5 clusters based on character patterns (using n-grams with lengths of 2 to 6). Clustering helped create distinct training and validation sets by reducing similarity between sub-words. Mean and standard deviation analysis assessed model overfitting and stability.

3.2 Model

The Transformer model, known for its powerful architecture in natural language processing tasks, including transliteration, has shown exceptional performance in capturing long-range dependencies. This makes it particularly suitable for transliteration tasks involving languages like Iranian and Azerbaijan variants of the Azerbaijani language. It employs a self-attention mechanism that allows the model to focus on different parts of the input sequence, capturing contextual relationships crucial for accurate transliteration.

The Transformer model consists of an encoder-decoder architecture. The encoder generates a representation capturing relevant information from the source language sequence, while the decoder uses this representation to generate the transliterated output sequence. By utilizing positional encodings, multiple stacked layers, and large-scale parallel data during training, the Transformer model effectively learns the mappings between input and output sequences.

Given its ability to handle sequence-to-sequence tasks and capture long-range dependencies, the Transformer model has become a prominent choice for transliteration tasks across various languages. In our research on two variants of Azerbaijani transliteration, we employ the Transformer model as the primary framework. Our model operates at the character level, where each character serves as an individual token. Token embeddings are initialized with random values to capture a wide range of linguistic nuances.

To assess the performance of the trained model, we measure the BLEU score (Papineni et al., 2002) and Levenshtein distance. During training, cross-validation is employed, with one cluster serving as the test set and the others as training and validation data. A ten percent subset is allocated for validation. The optimal hyperparameters for training are determined based on this setup, with further details provided in the appendix 1.2. We calculated the mean length in a collection of prefixes, suffixes, and morphemes in Iranian Azerbaijani to be near 3. We thus opted to calculate the BLEU score using an n-gram level of 3.

4 Results

The comparison results presented in Table 3 highlight the significant performance improvements achieved by the Transformer model compared to the Azconvert rule-based and polyglot statistical methods. Remarkably, the trained transliteration model attained an impressive BLEU score of 0.94 for both directions, despite the limited resources and calligraphy-related challenges. Furthermore, to provide a comprehensive evaluation and comparison with our Transformer-based approach, we introduced an LSTM-based model, which achieved BLEU scores of 0.91 and 0.92 for Persian-to-Latin and Latin-to-Persian conversions, respectively. In contrast, the Azconvert and polyglot models struggled to surpass a BLEU score of 0.79 in any transliteration direction.

Furthermore, the superiority of the transformer model extends to the minimum edit distance met-

Method	Persian to Latin			Latin to Persian		
	Min. Edit Dist. d	Avg. Len.	BLEU	Min. Edit Dist	Avg. Len.	BLEU
Transformer	**0.31 ± 0.1**	8	**0.94 ± 2.6**	**0.33 ± 0.06**	7	**0.94 ± 1.7**
polyglot	3.56	8	0.45	3.03	7	0.53
Azconvert	1.12	8	0.79	1.33	7	0.74
LSTM	0.47	8	0.91	0.43	7	0.92

Table 3: Comparison of Transliteration Methods: Persian to Latin and Latin to Persian Scripts. The table presents the minimum edit distance, average word length, and BLEU score for each method. Performance of the Transformer method was evaluated using five folds in a cross-validation setup, and the mean and standard deviation are reported.

Method	Persian to Latin			Latin to Persian		
	Min. Edit Dist. d	Avg. Len.	BLEU	Min. Edit Dist	Avg. Len.	BLEU
Transformer	**0.17**	5.2	**0.96**	**0.32**	5.3	**0.91**
polyglot	2.34	5.2	0.21	2.09	5.3	0.51
Azconvert	0.64	5.2	0.83	0.96	5.3	0.71
ChatGPT	1.49	5.2	0.66	1.28	5.3	0.65

Table 4: Out-of-Domain Comparison of Transliteration Methods: Persian to Latin and Latin to Persian Scripts. The table presents the minimum edit distance, average word length, and BLEU score for each method on the out-domain data.

ric. The model's character-level output exhibits a high level of accuracy, requiring minimal edits compared to the existing models. Additionally, the low standard deviation of the BLEU score indicates the model's stability during training, even when faced with variations in subword patterns across different cross-validation folds. The consistently high BLEU scores across all categories, coupled with excellent subword discrimination, indicate that the model effectively avoids overfitting on the training data. A detailed analysis of the trained models and their corresponding outputs is presented in the appendix 1.3 for further examination and understanding.

For out-of-domain evaluation, we employed the 'Qurqud Dədə' dataset and also utilized ChatGPT's predictions to gain additional perspectives. The evaluation results revealed that there was no significant drop in the BLEU score. In fact, for Persian-to-Latin conversion, the BLEU score increased by 2 points, while for Latin-to-Persian conversion, it decreased by just 3 points. These findings indicate that the model's performance remains robust. Detailed results can be found in Table 4.

5 Conclusion

The rapid growth of language technologies emphasizes the importance of linguistic resources and computational tasks for endangered languages like Azerbaijani in Iran. It connects theologians with Azerbaijani speakers worldwide, benefiting both Azerbaijani communities.

In this research, we undertook the collection and pre-processing of Azerbaijani data, enabling the creation of a parallel dataset for Azerbaijani in both Latin and Persian scripts. Through this effort, we trained a two-way transliteration model capable of converting between Latin and Persian scripts. Despite the inherent challenges, we achieved remarkable accuracy. This work represents a significant milestone in the advancement of Azerbaijani language technologies, and we look forward to further research in related domains to enhance the development of this language.

6 Limitations

It is crucial to recognize that this study's limitations are primarily rooted in two key factors: resource constraints and the scarcity of digitized materials available for the Iranian Azerbaijani language. These constraints have inevitably led to limitations in the depth of our training dataset.

The most conspicuous limitation is evident when we delve into certain word contexts, especially those involving infrequent words and senses. In such cases, the data available for training the model remains sparse. Consequently, effectively addressing infrequent ambiguities becomes a challenging

task.

The scarcity of data has a direct impact on the model's ability to accurately identify the spellings of ambiguous words, particularly those that were not encountered during its training phase. When faced with such unseen words, the model may struggle to provide precise predictions, as its exposure to such linguistic nuances is inherently limited.

7 Acknowledgment

We wish to express our deep appreciation to numerous individuals and organizations whose exceptional contributions have greatly enriched our research. In particular, we extend our heartfelt thanks to Mohammad Ali Sadraei for his invaluable assistance.

References

Sina Ahmadi. 2019. A rule-based kurdish text transliteration system. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 18(2).

Natavan Akhundova. 2021. Named entity recognition for the azerbaijani language. In *2021 IEEE 15th International Conference on Application of Information and Communication Technologies (AICT)*, pages 1–7.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Deepti Bhalla, Nisheeth Joshi, and Iti Mathur. 2013. Rule based transliteration scheme for english to punjabi. *arXiv preprint arXiv:1307.4300*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Yanqing Chen and Steven Skiena. 2016. False-friend detection and entity matching via unsupervised transliteration. *arXiv preprint arXiv:1611.06722*.

Esmayil Jafarli. 2013. *Azerbaijani Türkçe Sözlüğü (Azerbaijani Sözlüğü) - Turkish Dictionary*. Ehrar Publications in cooperation with Sumer Publishing, Tabriz.

Faculty of Persian and Foreign Languages. Faculty of Persian and Foreign Languages. https://literature.tabrizu.ac.ir/en.

Kamran Huseynov, Umid Suleymanov, Samir Rustamov, and Javid Huseynov. 2021. Training and evaluation of word embedding models for azerbaijani language. In *Digital Interaction and Machine Intelligence*, pages 37–48, Cham. Springer International Publishing.

Mohammad Mahdi Mahsuli and Reza Safabakhsh. 2017. English to persian transliteration using attention-based approach in deep learning. In *2017 Iranian Conference on Electrical Engineering (ICEE)*, pages 174–178.

Nouri Marzia, Amani Mahsa, Zohrabi Reihaneh, and Ehsaneddin Asgari. 2023. The language model, resources, and computational pipelines for the under-resourced iranian azerbaijani. In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2023*. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Jong-Hoon Oh and Key-Sun Choi. 2002. An English-Korean transliteration model using pronunciation and contextual rules. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Prasad Pingali, Surya Ganesh, Sreeharsha Yella, and Vasudeva Varma. 2008. Statistical transliteration for cross language information retrieval using HMM alignment model and CRF. In *Proceedings of the 2nd workshop on Cross Lingual Information Access (CLIA) Addressing the Information Need of Multilingual Societies*.

Mihaela Rosca and Thomas Breuel. 2016. Sequence-to-sequence neural network models for transliteration. *arXiv preprint arXiv:1610.09565*.

Yan Shao and Joakim Nivre. 2016. Applying neural networks to English-Chinese named entity transliteration. In *Proceedings of the Sixth Named Entity Workshop*, pages 73–77, Berlin, Germany. Association for Computational Linguistics.

Umid Suleymanov, Behnam Kiani Kalejahi, Elkhan Amrahov, and Rashid Badirkhanli. 2019. Text classification for azerbaijani language using machine learning and embedding. *CoRR*, abs/1912.13362.

A Appendix

1.1 Examples of Parallel Azerbaijani Dataset

Table 5 presents several examples from the prepared dataset, illustrating parallel word pairs in both variants of the Azerbaijani language.

Latin	Iranian
eşitmək	ائشیتمک
özgə	اؤزگه
tökülüşmək	تؤکۆلۆشمک
Parildamaq	پاریلداماق

Table 5: Examples of parallel word pairs in Azerbaijani language.

Parameter	Value
Batch size	128
Number of attention heads	8
Feed-forward dimension	512
Number of encoder layers	4
Number of decoder layers	4
Loss	Cross-entropy
Optimizer	Adam
Learning rate	1e-4

Table 6: Transformer hyperparameters.

Parameter	Value
Insertion cost	0.5
Deletion cost	1
Substitution cost	2

Table 7: Edit distance hyperparameters.

1.2 Hyperparameters

The model training process is optimized with the following set of hyperparameters provided in table 6. Also, table 7 provides the test parameters used to calculate the minimum edit distance.

1.3 Model Output Analysis and Review

The table 8 showcases examples of transliteration model outputs for words between the Latin and Iranian scripts. The correct spellings of the words are presented in the first two columns, while the corresponding model outputs are displayed in the subsequent columns. Analyzing these outputs allows us to identify the strengths and weaknesses of each Azerbaijani transliterator.

The Polyglot model: Since there is no direct mapping between the Latin and the Iranian scripts, we have employed a linguistic bridge in this model. Consequently, the model does not generate special Azerbaijani vowels. Substituting unfamiliar vowels with Persian counterparts often results in unfamiliar words. Even for common words shared between Azerbaijani and Persian, like the word "ترفیع" in table 8, the model's

Latin Script	Iranian Script	Direction	Azconvert	polyglot	Our approach
oğlan	اوغلان	Persian to Latin	uğlan	aoqholan	oğlan
		Latin to Persian	اوغلان	وگلان	اوغلان
tərfi	ترفیع	Persian to Latin	tərfi	trfiya	tərfi
		Latin to Persian	ترفی	تارفی	ترفیع
düşünürəm	دۏشۇنۏرم	Persian to Latin	düşünürm	doşonorm	düşünürm
		Latin to Persian	دوشونوره م	دوشونورام	دۏشۇنۏرم
baxdı	باخدێ	Persian to Latin	baxdi	bachdi	baxdı
		Latin to Persian	باخ دی	بخدی	باخدێ
gördü	گؤردۏ	Persian to Latin	gördü	qordo	gördü
		Latin to Persian	گؤردو	گوردو	گؤردو
uzun	اوزون	Persian to Latin	uzun	aozon	uzun
		Latin to Persian	اوزون	وزون	اوزون
üzün	اۏزۏن	Persian to Latin	üzün	aozon	üzün
		Latin to Persian	اوزون	وزون	اۏزۏن
dedi	دئدی	Persian to Latin	dedi	dedi	dedi
		Latin to Persian	دئدی	دادی	دئدی
ayrılıq	آیرتلتق	Persian to Latin	ayrılıq	ayriliq	ayrılıq
		Latin to Persian	آیری لیق	ایرلیگ	آیرتلتق

Table 8: Sample Outputs of Transliteration Models for transliterating Azerbaijani Words between two variants.

performance was unsatisfactory. Furthermore, this model operates at a slower speed compared to other models.

The Azconvert model: This model does not have some Iranian Azerbaijani vowels such as "ۆ" and "ێ" which can be seen in table 8 for the words "دۆشۆنۆرم", "باخدێ" and "گؤردۆ". One of the advantages of this rule-based model is that it uses more appropriate writing for better readability; for example, for the word "آیرێلێق", separating the sub-words "آیری" (separate) and "لیق" (infinitive noun suffix) helps for its readability. This method does the same for this word.

The Transformer model: The transformer model can produce a variety of consonant and vowel patterns due to various learning data from the dictionary. Therefore, the trained model is very consistent with the standard script and can recognize words that are similar in appearance. However, sometimes it does not correctly recognize the spelling of ambiguous words that it did not see in the training, such as "اسیل" in the second line of table 9, which should be "اصیل".

It is important to highlight that the trained model can also be utilized for the transliteration of entire sentences since there is a one-to-one correspondence between words in both languages. Examples of sentence transliterations are illustrated in tables 9 and 10.

Latin	Persian
türkün dili tək sevgili istəkli dil olmaz	تۆرکۆن دیلی تک سئوگیلی ایستکلی دیل اولماز
özgə dili qatsan bu əsil dil əsil olmaz	اؤزگه دیلی قاتسان بو اسیل دیل اسیل اولماز
hər kəsin fikir və söz azadliği vardir	هر کسین فکر وه سؤز آزادلیغی واردیر
necə görsənir	نئجه گؤرسهنیر
mən onu dünən göndərdim	من اونو دۆنن گؤندردیم

Table 9: Sample Outputs of Azerbaijani Sentences transliterated from Latin to Persian Script.

Persian	Latin
سئوگیلیم عشق اولماسا دۆنیا بۆتۆن افسانه دیر	sevgilim əşəq olmasa dünya bütün əfsanə dir
گؤر خبرلر وار یا یوخ	gör xəbərlər var ya yox
دۆز دئییرسن	duz deyirsən
چوخ گؤزلدیر	çox gözəldir

Table 10: Sample Outputs of Azerbaijani Sentences transliterated from Persian to Latin Script.