School of Information Technology and Engineering at the ADA University

School of Engineering and Applied Science at the George Washington University

LEMMATIZATION FOR AZERBAIJANI LANGUAGE USING THE ATTENTION MECHANISM

A Thesis
Presented to the Graduate Program of Computer Science and Data Analytics
of the School of Information Technology and Engineering
ADA University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Computer Science and Data Analytics
ADA University

By
Nurlan Imanov

Supervisor Dr. Samir Rustamov

April 2024

THESIS ACCEPTANCE

This Thesis by: Nurlan Imanov
Entitled: *Lemmatization for Azerbaijani language using Attention Mechanism*

has been approved as meeting the requirement for the Degree of Master of Science in Computer Science and Data Analytics of the School of Information Technology and Engineering, ADA University.

Approved:

(Adviser)	(Date)
(Program Director)	(Date)
(Dean)	(Date)

ACADEMIC INTEGRITY STATEMENT

"I affirm that this is my own work, I attributed where I used the work of others, I did not facilitate academic dishonesty for myself or others, and I used only authorized resources for my Thesis, per the ADA University Academic Integrity requirements. If I failed to comply with this statement, I understand consequences will follow my actions. Consequences may range from failing the course to expulsion from the program/university and may include a transcript notation."

Nurlan Imanov		24.04.2024
(Full Name)	(Signature)	(Date: DD.MM.YY)

ABSTRACT

Lemmatization is often handled using rule-based methods, which work well for many languages. However, it struggles with agglutinative languages such as Azerbaijani, where writing all the necessary rules can be challenging and sometimes impossible. While preprocessing steps like lemmatization and stemming are becoming less crucial in well-studied languages such as English, they remain critical for agglutinative languages with complex structures such as Azerbaijani. Recently, the current state-of-the-art hybrid lemmatizer for Azerbaijani has improved sentiment analysis results by 8.9%. This improvement underscores the continued importance of these preprocessing steps in agglutinative languages. This thesis introduces a novel approach to lemmatization for the Azerbaijani language utilizing an attention-based sequence-to-sequence model, achieving 98% character-level accuracy across a 91941-word sample and 96% word-level accuracy, surpassing the 95% word-level accuracy of the of current state-of-the-art hybrid lemmatizer. This approach demonstrates the potential to replace rule-based or hybrid lemmatization methods with a fully machine-learning-based model, eliminating the need to manually write complex rules for agglutinative languages.

Contents

List of Figures

List of Tables

LIST OF ABBREVIATIONS

Abbreviation	Explanation
POS	Part Of Speech
LR	Logistic Regression
SVM	Support Vector Machine
LSTM	Long Short-Term Memory
NLP	Natural Language Processing

1 Introduction

Lemmatization is a crucial step in the preprocessing of textual data and has traditionally been approached with rule-based methods that have shown considerable success across various languages. However, these methods often encounter limitations when applied to agglutinative languages such as Azerbaijani, where the morphological complexity requires considering an extensive range of possible rules which is often challenging and sometimes impractical. While languages like English may not heavily rely on such preprocessing steps due to the vast amount of research and resources available, agglutinative languages, with their complex and layered linguistic structures, continue to depend on effective lemmatization to improve text analysis accuracy. In agglutinative languages, words are formed by combining morphemes, each carrying unique meaning and grammatical roles. For example, in Azerbaijani, the word 'elektrikləşdirdiklərdinizdənsinizmi'—considered one of the longest words in the language stems from the lemma 'elektrik,' with eight suffixes added. For NLP applications, understanding that 'elektrikləşdirdiklərdinizdənsinizmi' conveys a meaning closely related to 'elektrik' is profoundly challenging and demands extensive contextual understanding and examples. The potential to append nearly limitless suffixes dramatically increases vocabulary size, further complicating text processing. This thesis proposes a novel lemmatization methodology for the Azerbaijani language, utilizing an attention-based sequence-to-sequence model to show that the traditional rule-based and hybrid approach can actually be replaced by the fully-machine learning approach that eliminates the need to write the complex rules which is challenging for agglutinative languages. The current state-of-the-art lemmatization has improved sentiment analysis results by 8.9% for Azerbaijani, demonstrating the significant impact of effective preprocessing. The current research aims to build upon these developments by achieving 98% character-level accuracy across a 91941-word sample and 96% word-level accuracy, surpassing the 95% word-level accuracy of the of current state-of-the-art hybrid lemmatizer. By introducing this attention-based sequence-to-sequence approach, this study not only seeks to advance lemmatization practices for Azerbaijani but also to explore its applicability to other agglutinative languages, potentially revolutionizing language processing techniques in this challenging linguistic category.

1.1 Definition of the Problem

In the structure of agglutinative languages, the potential to append nearly infinite suffixes to a word's lemma substantially complicates natural language processing tasks. This characteristic is particularly challenging for low-resource languages such as Azerbaijani, where the availability of labeled data is limited. The limited dataset restricts the algorithm's ability to recognize that the lemmatized form of a word conveys a similar meaning to its inflected forms, and this identification difficulty increases as more suffixes are added. The crucial role of lemmatization becomes evident when considering its impact on model accuracy. For instance, incorporating lemmatization in processing Azerbaijani text for sentiment analysis tasks has proven to enhance accuracy by 8.9%. Therefore, addressing lemmatization challenges is crucial in improving NLP outcomes for agglutinative languages. Traditional rule-based methods, while effective in many cases, fall short when they encounter cases not covered by existing rules or when faced with linguistic exceptions. Moreover, current state-of-the-art hybrid lemmatizers rely on machine learning-based part-of-speech tagging, which can misidentify a word's POS and subsequently its

lemma, leading to inaccuracies. A fully machine learning-based approach can eliminate the need to manually write extensive rules, instead learning to generalize from labeled data and adapting to exceptions that are otherwise difficult to handle through rule-based methods.

1.2 Objective of the Study

The primary goal of the study is to demonstrate that machine learning can efficiently replace traditional rule-based and hybrid methods for lemmatization, particularly in the context of agglutinative languages, which exhibit significant morphological complexity and natural exceptions. The traditional rule-based approaches, while suitable for languages with simpler morphological structures like English, often struggle with agglutinative languages due to the presence of complexity. This study aims to show the feasibility of employing a fully machine learning-driven model that can autonomously learn the linguistic rules and exceptions directly from a labeled dataset, thereby eliminating the necessity of writing manual rules. The goal will be pursued by developing an attention-based sequence-to-sequence lemmatization model that specifically addresses the morphological complexity of the Azerbaijani language. This approach is anticipated to be a scalable solution applicable to other agglutinative languages as well. The effectiveness of the developed model will be evaluated by conducting a comparative analysis with an existing state-of-the-art hybrid lemmatizer. The importance of the lemmatization preprocessing process for the Azerbaijani language will be shown by measuring performance enhancements in sentiment analysis and word embeddings with a particular emphasis on accuracy which are critical NLP tasks for Azerbaijani. To ensure the accessibility and practical application of both the current hybrid state-of-the-art and attention-based sequence-to-sequence lemmatizers for Azerbaijani will be available in the "nlp_az" library, which will be developed as part of this thesis. This library will make those lemmatization techniques and other already developed NLP techniques for Azerbaijani available to developers and researchers, further facilitating the integration of this innovative approach into real-world NLP applications. By directly addressing the identified challenges in processing Azerbaijani, this study seeks to overcome the limitations of current NLP tools and set a new standard for lemmatization practices in agglutinative languages.

1.3 Significance of the Problem

The implementation of a fully machine learning-based approach to lemmatization represents a significant deviation from traditional methods that rely on extensive manual rule-setting. This novelty is crucial as it allows the system to independently learn and adapt to linguistic details from actual language use, bypassing the limitations and inflexibility of pre-coded rules. The rule-based approach is based on extensive restrictions over the morphological rules that must be either periodically updated or altered as the linguistic data change. This process is not only tedious but also error-prone as it is not possible to foresee every linguistic exception. Unlike the machine-learning model utilized in this project which learns from English text, the new model is capable of tuning itself from Azerbaijani labeled text corpus, recognizing new patterns and automatic exceptions without human intervention. The said technique is not only meant to give Azerbaijanian language a boost but the very scalability of this approach is also evident for agglutinative languages whose service is to bring down morphological complexity. The model can au-

tonomously learn and adapt itself which can be used in the agglutinative languages. As the machine-learning model is able to learn and adjust the work process by itself, it turns out to be an acceptable solution to the changes that happen in the sphere of language, along with a constant improvement of the result. The natural language processing task is performed successfully, and it is a good replacement of the classical rule-based systems.

1.4 Review of Significant Research

The value of establishing accurate lemmatization methods in Azerbaijani becomes more evident with the recent studies within computational linguistics field that are devoted to agglutinative languages. Intensive research done throughout Turkish, a fellow Turkic language, that belongs to the same linguistic group as Azerbaijani does, is valuable as we share most of the morphological features. The study by Dinçer and Karaoğlan (2003) revealed the possibility of probabilistic models to deal with the morphological complexity of Turkic languages effectively. Their discoveries prove that similar methods could be as effective for Azerbaijani that has a lot of similar features with Turkish when it comes to morphology. Created success of probabilistic models in Turkish language proves their precision to be used in Azerbaijan, when traditional rule-based systems cannot work effectively with a lot of morphological variations of Azerbaijani language.

Moreover, the work of Sever and Bitirim (2003) stresses the significance of the context in the stemming process, which is a key factor that should be taken into account for any stemming algorithm, especially in the languages with complicated inflectional structures. As a first step, they evaluated the stemming accuracy assessing the Turkish language that provides a generalized approach for applying similar methodologies to Azerbaijani language, assuring that stemming keeps in mind the context of the language.

The hybrid methods, where the decision systems mix the rule-based and statistical methods, are applicable in morphologically complex languages. Kışla and Karaoğlan (2016) got more accuracy and adaptability while trying out these methods in Turkish language as well. This kind of hybrid approach may provide a solution for the linguistic nature in the Azerbaijani language from a practical viewpoint, enabling us to apply it in the variety of NLP applications.

Unsupervised methods which combine tagging and stemming methods are also proposed as a technique with a lot of potential especially for Azerbaijani language, specifically when it comes to the cases where annotated resources are scarce. Bölücü (2017) has provided a deep insight on these advanced techniques, giving a solid theory that would useful for development of efficient and flexible stemming algorithms for Azerbaijani language.

These studies provide a basis for the current research approach, which implies the model that is stable, flexible and able to cope with the specific problems of the Azerbaijani language. The purpose of this research is to expand the existing knowledge base by increasing the accuracy and the efficiency of Azerbaijani text processing in the digital world through the integration of these important findings which potentially can increase the efficiency of NLP applications for our language.

These studies collectively inform the current research approach, suggesting a model that is robust, adaptable, and capable of handling the unique challenges posed by the Azerbaijani language. By integrating these significant insights, this research aims to contribute to the body of knowledge by enhancing the precision and efficiency of Azerbaijani text processing in digital environments.

1.5 Assumptions and Limitations

This research which concerns a machine learning based model for lemmatization of the Azerbaijani language rests on some essential principles, which limit the way how the methodology and results can be interpreted. First of all, the study assumes a high level of linguistic homogeneity within the data sources, and that the results for the training and the testing datasets do not vary significantly from each other. It is possible that narrowing the scope in this way will not be a complete reflection of regional dialects and historical forms of the language. Moreover, it is presumed that the raw data which the model is trained on are precise and adequately represent the morphology of the language used. Further, this presumption spreads to the accuracy of the hybrid lemmatizer used for annotation, though it is known that this tool has a 5% error rate. Besides, the research assumes that the results from the lemmatization of Azerbaijani can be generalized to other agglutinative languages, through the use of morphological similarities.

In addition, the research bears inherent limitations that affect its scope and applications. The overall performance of machine learning depends heavily on the data quality; thus, clean and accurate data are a highly needed in this case. The training data errors, which are 5% in the hybrid model's annotations, will cause inaccuracies in the lemmatization method and those inaccuracies will be transmitted to the learning process of the model. Though the model is created to deal with the complicated grammatical structure of Azerbaijani language; however, its ability to handle highly rare or irregular morphological patterns is not fully tested so their impact on the model's performance could be worse in situations where such anomalies are present. Moreover, the proposed model's scalability is somewhat affected by the computational resources that are available to perform the training process, and this is a factor that determines how well the model can be trained with extensive data and how complex its architecture can be. Lastly, the study implies that the results can be generalized to the other agglutinative languages, but for now it is only for Azerbaijani language. Thus, the direct implementation of the model to other agglutinative languages not experiencing simultaneous translation without the additional adjustments would be speculative and may not accomplish the tasks aimed.

These assumptions and challenges have greatly contributed to articulating the findings of the study. They actually provide the boundaries that are present at which the findings are valid, and basically show the positions where the conclusions should be used carefully when going outside of the immediate scope of our study.

2 Literature Review

Lemmatization is one of the basic natural language processing (NLP) techniques that mainly targets the reduction of words to their basic or root form. Such a sort of linguistic reduction becomes needed in making quite a few text processing applications like search optimization, text analytics, and information retrieval both faster and effective. Our literature review looks at the adaptations of lemmatization methodologies for the Azerbaijani language. Like in the case of many Turkic family languages, Azerbaijani is an agglutinative language, with words formed by extensive morpheme combination and mostly in a prefixing or suffixing manner.

The agglutinative nature of Azerbaijani is the biggest obstacle for any lemmatization system that is intended to process it. In an agglutinative structure, each morpheme has the ability of changing the meaning of a word; to these, multiple affixes can be added to a root, which, however, usually ends up with long and complex words. Such complexity proves lemmatizer's task unbelievably hard as the tool needs to distinguish between multiple suffixes (or prefixes) without contextualizing meaning of the base word. Additionally, the morphological richness of Azerbaijani, which includes a lot of inflections to convey grammatical relationships, makes it even more difficult to develop effective lemmatization algorithms.

Considering the morphological similarities between Azerbaijani and Turkish languages, the Turkish stemming research are of great value, especially the earlier research efforts. The studies of Dinçer and Karaoğlan (2003) have proven the effectiveness of probabilistic models in stemming, and this may indicate that the Azerbaijani language could also be improved by using probabilistic models to manage its complex morphological structure effectively [1]. Furthermore, Stemming Accuracy in Turkish, written by Sever and Bitirim (2003) puts forward the significance of the context in deriving the correct stems. What is more, Azerbaijani lemmatization algorithms should be able to adapt to different linguistic contexts and accurately process various textual environments in order to increase their effectiveness [4].

Hybrid approaches, which are a combination of both the rule-based and statistical methods, have been demonstrated to attain a higher accuracy and flexibility, which is very important for languages with the rich morphological structures like Azerbaijani. For one case, the research conducted by Köşülü and Karaoğlan (2016) shows that it's possible to get nicely balanced approach between statistical-based approach and rule-based approach for our language [3]. Considering the combination of tagging and stemming as an unsupervised method of the research conducted by Bölücü (2017), it can be a promising approach towards the Azerbaijani lemmatization, and more advantageous when there is a limitation of available annotated resources [2].

Finally, this literature review highlights the importance of adapting the lemmatization techniques accordingly to Azerbaijani language, with the consideration of its unique morphological complexity and the experience of similar agglutinative languages. Through integrating these results, our aim is to target improving NLP applications in Azerbaijani language sphere, and hence facilitating better and more efficient Azerbaijani language processing in different digital environments.

3 Research Methodology

3.1 Workspace Setup

For the training of the model Nvidia RTX A5000 with the 24GB GPU and 128GB RAM machine was used. The machine has an Intel Xeon W-2295 CPU, 4.1TB HDD and the operating system is Ubuntu 20.04 LTS.

3.2 Data Collection and Labeling

In this study total of 505,492 sentences as training data and 9,423 sentences as testing data were used. Both training and testing datasets was taken from the preprocessed and normalized corpus that was previously used in the research project for Azerbaijani speech recognition. The purpose of using this already preprocessed and normalized dataset in this project is ensuring the consistency of the used data and saving time in the project process.
The actual source of the previously used dataset is publicly available resources such as Azerbaijani news websites Wikipedia and books. The variety in sources of data ensures a wide linguistic representation of the built lemmatization model.
The 9,423 sentences for the test data were manually labeled by one linguist and three last year computer science student as a part of their senior design project. This senior design project also involved developing a hybrid lemmatizer for the Azerbaijani language with a 95% word-level accuracy. During the labeling process, the students continuously interacted with the linguist to ask linguistic questions to ensure labeling accuracy.
Due to the substantial data requirements of deep learning models, a large amount of labeled data was necessary for effective training. To meet this demand, the hybrid lemmatizer developed in a senior design project, which achieves 95% word-level accuracy, was utilized to label the 505,492 training sentences. It is acknowledged that this lemmatizer, while highly accurate, still contains a 5% error rate. Given that the training data was labeled using this hybrid lemmatizer, there is an awareness that inaccuracies are present in the dataset.
Since there is a 5% error rate in the training data the potential impact of that inaccuracy is elaborated in the "Error Analysis" section of the thesis.

3.3 Data Pre-Processing

During the data preprocessing step of both training and testing datasets, all characters that are not digits or letters were removed from the beginning and ending of each single word. This process is important because the unnecessary punctuations are removed which could affect the model's learning process.
Moreover, all the sentences in training and testing data are lowercased which makes the model case insensitive. This feature makes the approach robust to the potential case errors of the input data that is given by the user. The model treats input as lowercased so it does not depend on the correctly given input in terms of cases of each character.
The steps that are taken into account as the preprocessing step are crucial since they eliminate the unnecessary potential variability of input data.

3.4 n-gram Approach in Training Process

The model was trained by first mapping full sentences from their original form to their stemmed versions. While the model achieved 96% accuracy at the character level in the testing process, its performance was only up to 50% at the word level. One of the identified problems was that the model created irrelevant or unrelated sentences whenever text that was not in the training data was shown to it. This kind of behavior may well suggest that the model must have gone through many examples to not commit such errors and, therefore, may be related to underfitting.

Besides, the original training strategy also involved the truncation of longer sentences, thus also leaving the performance on inputs of greater lengths somewhat unclear as compared to that obtained during training. This now poses a risk for the model output to be unpredictable on extended sentences and hence unsuitable for real-world application where the length of sentences is of significant variety.

These vulnerabilities were mitigated and the model made more robust by moving from full sentence training to a 2-gram based approach. In the aforementioned training approach, each sentence is tokenized into 2-grams and, similarly, the pre-processed forms of the sentences are also tokenized into 2-grams. Then, the model with a sequence-to-sequence architecture with attention was trained to map these 2-grams to their corresponding stemmed 2-grams. This brings the model into a position where it will be able to properly handle inputs of arbitrary lengths, since in each case the handling of each 2-gram is independent of its position in the sentence.

This is done during training: sentences are taken, broken into 2-grams, mapped serially into their lemmatized version, and then reconstructed back to sentences. In application, input sentences are broken into 2-grams and fed to the model. The model outputs lemmatized 2-grams, which are then reconstructed into sentences. This is especially useful for Azerbaijani because the lemma of a word is dependent on the following word. It allows for the overlap in 2-grams to provide contextual cues necessary for accurate lemmatization, ensuring that the first occurrence of each word in overlapping 2-grams determines its correct lemma.

After transitioning to the 2-gram approach, the training dataset expanded to include 4,383,395 2-gram pairs, while the test dataset comprised 74,796 2-gram pairs. This significant increase in training instances, from 505,492 original sentences to millions of 2-gram pairs, substantially augmented the model's exposure to varied linguistic patterns. Such an expansion has been instrumental in enhancing the model's ability to recognize and accurately reproduce the character-level patterns of Azerbaijani words. The model was trained with 3 epochs and 40 batch sizes.

In the 2-gram approach, it is assumed that to find the lemma of the word the next word in the 2-gram determines the lemma of the actual word. As shown in Figure 1 the first words in 2-grams are taken as the result of the lemmatization output.

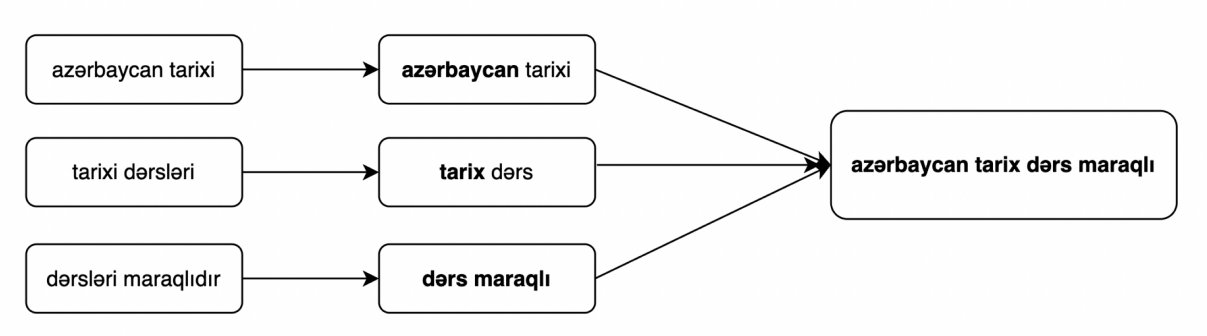

Figure 1: The next word determines the word's lemma in 2-gram

In order to add more context to find the lemma of the word 3-gram approach was also tested. In the experiments, it was observed that the 3-gram approach showed **96.4085** word level accuracy. Theoretically, the 3-gram's performance should surpass the result of the 2-gram. However, due to the fact that there are not random mistakes in the training data the model is actually forced to learn not correct rules for the lemmatization. Thus the 3-gram shows less accuarvy in the test data. In the 3-gram model, the middle word is taken as the result of the lemmatization. In that approach, it is assumed that the middle word's lemma depends on the previous and the next word. Figure 2 shows that approach visually.

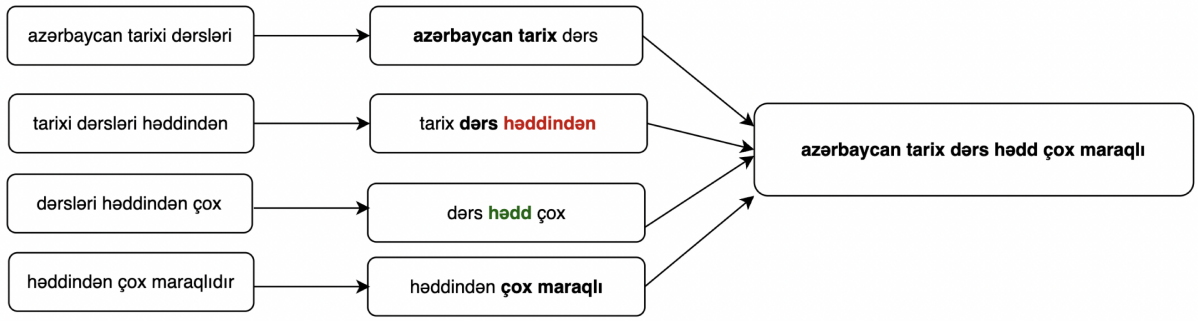

Figure 2: 3-gram approach middle word's lemma determined by previous and next word

In the training process of 3-gram, the train data consisted of 3,878,345 3-gram pairs, and the test data was 73,105 3-gram pairs. The model was trained with 6 epochs and 40 batch sizes.

3.5 Proposed Model and Architecture

The research introduces a character-based sequence-to-sequence (Seq2Seq) model enhanced with an attention mechanism, tailored to address the intricate morphological structure of the Azerbaijani language. The Figure 3 depicts the visual architecture of the proposed model.

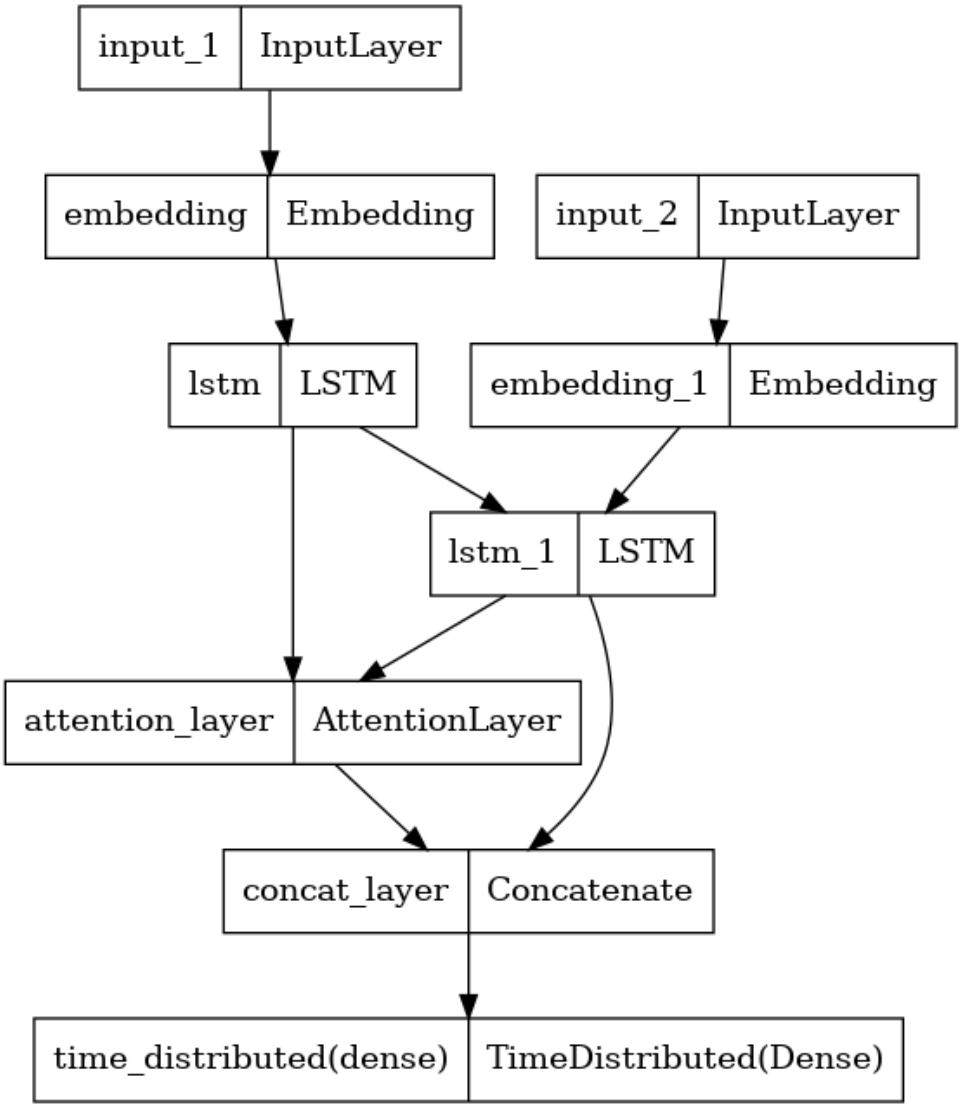

Figure 3: Visual architecture of the proposed model

3.5.1 Data Preparation and Input Handling

Given the rich morphological nature of Azerbaijani, where new words can be formed by adding prefixes, suffixes, or infixes, a character-level tokenization approach is adopted. The reason behind character-level tokenization is that unlike word tokenization, which could miss out on morphological derivations, character tokenization captures all possible variations in word formation. This process begins with tokenizing the sentences into character tokens, which convert text into sequences of character tokens. The character-level tokenization approach ensures that the model can generalize well across different forms of the same lemma, essential for robust lemmatization.

Each 2-gram pair is augmented with special characters “<” and “>” to denote the beginning and end, facilitating the model’s understanding of sequence start and stop points.

The proposed approach requires the same length to be fed into the model. Therefore, the maximum length for the 2-grams is set at 85 characters, which corresponds to the longest 2-gram observed in the dataset. To standardize the input lengths, ’post’ padding is applied, which adds zeros at the end of shorter sequences. This method ensures that no informational content from the beginning of the 2-grams is lost during padding.

3.5.2 Embedding Layer

Both the encoder and decoder incorporate embedding layers that transform the indices into dense vectors of fixed size (300 dimensions in this study). These embeddings capture more detailed information about each character than the indices themselves, providing a richer input for the subsequent recurrent layers.

3.5.3 Encoder-Decoder Architecture

The core of the model is the encoder-decoder architecture, commonly employed in machine translation.
The encoder module uses an LSTM (Long Short-Term Memory) layer to process the input character sequences. The embedding layer transforms these characters into a 300-dimensional vector space to capture semantic meanings more effectively. The LSTM processes these embeddings, capturing temporal dependencies among characters and encoding the entire sequence into a set of feature vectors (hidden states). These vectors provide a simplified representation of the input sequence, carrying crucial information for each timestep.
The decoder also employs LSTM layers but is initialized with the final hidden state of the encoder, providing a starting point for generating output. It uses its own embeddings to convert target sequences for training into a similar 300-dimensional space. As the decoder predicts characters sequentially, it relies on the encoder's output and its previous states to generate the next character in the sequence

3.5.4 Attention Mechanism

To enhance the decoder's ability to focus on relevant parts of the input sequence when generating each character in the output,Bahdanau attention mechanism is integrated. This mechanism computes context vectors by dynamically weighting the importance of each encoder output, thus helping the decoder to "attend" to the specific parts of the input when it is most relevant. This is particularly crucial for handling the variability and complexity in morphological structures, as it allows the model to focus more on morpheme boundaries and syntactic markers that are critical for accurate lemmatization and addresses the limitations of traditional Seq2Seq models by mitigating information loss, especially in longer sequences.

3.5.5 Output and Training

The concatenated output from the attention layer and the decoder LSTM feeds into a dense softmax layer, which predicts the probability distribution of the next character in the sequence. The model is trained end-to-end with the Adam optimizer, using sparse categorical crossentropy as the loss function to effectively handle the multi-class classification nature of the output space. The model is trained in 3 epochs with batches of 40 to optimize the learning process and manage computational resources efficiently. Figure 4 shows the proposed model's summary. There are 1,686,953 trainable parameters in the model.

```
Model: "model"
__________________________________________________________________________________________________
 Layer (type)                Output Shape                 Param #   Connected to
==================================================================================================
 input_6 (InputLayer)        [(None, 85)]                 0         []

 input_7 (InputLayer)        [(None, None)]               0         []

 embedding (Embedding)       (None, 85, 300)              16500     ['input_6[0][0]']

 embedding_1 (Embedding)     (None, None, 300)            15900     ['input_7[0][0]']

 lstm (LSTM)                 [(None, 85, 300),            721200    ['embedding[0][0]']
                              (None, 300),
                              (None, 300)]

 lstm_1 (LSTM)               [(None, None, 300),          721200    ['embedding_1[0][0]',
                              (None, 300),                           'lstm[0][1]',
                              (None, 300)]                           'lstm[0][2]']

 attention_layer (Attention  ((None, None, 300),          180300    ['lstm[0][0]',
 Layer)                       (None, None, 85))                      'lstm_1[0][0]']

 concat_layer (Concatenate)  (None, None, 600)            0         ['lstm_1[0][0]',
                                                                     'attention_layer[0][0]']
...
Total params: 1686953 (6.44 MB)
Trainable params: 1686953 (6.44 MB)
Non-trainable params: 0 (0.00 Byte)
```

Figure 4: The proposed model's summary

3.6 Model Evaluation

The model's performance was comprehensively evaluated through two distinct approaches: character-level and word-level accuracies. As a character-based sequence-to-sequence model, its primary training focus was on achieving high character-level accuracy, which is directly related to the model's ability to learn the underlying patterns of the Azerbaijani language.

Figure 5 illustrates the character-level accuracy observed during both training and testing phases. This visual representation helps in understanding the model's learning curve and its performance stability across different data sets.

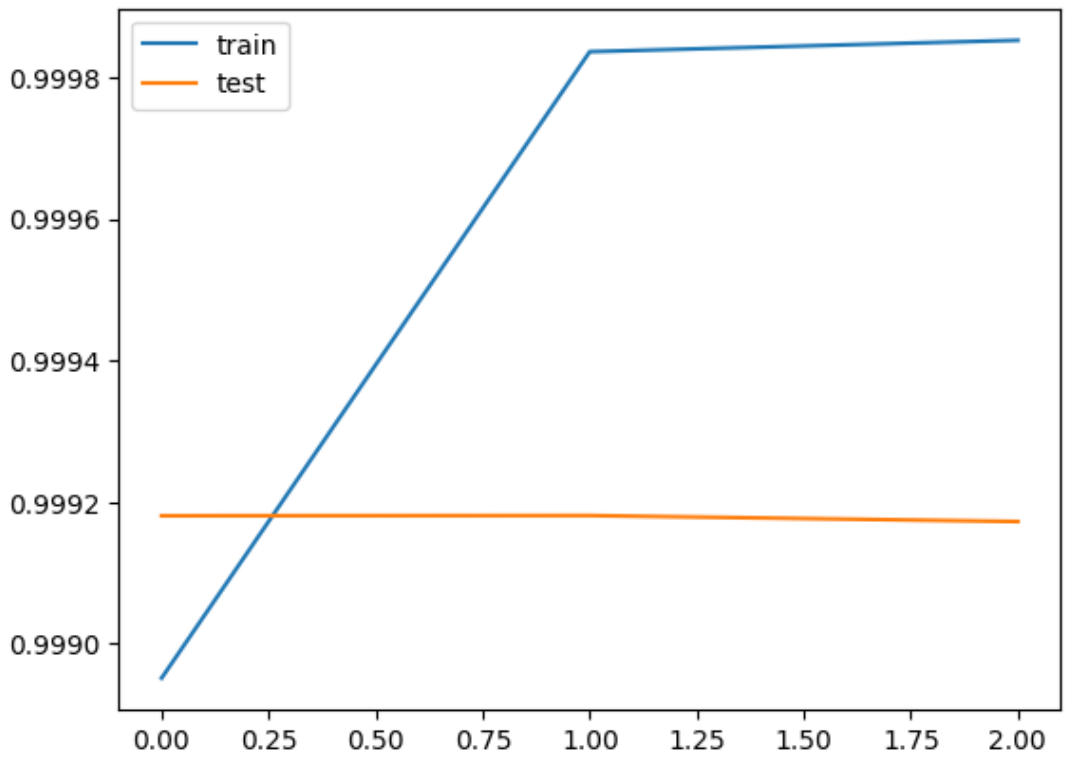

Figure 5: Train and Test accuracy over epochs

While character-level accuracy is crucial for a sequence-to-sequence model, word-level accuracy is particularly significant for tasks like stemming and lemmatization, where the correct interpretation of entire words directly impacts the usability of the processed text. The model was afterwards tested on a test set of 91,941 words, giving a word-level accuracy of 96%, as it predicted 88,407 words right. So, a high level of accuracy signifies that the model is practically sound for real-world applications.
Subsequently, the loss metrics of our model are also analyzed, as it is illustrated in Figure 6, for both the training and testing losses. This basically also provides an insight into how the model is behaving with respect to generalization beyond the training data.

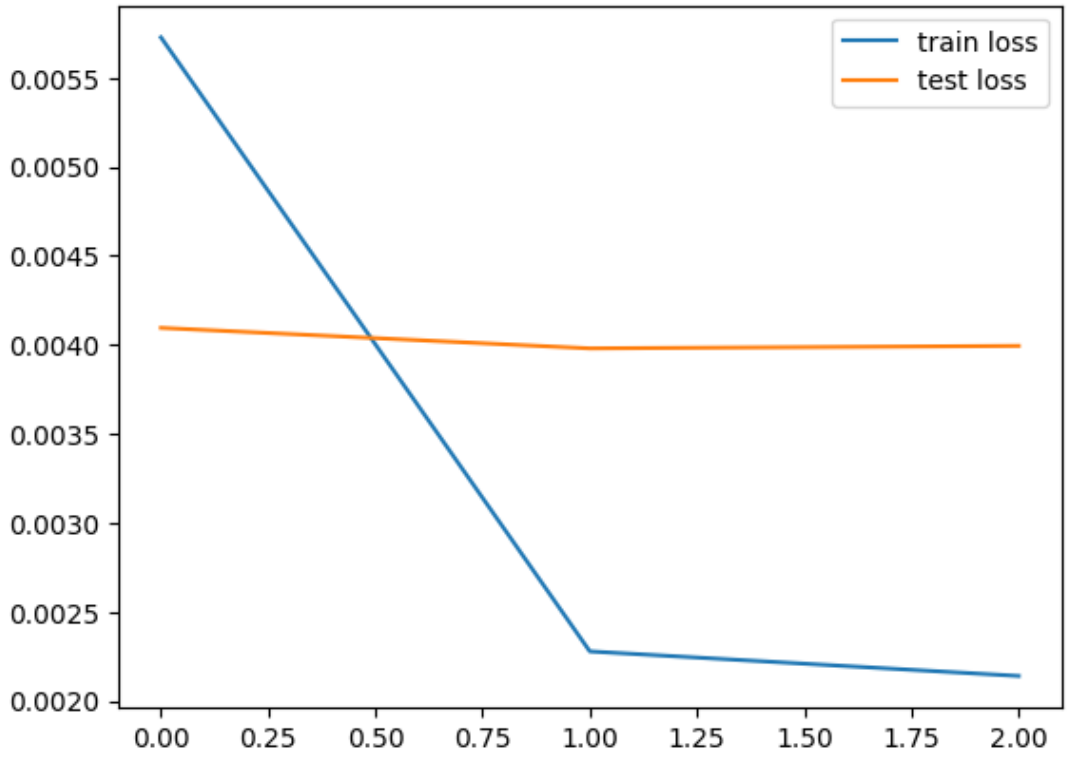

Figure 6: Training and Testing Loss across epochs

The combined test results confirmed that the model performs well in high-level lemmatization and stemming tasks, respectively; therefore, it could be applied to process texts in Azerbaijani in different computational linguistics applications.

3.7 Advantages of the Proposed Approach

The proposed fully machine learning approach in the study shows several advantages over the traditional rule-based and hybrid approaches.

One of the main advantage of the proposed approach is it eliminates the need of writing manual rules in order to solve the lemmatization problem. Traditional rule-based or hybrid methods require to develop set of detailed rules. Since there are plenty of exceptions in the linguistic rules this process is really challenging and prone to errors in exception cases. In contrast, the machine learning model automatically learns the rules from labeled data, requiring no explicit linguistic intervention, which simplifies the development process and potentially increases the adaptability and accuracy of the lemmatization process.

Another key benefit is the machine learning model's ability to handle exceptions in linguistic rules. Typically, rule-based systems struggle with exceptions since each exception must be individually coded into the system. With machine learning, however, increasing the representation of exceptional cases in the training data allows the model to learn these anomalies naturally. This adaptability enhances the model's effectiveness, as it can generalize from the given examples to handle similar exceptions in unseen data.

Additionally, the proposed approach reduces dependency on the accuracy of part-of-speech (POS) tagging models, which are crucial in current hybrid lemmatization system. Hybrid lemmatizers depend heavily on the correct identification of POS tags, and any errors in POS tagging can significantly affect their performance. Our approach only requires

improvements in the labeled data used for training the lemmatizer, bypassing the complexities associated with refining POS tagging models alongside the lemmatization rules. This singular focus on refining one model rather than multiple components simultaneously simplifies ongoing maintenance and enhancements.
Moreover, by training the model on lowercased sentences, we avoid the complications arising from case sensitivity, which is a notable issue in many hybrid lemmatization systems. The case insensitivity of our model ensures robustness, as it is less likely to be misled by errors or variations in capitalization that might otherwise affect the performance of case-sensitive systems.
Lastly, the model's reliance on 2-grams for training circumvents the need for precise sentence tokenization, a requirement in systems that utilize POS tagging. This design choice allows the model to process continuous text streams without needing explicit segmentation into sentences, thereby simplifying the preprocessing steps and reducing the potential for errors associated with sentence boundary detection.
In summary, the machine learning approach not only streamlines the development and maintenance of lemmatization tools for agglutinative languages but also enhances the robustness and accuracy of these tools in operational settings.

3.8 Limitations of the Approach

One of the primary limitations of the current model stems from the inherent error rate in the training data. Specifically, the model was trained on 505,492 sentences with a known error rate of 5%, arising from the inaccuracies of the hybrid lemmatizer used for labeling. These errors are not randomly distributed but are systematic in certain linguistic patterns, causing the model to learn these inaccuracies as valid patterns. Consequently, the model might replicate these errors during predictions, even though such mistakes may not exist in the test data. Since the model is character-based, these errors typically appear as one or two character differences, which minimally impact the overall accuracy but could still degrade the performance in nuanced linguistic contexts. Addressing this limitation could involve manually correcting these systematic errors in the training data or enhancing the accuracy of the hybrid lemmatizer to reduce the frequency of these errors.
Another significant limitation concerns the model's execution time. Current analysis indicates that the proposed attention-based sequence-to-sequence model requires approximately ten seconds to process a sentence containing ten words, which is impractical for real-time applications. In contrast, the current state-of-the-art hybrid model processes the same sentence length in about one second, making it more viable for practical use. It is often the case that less accurate models perform better in terms of execution speed. The focus of this study was to demonstrate that a fully machine-learning-based approach could feasibly replace rule-based or hybrid methods in terms of accuracy, not execution speed. Consequently, due to time constraints inherent in the scope of a master's thesis, improvements to reduce the execution time of the model were not explored. Future work could therefore focus on optimizing the model's architecture to enhance processing speed without compromising the accuracy gains achieved.

3.9 Assumptions of the Model

This model operates under several key assumptions that influence its training and expected performance. One primary assumption is that the training data, sourced from open-access Azerbaijani content such as news websites and books, is predominantly composed of correctly spelled words. Consequently, the model has been trained to recognize and learn from these accurately spelled sequences. Therefore, it is assumed that the inputs to the model during its operational phase are similarly spell-checked to ensure consistency with the training conditions. Any deviation in spelling accuracy between the training data and real-world application could impair the model's effectiveness.
Furthermore, the model presupposes that certain preprocessing steps are consistently applied to both training and test data. This includes converting all input text to lowercase and removing punctuation at the beginning and end of words. These preprocessing steps are critical as they standardize the input data, reducing variability and focusing the model's learning on meaningful textual content rather than formatting differences. These steps will be automated within a library that accompanies the model, ensuring that data fed into the model during practical applications is processed uniformly, adhering to the same standards established during training.

3.10 Experiments

To demonstrate the importance of lemmatization for Azerbaijani, three distinct experiments were conducted:

- The first experiment focused on sentiment analysis of Azerbaijani news articles. It compared the performance of TF-IDF vectorization with and without lemmatization. The results indicated that incorporating lemmatization improved accuracy by 8.9%, showcasing its beneficial impact on processing Azerbaijani text for sentiment analysis.

- In the second experiment, the word embedding for the Azerbaijani language was trained using the Word2Vec algorithm on both lemmatized and non-lemmatized sentences. The 2,261,237 sentences were lemmatized and both versions were used in the experiment. In order to evaluate which embedding model performs better both extrinsic and intrinsic evaluation was conducted. In extrinsic both lemmatized and non-lemmatized version of embedding are tested on sentiment analysis task for Azerbaijani and word embeddings are used as the vectorization stage of sentences. In intrinsic evaluation on the other hand both lemmatized and non-lemmatized word embedding tested on word analogy evaluation and defining the related distinct terms.

- In the third experiment the effect of the lemmatization in number of unique words in 2,261,237 sentences corpus was analysed. By lemmatizing the text and counting the unique words before and after lemmatization, the experiment quantified how lemmatization normalized the text. The results were expressed as a percentage reduction in unique word count, illustrating the extent to which lemmatization consolidates linguistic variations and simplifies text analysis.

4 Research Results and Analysis of Results

4.1 Experiment I

In this experiment the lemmitizer is used in the preprocessing step of the sentiment analysis of the news articles for Azerbaijani. The dataset was 12,200 news articles and the target distribution is 7600 (62.61%) negative and 4600 (37.39%) positive news articles. The effect of the lemmtizer was it improved the accuracy of the Logistic Regression's accuracy by 8.9% f1-score by 8.4% and Support Vector Machine's accuracy by 6.4% f1-score by 6.1%.

Model	Lemmatized	Feature Extractor	Accuracy (%)	F1 Score (%)
LR	**True**	TF-IDF, n-gram range (1,3)	**94.6**	**94.6**
LR	False	TF-IDF, n-gram range (1,3)	85.7	86.2
SVM	**True**	TF-IDF, n-gram range (1,3)	**93.1**	**93.1**
SVM	False	TF-IDF, n-gram range (1,3)	86.7	87.0

Table 1: Effect of the lemmatizer in Sentiment Analysis model performance

4.2 Experiment II

4.2.1 Word Embedding Analogy Test: Ankara is to Turkey as Tbilisi is to ?

The test employed is mathematically represented as:

$$\vec{v}(Tbilisi) + \vec{v}(Turkey) - \vec{v}(Ankara) = \vec{v}(?)$$

where $\vec{v}(word)$ denotes the vector representation of the word. The results, summarized in the table below, highlight the comparative performance of the lemmatized and non-lemmatized models on the analogy "Ankara is to Turkey as Tbilisi is to ?". This table underscores the importance of lemmatization in enhancing the model's capability to pinpoint correct geographical associations, as evidenced by the correct identification of "gürcüstan" in the lemmatized model.

Lemmatized Version Output	Score	Non-Lemmatized Version Output	Score
gürcüstan	0.5381	özbəkistan	0.5538
ukrayna	0.4865	moldova	0.5493
rustavi	0.4679	qazaxıstan	0.5407
azərbay	0.4620	**gürcüstan**	0.5338
kutais	0.4505	tacikistan	0.5097
moldova	0.4464	belarus	0.5072
sloveniya	0.4458	bolqarıstan	0.5070
batumi	0.4405	qırğızıstan	0.5040
rumıniya	0.4383	turkiyə	0.5029
belarus	0.4344	ukrayna	0.4996

Table 2: Word Embedding Analogy Test Result 1: Ankara is to Turkey as Tbilisi is to ?

4.2.2 Word Embedding Analogy Test: Daughter is to Mother-in-law as Son is to ?

The test analogy is mathematically represented as follows:

$$\vec{v}(Son) + \vec{v}(Mother - in - law) - \vec{v}(Daughter) = \vec{v}(?)$$

where $\vec{v}(word)$ denotes the vector representation of the word. The results of this analogy test, comparing lemmatized and non-lemmatized model outputs, are presented below.

Lemmatized Version Output	**Score**	**Non-Lemmatized Version Output**	**Score**
qayınata	0.5774	bədənim	0.6962
baldız	0.5274	namusu	0.6878
yeznə	0.5271	suzann	0.6878
iffət	0.5212	övladımız	0.6826
zəkəriyyə	0.5212	ilahələri	0.6820
kamrani	0.5210	vayaleti	0.6798
qubadov	0.5157	deməyəcək	0.6793
səlimə	0.5147	pıçıltı	0.6785
qayın	0.5137	isaakov	0.6782
gülgəz	0.5137	kəlamıdır	0.6781

Table 3: Word Embedding Analogy Test Result 2: Daughter is to Mother-in-law as Son is to ?

The results indicate significant differences in the model's performance based on text preprocessing techniques. The lemmatized model more accurately identifies family-related terms, while the non-lemmatized model produces outputs that are less contextually relevant.

4.2.3 Word Embedding Analogy Test Result 3: King is to Man as Queen is to ?

In this experiment in order to see which version of lemmatization can actually find the gender relationships the famous royal titles example was taken into account. The famous analogy of "king is to man as queen is to ?" is represented as a vector in that equation:

$$\vec{v}(queen) = \vec{v}(woman) + \vec{v}(king) - \vec{v}(man)$$

the $\vec{v}(word)$ is a vector representation of the word. Table 4 shows the results from the lemmatized and non-lemmatized models.

Lemmatized Version Output	Score	Non-Lemmatized Version Output	Score
kraliça	0.5085	prussiya	0.4647
monarx	0.4666	kralı	0.4568
hersoq	0.4544	imperator	0.4504
vəliəhd	0.4459	karlın	0.4414
krallıq	0.4451	kraliyyət	0.4308
taxt-tacının	0.4426	**kraliça**	0.4306
karlı	0.4400	kennedi	0.4305
imperator	0.4369	aleksandrın	0.4225
elizabet	0.4348	versal	0.4219
papa	0.4339	bismark	0.4215

Table 4: Word Embedding Analogy Test Result 3: King is to Man as Queen is to ?

The results showcase the nuanced understanding of gender roles and titles by the lemmatized model, which more accurately identifies 'kraliça' (queen) as the top result. In contrast, the non-lemmatized model presents a wider array of results, some of which are less relevant to the specific gender context of the query.

4.2.4 Lemmatization effect of related distinct terms

The experiment aims to demonstrate the effects of lemmatization on a word similarity test using the word "polis", comparing the top 10 similar words as generated by both lemmatized and non-lemmatized versions of the model. The test is designed to illustrate how non-lemmatized models may confuse different grammatical forms of the same word as distinct entries(*polisi*, *polislər*, *polisin*, *rpi-nin*), whereas lemmatized models provide a more varied set of related but distinct terms.

Lemmatized Output	Score	Non-Lemmatized Output	Score
dyp	0.6189	**polisi**	0.6518
milis	0.5963	**polislər**	0.6373
şpi	0.5851	milis	0.6154
jandarmeriya	0.5739	dyp	0.5860
ypx	0.5711	**polisin**	0.5802
hüquq-mühafizə	0.5705	jandarmeriya	0.5741
patrul	0.5646	**rpi-nin**	0.5737
pb	0.5622	polsi	0.5633
rpi	0.5612	jandarma	0.5552
şrpş	0.5610	jandarm	0.5551

Table 5: Comparison of Lemmatized vs. Non-Lemmatized Model Outputs for "Polis"

The results indicate that lemmatization leads to a broader, more accurate representation of related terms, avoiding the repetition of the root word with various suffixes, which is more common in non-lemmatized outputs. This enhances the semantic quality of the model, making it more useful for practical applications where diverse semantic relationships are important such as Part of Speech Tagging, Named Entity Recongnition, Sentiment Analysis and so on.

4.2.5 Comparison of Sentiment Analysis Results Using Lemmatized and Non-Lemmatized Data

In this study,the performance of sentiment analysis tasks is compared using Word2Vec embeddings trained on lemmatized versus non-lemmatized data. The results show that while both embeddings perform well, the lemmatized version achieves slightly better accuracy, indicating its potential benefits for tasks such as sentiment analysis, POS tagging, and named entity recognition where pre-trained word embeddings significantly influence performance.
The following table presents the detailed performance metrics for both lemmatized and non-lemmatized data:

Data Type	Precision	Recall	F1-Score	Support	Accuracy
Negative Class					
Lemmatized	0.95	**0.96**	0.95	1546	0.94
Non-Lemmatized	0.95	**0.95**	0.95	1546	
Positive Class					
Lemmatized	0.92	0.91	**0.92**	896	0.94
Non-Lemmatized	0.92	0.91	**0.91**	896	

Table 6: Comparison of Sentiment Analysis Results Using Lemmatized and Non-Lemmatized Data

4.3 Experiment III

In the third experiment in order to show the effect of the lemmitizer to the text how much does the lemmetizer decrease the unique word count is tested. It was observed that in 2,261,237 sentences there was 648,332 unique words when these sentences were lemmatized it was decreased to 251,322 which means the lemmatization decreases the word count by 61.24%. Figure 7 shows that statitics visually.

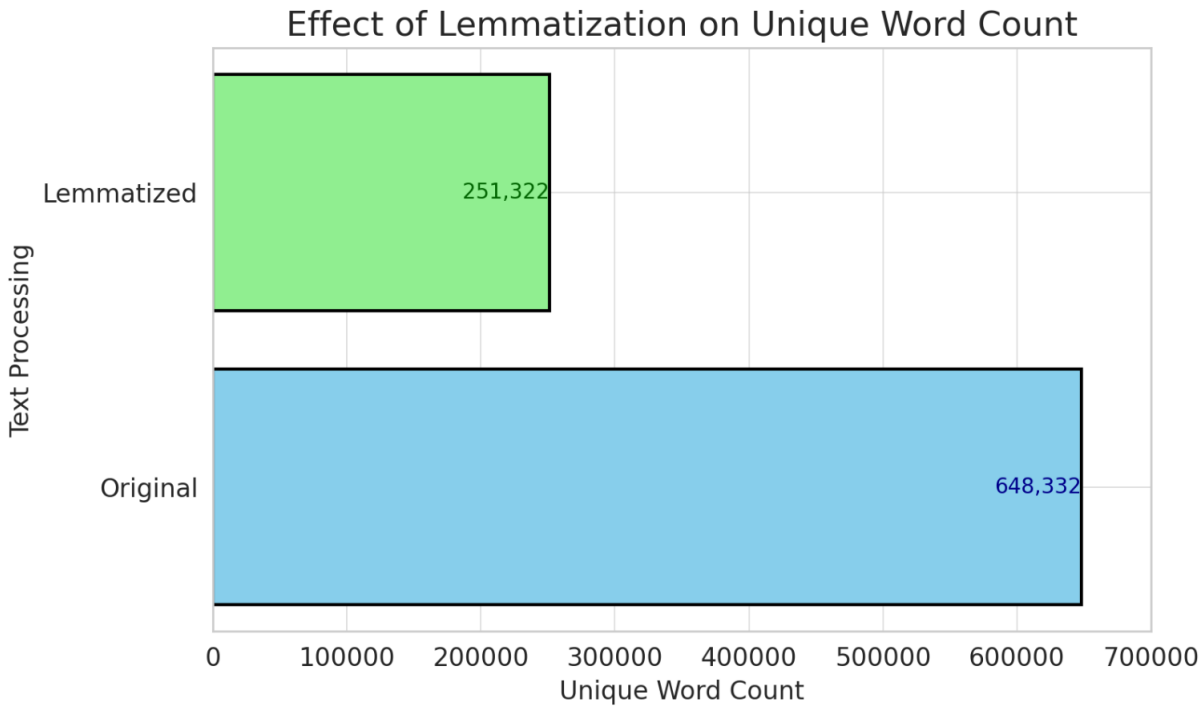

Figure 7: Effect of Lemmatization on Unique Word Count

4.4 Error Analysis

During the error analysis it was observed the the majority of the errors was due to the 5% error rate of the labeling process that the hybrid method was took into account.

5 Summary and Future Work

5.1 Summary

This study has demonstrated that traditional hybrid or rule-based lemmatization approaches, which require extensive manual rule-writing, can be effectively replaced by a fully machine-learning-based approach. This is particularly advantageous for agglutinative languages like Azerbaijani, where the complexity of the language makes rule-based systems cumbersome and often inefficient. The machine learning model developed in this research learns to perform lemmatization directly from the data, bypassing the need for manually crafting exhaustive linguistic rules. This approach not only simplifies the process but also enhances the adaptability and scalability of the lemmatization tool.

5.2 Future Work

The study successfully showed that the traditional rule-based lemmatization process can be replaced by a fully machine-learning-based approach. Moreover, it showed that this problem is still significant for agglutinative languages such as Azerbaijani. However, there are still some areas that can be improved in future research:

1. **Optimization of Execution Time:** Since the optimization of the execution time was not the primary goal of the study current execution time of the model is not practical for real applications. The study focused and showed that the fully-machine-learning approach surpasses the hybrid approach. As further work, the execution time can be optimized by refining the model's architecture or taking into account the advanced computational techniques that optimize the execution time by using the hardware solutions.

2. **Correction of Labeling Errors in Training Data:** The training data was labeled using the 95% word level accurate hybrid lemmatizer. Due to the 5% error rate in the training data, there are plenty of non-random errors in the training data that force the model to learn the not correct lemma version of the word. Future work should include either correcting this, not random mistakes in the training data, or improving the hybrid model's accuracy so that it will label training data more accurately.

3. **Exploring Alternative Architectures:** To address the limitations associated with the n-gram-based approach, which might miss some contextual information, further research could explore alternative model architectures. These alternatives could include deeper neural networks that capture longer dependencies without relying on n-grams, thus preserving more contextual information and potentially improving the model's performance.

By addressing these aspects, future research can build on the foundations laid by this study, pushing the boundaries of what machine-learning-based lemmatization tools can achieve, particularly for complex languages like Azerbaijani.

6 Development of nlp_az library

The research of Natural Language Processing in the Azerbaijani language has been continuing for several years. There are plenty of paper's published which are useful in the development of NLP applications for the Azerbaijani language.

The main problem is although lot's of successful projects developed for Azerbaijani NLP even simple preprocessing functions such as word tokenization is not publicly available to use.

To both give the lemmatizer publicly available and solve the problem of not using the already done projects throughout the thesis the first nlp library of Azerbaijan called 'nlp_az' was developed.

6.1 Lemmitezer for Azerbaijani language: lemmatizer_aze

There will be some functionalities in this library as follows. First since this library was developed throughout this thesis the lemmatization is integrated to that project. Both hybrid and fully machine learning-based lemmitizers is available in the 'nlp_az' library. In order to use fully machine learning-based lemmitizers the 'ml' parameter should be set to 'True'. By default, the 'lemmatize_az' function predicts by using a hybrid approach.

```
from nlp_az import lemmatize_aze
lemmatize_aze("ona dedimki alma alma", model, w2i, idx2tag)
```

```
'o demək alma almaq'
```

Figure 8: Hybrid Lemmatizer

7 Bibliography

[1] Bahar Karaoğlan Bekir Taner Dincer. Stemming in agglutinative languages: A probabilistic stemmer for turkish. In *Computer and Information Sciences - ISCIS 2003, 18th International Symposium*, pages 244–251. Springer, 2003.

[2] Necva Bölücü and Burcu Can. Joint pos tagging and stemming for agglutinative languages. *CICLING 2017*, 2017.

[3] Tarik Kisla and Bahar Karaoğlan. A hybrid statistical approach to stemming in turkish: An agglutinative language. *Anadolu University Journal of Science and Technology-A Applied Sciences and Engineering*, 17(2), 2016.

[4] Hayri Sever and Yıltan Bitirim. Findstem: Analysis and evaluation of a turkish stemming algorithm. In M.A. Nascimento, E.S. de Moura, and A.L. Oliveira, editors, *String Processing and Information Retrieval. SPIRE 2003*, volume 2857 of *Lecture Notes in Computer Science*, page Include page range if available, Berlin, Heidelberg, 2003. Springer.

8 Acknowledgments

The research presented in this study was conducted at the Center for Data Analytics Research (CeDAR) at ADA University. The author extends heartfelt gratitude to Nəzakət Qaziyeva for her invaluable linguistic assistance and to Dr. Samir Rustamov for his guidance and supervision throughout this project.