applied sciences

MDPI

Article

Automatic Translation between Mixtec to Spanish Languages Using Neural Networks

Hermilo Santiago-Benito [1,*], Diana-Margarita Córdova-Esparza [1,*], Noé-Alejandro Castro-Sánchez [2,*], Teresa García-Ramirez [1], Julio-Alejandro Romero-González [1] and Juan Terven [3]

[1] Facultad de Informática, Universidad Autónoma de Querétaro, Av. de las Ciencias S/N, Queretaro 76230, Mexico; teregar@uaq.mx (T.G.-R.); julio.romero@uaq.mx (J.-A.R.-G.)
[2] Centro Nacional de Investigación y Desarrollo Tecnológico, Tecnológico Nacional de México, Interior Internado Palmira S/N, Palmira, Morelos 62493, Mexico
[3] Instituto Politécnico Nacional, CICATA—Unidad Querétaro, Cerro Blanco 141, Col. Colinas del Cimatario, Queretaro 76090, Mexico; jrtervens@ipn.mx
* Correspondence: hsantiago13@alumnos.uaq.mx (H.S.-B.); diana.cordova@uaq.mx (D.-M.C.-E.); noe.cs@cenidet.tecnm.mx (N.-A.C.-S.)

Abstract: This paper introduces a novel method for collecting and translating texts from the Mixtec to the Spanish language. The method comprises four primary steps. First, we collected a Mixtec–Spanish corpus that includes 4568 sentences from educational and religious domain texts. To enhance the parallel corpus, we generate synthetic data with GPT-3.5. Second, we cleaned the data with a semi-automatic approach followed by preprocessing and tokenization. In preprocessing, we removed stop words, duplicated sentences, special characters, and numbers and converted them to lowercase. Third, we performed semi-automatic alignment to find the correspondence of Mixtec–Spanish sentences to generate sentence-level aligned texts necessary for translation. Finally, we trained automatic translation models based on recurrent neural networks, bidirectional recurrent neural networks, and Transformers. Our system achieved a BLEU score of 95.66 for Mixtec-to-Spanish translation and 99.87 for Spanish-to-Mixtec translation. We also obtained a translation edit rate (TER) of 0.5 for Spanish-to-Mixtec and a TER of 16.5 for Mixtec-to-Spanish. Our research stands out as a pioneering effort in the field of automatic Mixtec-to-Spanish translation in Mexico, filling a gap identified in the current literature.

Keywords: low-resource digital languages; automatic translation; transformers

Citation: Santiago-Benito, H.; Córdova-Esparza, D.-M.; Castro-Sánchez, N.-A.; García-Ramírez, T.; Romero-González, J.-A.; Terven, J. Automatic Translation from Mixtec to Spanish Languages Using Neural Networks. *Appl. Sci.* **2024**, *14*, 2958. https://doi.org/10.3390/app14072958

Academic Editor: Chilukuri K. Mohan

Received: 4 March 2024
Revised: 29 March 2024
Accepted: 29 March 2024
Published: 31 March 2024

1. Introduction

Automatic translation involves finding the equivalence of a text from a source language to a target language. The text generated after these tasks is called the target text. It is a valuable resource for supporting human translators [1], linguistic studies [2], automatic dictionary construction [3], automatic language analyzers [4], information retrieval systems [5], and resources for multilingual translation support in other languages [6]. The purpose of generating parallel corpora, automatic tokenization, and the development of the Mixtec–Spanish translation methodology is to contribute to the preservation of the Mixtec language, an indigenous language spoken by the Mixtec people in southern Mexico. The main challenge in this work is finding translations of texts written in Mixtec into Spanish, coupled with the difficulties arising from corpus scarcity, as the number of collected texts is limited. These texts are available physically or are not accessible online in digital format. The complexity of Mixtec grammar, such as manual construction and implementation of rule-based techniques, can take much work. Challenges include orthographic normalization and dealing with dialectal variants in Mixtec, in addition to the need for lemmatization tools, grammatical taggers, and tokenizers.

According to the catalog of Mexico's National Institute of Indigenous Languages (INALI) [7], there are 68 linguistic groups and 11 linguistic families. Linguistic groups form a set of linguistic variants related to an indigenous community. A linguistic family comprises languages that share a similar structure and lexicon, as well as a common historical origin. Mixtec belongs to the Oto-Manguean family, sharing characteristics with languages such as Triqui, Mixe-Zoque, Seri, Pame, Tlapaneco, Zapotec, and Mazahua, among others.

As per the census by the National Institute of Statistics and Geography (INEGI) [8], in Mexico, 7,364,645 people aged three years or older speak an indigenous language, which is equivalent to 6% of the population. The languages with the highest number of speakers are Nahuatl, Maya, Tseltal, Tsotsil, and Mixtec. Mixtec has a total of 526,593 speakers and is the fifth most spoken language in Mexico. In accordance with the National Institute of Indigenous Peoples (INPI) [9], Mixtec communities are located in the states of Oaxaca, Puebla, and Guerrero.

As per the catalog of (INALI) [7], the Mixtec language has 81 variants in Mexico. There could be even more variants, as each town may have its own Mixtec variant.

In this work, we address the translation of texts from the Mixtec variants of the states of Guerrero and Oaxaca since the majority of the texts in the collected corpus are from these variants.

The proposed methodology is applicable to any pair of languages with low digital resources. However, in this project, we focused on Mixtec–Spanish.

A survey on the topic points out that there are no scientific publications regarding Mixtec–Spanish translation, causing a need for more methodology for Mixtec–Spanish translation. The objectives of this study are as follows:

- Propose a methodology for Mixtec–Spanish language translation.
    1. Corpus compilation. We performed the search for parallel corpora, manually corrected, and generated synthetic data.
    2. Corpus preprocessing. This step involved cleaning special characters, numbers, duplicated sentences, and empty sentences.
    3. Semi-automatic alignment. The texts are aligned semi-automatically to generate paired parallel corpora at the sentence level.
    4. Translation training. We trained recurrent neural networks, bidirectional recurrent neural networks (BRNNs), and Transformers.
- Generate translation memories.
- Create a tokenized corpus at the word level.
- Generate synthetic Mixtec data with GPT-3.5.
- Evaluate the translation using the BLEU and TER metrics.
- Identify challenges in Mixtec–Spanish translation.

This work focuses on the translation of educational and religious texts, as these constitute the majority of our current corpus domain. Our proposal represents the first methodology for Mixtec–Spanish translation in Mexico.

This paper is structured as follows. Section 1 presents the introduction. Section 2 outlines the theoretical foundations of neural networks. Section 3 describes relevant works regarding translation. Section 4 describes our methodology for Mixtec–Spanish translation. Section 5 presents the results, followed by a discussion and conclusion in Sections 6 and 7.

2. Theoretical Foundation

Before describing the proposed methodology, it is essential to address the fundamentals of recurrent neural networks, Transformer networks, the BLEU evaluation metric, and Translation Edit Rate (TER).

2.1. Recurrent Neural Network

Recurrent Neural Networks (RNNs) are networks that generate sequences, such as texts and videos. RNNs maintain the context of previous sequences and the subsequent sequence, making them an ideal technique for text analysis. Additionally, RNNs have long short-term memory (LSTM) capabilities, which aid in handling sequences.

There are three types of recurrent neural networks: One-to-Many, Many-to-One, and Many-to-Many. The Many-to-Many type allows for handling sequences of inputs and outputs, enabling tasks such as automatic translation, text-to-speech conversion, or speech-to-text.

RNNs are suitable for generating short sequences with a limited number of characters, which is known as short-term memory. This poses a challenge when dealing with long sequences, as RNNs may not effectively manage large amounts of data. To address this issue, LSTM [10] was developed to overcome the limitations of recurrent networks. Figure 1 explains the architecture of an LSTM neural network.

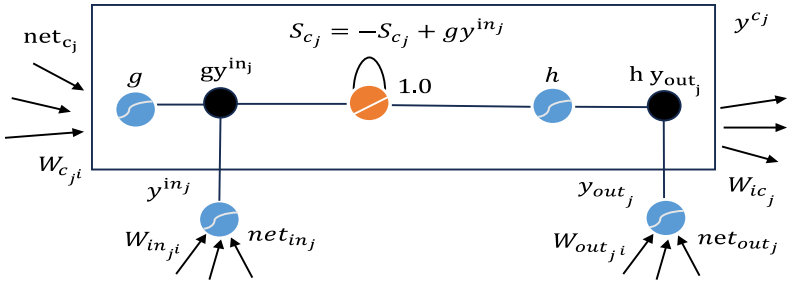

Figure 1. Schematic representation of an LSTM cell based on [10]. The diagram illustrates the data flow through an LSTM unit for sequence learning tasks. Input gates (g) control the flow of input activations into the memory cell. Output gates (h) manage the output flow of cell activations into the rest of the network. W terms represent the weight matrices associated with each gate and cell state (S_{cj}) for current (c) and previous time steps (in). Net input and output (net_{inj}) and (net_{outj}) activations are also depicted, along with the cell output (y_{cj}). This structure enables the LSTM to capture long-range dependencies in data. It is widely used in complex sequence modeling tasks such as natural language processing, speech recognition, and time-series prediction.

The architecture allows a constant error flow through the units. It includes a multiplicative input unit to preserve the memory data stored in j from alterations. Another multiplicative output unit protects the irrelevant data stored in j.

2.2. Transformer Network

Vaswani et al. [11] proposed the transformer network, a new approach that includes attention mechanisms and positional encoding. The transformer network is based on an encoder–decoder architecture with attention mechanisms. Figure 2 illustrates the general architecture of a Transformer. The components of a Transformer network are detailed below.

- Encoder: The encoder consists of a multi-head attention mechanism followed by a feed-forward neural network. Each sublayer within the encoder has a residual connection around it, followed by layer normalization. The term "flows" refers to the data flowing through these residual connections.
- Decoder: The decoder's architecture mirrors that of the encoder, with the addition of masking to the input. This masking prevents the decoder from attending to subsequent positions, ensuring that predictions for a given position can only depend on known outputs at previous positions.
- Attention Mechanism: The attention mechanism is built around a function that computes outputs based on four types of vectors: query, key, value, and output. The mechanism operates by calculating a weighted sum of the values, where the weight assigned to each value is determined by a compatibility function of the query with the corresponding key. This process involves a scaled dot-product attention function,

which scales the dot products of the queries and keys by the inverse square root of the dimension of the keys to facilitate training stability. The weights are then applied using a softmax function to ensure they sum to one.

- Multi-Head Attention: This component enhances the model's ability to focus on different positions. It does so by projecting the queries, keys, and values multiple times with different, learned linear projections. Attention is then applied in parallel to these projections, producing multiple output vectors that are concatenated and once again projected. This process allows the model to attend to information from different representation subspaces at different positions simultaneously.
- Positional Encoding: To account for the sequence order in the absence of recurrent or convolutional layers, positional encodings are added to the input embeddings at the bottoms of the encoder and decoder stacks. These encodings use sine and cosine functions of different frequencies to inject information about the position of each token in the sequence. This allows the model to utilize the order of the sequence, which is crucial for understanding the structured data in many tasks.

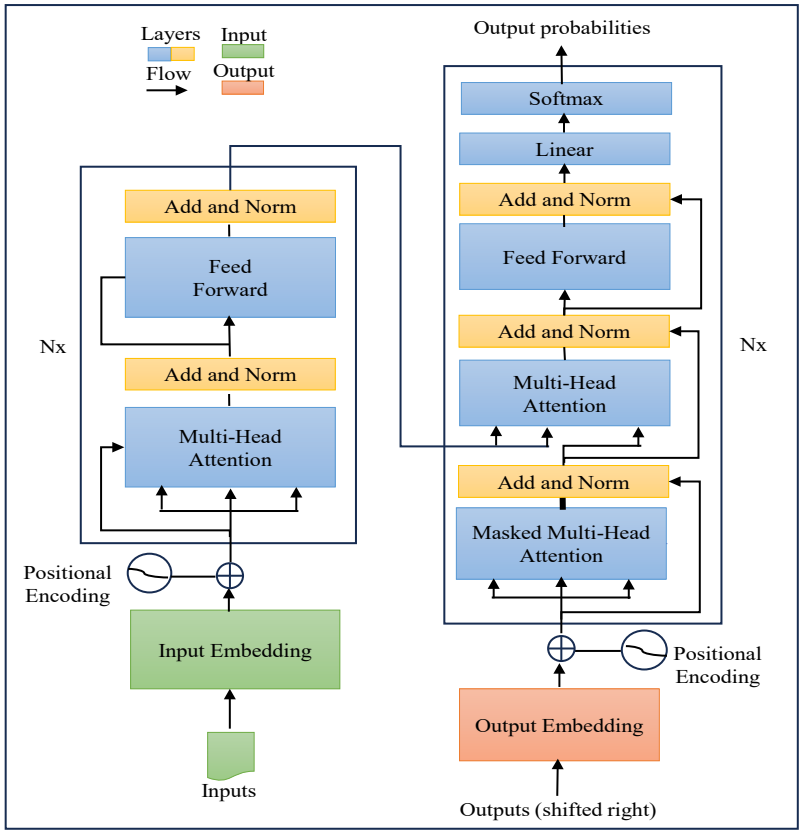

Figure 2. Architecture of the Transformer model's encoder–decoder framework based on [11]. The encoder (**left**) transforms inputs into embeddings, augmented with positional encodings, and processes them through N_x layers of Multi-Head Attention and feed-forward networks. The decoder (**right**) uses these encodings, applying Masked Multi-Head Attention to prevent future position information leakage and generating output through subsequent N_x layers. The process culminates in a linear and softmax layer that produces the sequence's output probabilities. This design allows for efficient parallel processing and is fundamental for tasks like machine translation.

2.3. Bilingual Evaluation Understudy (BLEU)

As stated by Papineni et al. [12], BLEU is an automatic metric for evaluating the quality of translation in comparison to a reference. The metric is language-independent and fast.

The reference is the correct translation proposed by a human. BLEU scores range from 0 to 1, where a score of 0 indicates no match with the reference, and 1 indicates an exact match. In the evaluation, two approaches are proposed, as outlined in Equations (1) and (2):

- Modified Precision. The BLEU metric is implemented at the phrase level to obtain precision using n-grams of the candidate translation and the reference translation. However, precision has some drawbacks, hence the proposal of a modified precision P_n.
- Brevity Penalty. In the evaluation of the candidate translation compared to the reference translation, there can be variations in length. This impacts the precision obtained. The candidate translation may have more n-grams that are not present in the reference translation, resulting in lower precision. Conversely, the candidate translation may be shorter while the reference translation is longer. To address this, the brevity penalty BP is proposed in Equation (2).

In Equation (1), the geometric mean of the modified precision of n-grams p_n is obtained using n-grams of length N and the positive weights are incremented by one w_n. Here, $w_n = 1/N$.

$$BLEU = BP \cdot exp(\sum_{n=1}^{N} w_n \log P_n) \tag{1}$$

Equation (2) is shown below, where c is the length of the proposed translation, and r is the length of the reference translation.

$$BP = \begin{cases} 1 & \text{if c>r} \\ e^{(1-r/c)} & \text{if c} \leq \text{r} \end{cases} \tag{2}$$

2.4. Translation Edit Rate

Snover et al. [13] refers to TER as an automatic evaluation metric for machine translation that allows calculating the number of edits necessary to modify a hypothesis in such a way that it matches the reference. Only the number of editions to the closest reference is measured. Below is Equation (3), used to obtain the TER metric.

$$TER = \frac{\#\, of\ edits}{average\ \#\, of\ reference\ words} \tag{3}$$

Edits consist of the insertion, deletion, and substitution of individual words and changes in word sequences. The change moves an immediate sequence of words within the hypothesis to another location.

3. Related Work

As part of our research into the latest developments in the field, we came across some noteworthy studies on the automatic translation of digital languages that have a limited amount of resources available. We examined these studies and grouped them into three distinct categories based on the types of corpora they used for translation. These categories are parallel corpora, pseudo-parallel corpora, and corpus combination.

Parallel corpora aid in translation by providing equivalent translations between a source and target language. The advantages include error correction and contextual usage of words in both languages. Disadvantages include unavailability for low-digital-resource languages, noise in parallel texts, and linguistic variations causing translation difficulties.

Rule-based translation refers to the use of a set of grammatical rules to translate content from one language to another. These rules are created by language experts and are used to develop the translation system. One advantage of this type of translation is that it does not

require a corpus collection for the translation process. Additionally, users can add their own terminology to a dictionary to improve the quality of the system. However, there are some disadvantages to rule-based translation, such as the difficulty in defining grammatical rules, especially in languages with many linguistic variants. Another disadvantage is the lack of a bilingual dictionary of considerable size, which can affect the quality of translations generated by the system.

Translation based on pseudo-parallel corpora involves using a monolingual corpus in the opposite direction of the intended translation. The advantage of this type of translation is the availability of large quantities of data in either the source or target language, allowing for subsampling of the majority class samples. However, a disadvantage is that data are only available in a single language, meaning that synthetic data must be generated at either the source or destination in a translation scenario with parallel data.

Combining parallel, rule-based, and pseudo-parallel corpora presents advantages, since the use of a combination of these approaches strengthens the training data and improves the quality of the results. The use of parallel corpora in a pseudo-parallel combination helps to improve the accuracy of the results and generate synthetic data in both the source and destination directions.

Further details on each of these categories can be found below.

3.1. Parallel Corpora

Table 1 shows studies in the field of translation using parallel corpora. The table includes studies such as [14], which explores Wixarika–Spanish translation using statistical methods and evaluates the translations with WER (Word Error Rate) and TER (Translation Edit Rate) metrics. In this work, the authors also incorporated techniques like morphological segmentation and tagging-based segmentation. Notably, Zacarías-Márquez and Meza-Ruiz [15] applied Transformer networks to Ayuuk–Spanish translation, evaluated using the Bilingual Evaluation Understudy (BLEU) metric [12]. Further, Mager et al. [16] conducted experiments across multiple languages, including Ashaninka, Aymara, Bribri, Guarani, Náhuatl, Otomí, Quechua, Rarámuri, Shipibo-Konibo, Wixarika, and Spanish, using BLEU and ChrF (Character Level F-score) for assessment. Another work by Mager et al. [17] is notable for its experiments in Mexicanero, Purépecha, Yorem Nokki, Wixarika, Náhuatl, and Spanish languages using statistical and neural translation techniques, with BLEU as the evaluation metric. Knowless and Littell [18] proposed neural translation with translation memories for digitally low-resource American languages, such as German, Upper Sorbian, English, Inuktitut, Wixarika, Raramuri, Náhuatl, Guaraní, and Spanish, using ChrF and BLEU for evaluation. Additionally, Gezmu et al. [19] discussed translation involving preprocessing, segmentation, and alignment in an Amharic–English parallel corpus, evaluated using neural networks with BLEU, BEER [20], and characTER metrics [21]. Dione et al. [22] revealed approaches in neural translation with back-translation using French to Wolof and Wolof to English parallel corpora, employing BLEU for evaluation. Finally, Rubino et al. [23] presented neural translation in low-resource Asian languages, including English, Japanese, Lao, Malay, and Vietnamese, evaluated using the ChrF and BLEU metrics.

Table 1. Comparison of various translation studies utilizing parallel corpora within Statistical Machine Translation (SMT) and Transformer models. It details the references of the studies, the translation techniques employed, the specific tools used for the translation, and the metrics applied to evaluate translation accuracy, such as Word Error Rate (WER), BLEU score, and character error rate (ChrF).

References	Translation Techniques	Tools	Metrics
[14]	SMT	Giza++, Moses	WER 38, TER 0.86
[15]	Transformer	JoeyNMT	BLEU 5
[16]	Transformer	Moses	BLEU 5.67, ChrF 39.9
[17]	Transformer, SMT	Moses, OpenNMT	BLEU 23.47
[18]	Transformer	Sockeye	BLEU 38.2, ChrF 16.2
[19]	Transformer, SMT	Moses	BLEU 33.0, BEER 0.576, characTER 0.705
[22]	Transformer	Not specified	BLEU 37.5
[23]	Transformer, SMT	Fairseq, Moses	BLEU 21.9, ChrF 0.484

3.2. Pseudo-Parallel Corpora

Table 2 outlines studies on translation with low-resource languages using pseudo-parallel corpora. In their research, Kumar et al. [24] explored semi-supervised learning for transfer translation with a pseudo-corpus. They delved into languages such as Bhojpuri, Magahi, Hindi, and Magah, utilizing BLEU, ChrF, and TER for performance assessment. On the other hand, Imankulova et al. [25] demonstrated that the application of filtering techniques to a pseudo-parallel corpus can enhance neural machine translation. This study concentrated on a diverse linguistic range, including Russian, Japanese, Malagasy, German, and English, with the evaluation metrics being BLEU, AAS, and MAS.

Table 2. Evaluation of translation approaches using pseudo-parallel corpora.

References	Translation Techniques	Tools	Metrics
[24]	Transformer, SMT	Fairseq, Moses	BLEU 32.94, ChrF 0.59, TER 0.656
[25]	Transformer	OpenNMT	AAS 14.73, MAS 14.24, BLEU 14.70

3.3. Combination of Corpora

Table 3 presents studies related to machine translation using combined corpora. Hlaing et al. [26] showed translation leveraging POS tagging in Thai, Myanmar, and English languages. They utilized BLEU, ChrF, and WER as evaluation metrics. Hujon et al. [27] explored transfer translation using transformer networks with a shared, subword-tokenized vocabulary, focusing on English and Khasi languages. The evaluation employed BLEU, precision, recall, and F1 metrics. Meetei et al. [28] proposed a multimodal translation approach using neural networks. Their corpus included images with titles translated from English to Hindi, evaluated using BLEU and ChrF metrics. Meanwhile, Maimaiti et al. [29] emphasized the use of a shared lexicon and back-translation, introducing transfer translation with lexicon embedding pretraining for Azerbaijani and Uzbek. BLEU was the chosen evaluation metric. Sethi et al. [30] presented a hybrid translation method that integrated bilingual dictionaries, grammatical corpus, and syntactic rules between Sanskrit and Hindi. They evaluated their approach using the BLEU metric. Pirinen [31] discussed developing North Sámi–German translation utilizing morphosyntactic parsers and rules, with WER as the evaluation metric. Karakanta et al. [32] described a translation process through transliteration, transforming a high-resource language corpus into a low-resource language. This approach involved training transliteration models with Wikipedia texts and generating a parallel corpus through back-translation, using BLEU, BEER, and ChrF for evaluation. Tonja et al. [33] introduced a novel approach for neural translation in low-resource lan-

guages like Ethiopian, employing parallel texts and the BLEU metric for evaluation. Finally, Singh et al. [34] reported on improving translation quality in low-resource settings for English–Manipuri languages through semi-supervised neural machine translation, using BLEU and ChrF as metrics.

Table 3. Research summary on translation methodologies utilizing a combination of corpora, highlighting the integration of Statistical Machine Translation (SMT), Recurrent Neural Networks (RNNs), and other techniques. The table outlines the reference works, the translation techniques applied, the computational tools used, and a range of metrics for evaluating translation performance, including Precision, Recall, F-Measure, BLEU score, Word Error Rate (WER), and character F-measure (ChrF).

References	Translation Techniques	Tools	Metric
[26]	Transformer, POS tagging	Fairseq	BLEU 36.73, ChrF 54.40, WER 49.30
[27]	Transformer	Not specified	BLEU 51.11, P 51.11, R 53.49%, F 52.27%.
[28]	Transformer	Not specified	BLEU 25.6, ChrF 0.55
[29]	Transformer	Not specified	BLEU 48.62
[30]	Bilingual dictionaries, Syntactic rules	Not specified	BLEU 51.6
[31]	Morphosyntactic parsers	Apertium	WER 0.77
[34]	RNN, transformer, SMT	OpenNMT-py, Moses	BLEU 11.7, ChrF 0.47

3.4. Translation Software Including Low-Resource Languages

Table 4 presents software that performs translation for low-digital-resource languages such as Maya, Wixarika, Otomi, Guarani, Aymara, and Indi, when the majority of the languages included are high-digital-resource languages.

Table 4. Translation tools incorporating low-digital-resource languages. We detail the names of the tools, examples of low-resource languages they support, and the total number of languages available in each tool.

Tool Name	Language	Total Languages
Microsoft Bing	Otomí, Maya	111
Apertium	Hindi, Esperanto	51
Wixárika-traductor	Wixarika	2
Mainumby	Guaraní	2
ModernMT	Aymara, Guaraní, Igbo	200
Google Translate	Guaraní, Aymara	134

4. Methodology

This paper introduces a methodology for the automatic translation of Mixtec–Spanish texts. The proposed approach is divided into four phases (shown in Figure 3), each addressing specific aspects of the translation process.

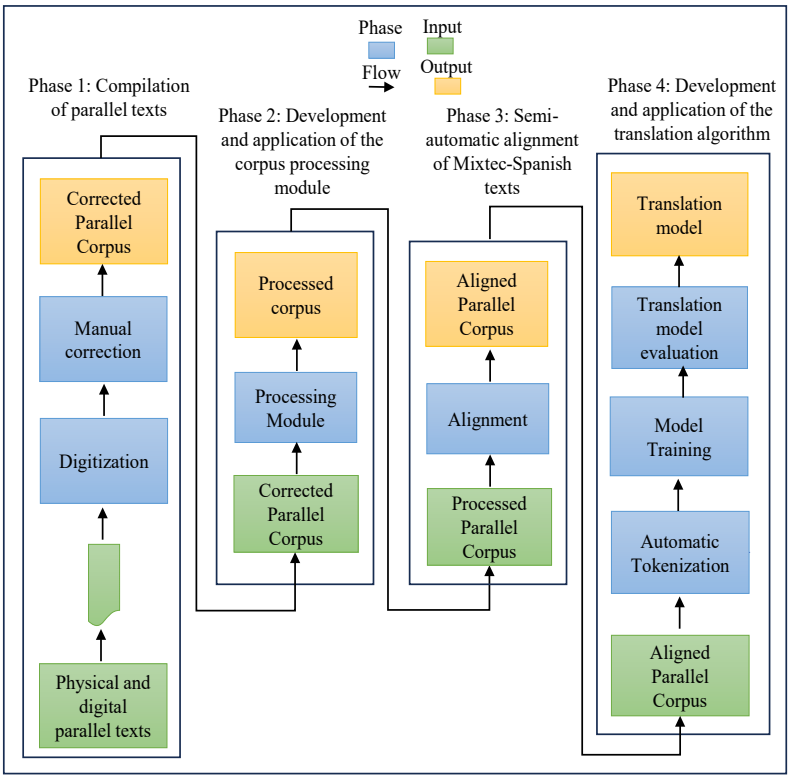

Figure 3. Overview of the automatic translation methodology. Phase 1 involves actively searching for parallel texts in both physical and digital formats as well as digitization and manual correction. Phase 2 encompasses the cleaning and normalization of the parallel corpus. Phase 3 performs a semi-automatic alignment of Mixtec–Spanish texts. Finally, phase 4 consists of developing and applying the translation algorithm.

4.1. Phase 1: Collection of Parallel Texts

Based on the review of the construction of parallel corpora in low-resource languages [31,35,36], we performed corpora search, digitization, and manual correction. Given the low-resource nature of the problem, we implemented techniques to increase the corpora by generating synthetic data with back-translation. The following section describes the search for parallel corpora, generation of synthetic data in Spanish with back-translation, generation of synthetic data with GPT3, digitization, and manual correction.

4.1.1. Corpus Search

We collected texts through in-person visits and via official indigenous language websites in Mexico.

- Searches for parallel texts in person. We carried out the in-person searches at the Faculty of Philosophy and Political Sciences of the Autonomous University of Querétaro. In the in-person searches, we obtained a total of 8 books.
- Searches for parallel texts on official websites. Also, we conducted web searches on the official website gob.mx, INPI, and the Federal Telecommunications Institute. We converted texts found on websites in PDF format into plain text format.

While parallel texts are available, there is a significant shortage of Mixtec language texts necessary for training translations. With this in mind, we generate synthetic data, as described in the next section.

4.1.2. Generation of Synthetic Data in Spanish–Mixtec with GPT-3.5

We used GPT-3.5 to create a Mixtec language dataset, as shown in Figure 4.

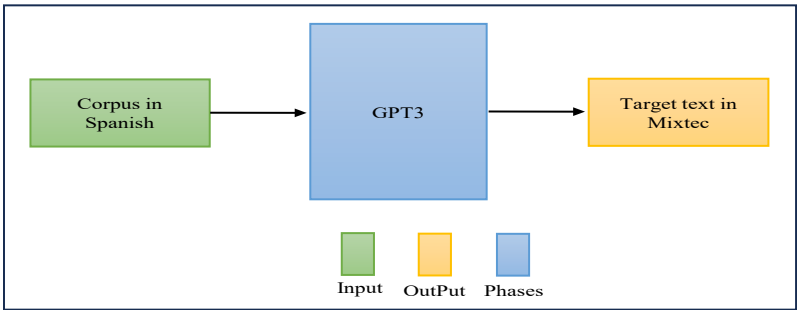

Figure 4. Synthetic data generation architecture. Input is marked in green, the translation phases with GPT-3.5 are in blue, and the final output is indicated in gold.

The Spanish corpus, obtained from back-translation, is fed into the Mixtec synthetic data generation architecture. GPT-3.5 then translates from Spanish to Mixtec, focusing on the Guerrero state variant. The main author, being a native Mixtec speaker, manually checked that GPT-3.5 results were correct. As a result of this procedure, we produced a total of 326 Spanish–Mixtec sentence pairs.

After the data collection from the web search, in-person searches, and synthetic data generation, we performed digitization and manual correction.

4.1.3. Digitization and Manual Correction

Digitization was performed only for the physical books. We scanned each page of the books and saved them in document image format. Then we use the Abbyy FineReader software (https://finereader.en.softonic.com/) to convert the image documents into plain text. The document image reading in Abbyy FineReader was performed in Spanish, as the software does not include Mixtec in its language catalog. During the document reading, two main issues were identified: first, unreadable words in the Mixtec language, and second, confusion in selecting texts adjacent to images. The strategies implemented in text correction are detailed below.

- Identify and Modify Unreadable Words in Mixtec: During the document image reading, we encountered Mixtec words that were unreadable. In these cases, we manually modified the text to the correct word in Mixtec. Additionally, we added the characters identified as causing difficulties in reading the document image to Abbyy FineReader's alphabet. Then, we re-read the document with the added characters, allowing us to correct these issues. Below is the list of characters added to the Abbyy FineReader dictionary: A̲, a̲, E̲, e̲, I̲, i̲, O̲, o̲, U̲, u̲, Ā, ā, Ē, ē, Ī, ī, Ō, ō, Ū, ū. Table 5 gives the words that Abbyy FineReader could not recognize and their corrected forms.
- Confusion in Selecting Texts Adjacent to Images: Sometimes, during the digitization of documents, the software selected both the image and the text, making it difficult to separate the text from the image. In these cases, the solution was to move the selection area over the text and then remove the image from the text.

Table 5. Words not recognized by Abbyy FineReader in Mixtec language.

Words	Corrected Words
vaxi	va̲xi
kpó	ko̲ó
thava	tava
kue´e	kue̲´e̲
Ihjjin	tasi̲i̲n
ʸ? yó	ta̲´ya̲ yó

4.1.4. Final Corpus

We compiled a total of 30 books (8 physical and 22 digital), identifying texts from different categories. Table 6 presents the categories and the total number of texts compiled in each category. Regarding the number of sentences, there is a total of 3424 pairs of Spanish–Mixtec sentences, of which the synthetic data with GPT-3.5 correspond to 326 pairs in total.

Table 6. Categories of texts identified in the corpus.

Type	Quantity
Educational	12
Religious	12
History	6

The categories listed in Table 6 are based on the texts that are most abundant in the corpus. In the case of parallel educational texts, these categories include common phrases, daily conversations, greetings, names of fruits and animals, cooking recipes, and grammar rules for verbs and nouns. The Mexican national anthem is also included in this corpus. We believe that generating translations in the educational domain can help Mixtec speakers to communicate with Spanish speakers. Additionally, by incorporating parallel corpora from other domains like medicine or law, the translation system can be expanded to allow for greater generalization. This method enables the use of various corpus domains to generate accurate translations.

4.2. Phase 2: Developing and Applying the Corpus Processing Module

The research conducted by Tran et al. [36] and Karunesh et al. [37] suggests that corpus processing should be executed in three critical stages: (1) removal of junk words, (2) exclusion of sentences that are not genuinely parallel, and (3) elimination of duplicated sentences. In our project, we focused on stages (1) and (3). We opted not to remove sentences that were not perfectly parallel due to the implementation of a semi-automatic alignment process. Additionally, we introduced a processing stage that involved the removal of special characters and numbers, as well as converting all text to lowercase.

1. Remove junk words. Parallel texts sourced from websites and digitized documents included irrelevant phrases or words, such as editorial information found at the end of texts and chapter titles from the Bible positioned at the beginning of sentences. These were removed from the texts. Additionally, proper names like "IGNACIO", "JUAN", and "TIBURCIO", typically found at the beginning of sentences within dialogues, were also deleted as they were not pertinent to the parallel corpus.
2. Suppress duplicated sentences. The parallel corpus, including synthetic data generated through back-translation and GPT-3.5, contained redundant sentences. These unnecessary repetitions were removed to improve the quality of the parallel corpus.
3. Remove special characters. The corpus featured various special characters, such as parentheses, square brackets, hyphens, and angular quotation marks. These characters were eliminated from the texts because they did not contribute meaningfully to the parallel corpus.
4. Delete numbers and convert to lowercase. Texts retrieved from web searches contained numbers, such as those in prayer lists and Bible verse numbers, which were irrelevant to the parallel corpus. To achieve text standardization, all texts were converted to lowercase.

Automatic Module for Processing Parallel Mixtec–Spanish Corpus

Before processing the parallel corpus, we compiled it into three categories: training, validation, and testing. We generated a total of 6 text files, with each category having two

files—one for the Mixtec language and one for Spanish. Figure 5 shows the total number of words (tokens) and sentences in the corpus.

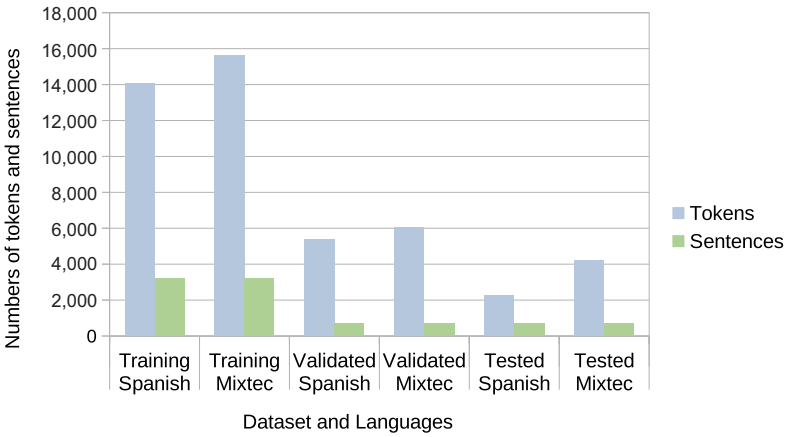

Figure 5. Comparative analysis of token and sentence Counts in processed corpus dataset. The bar chart displays the number of tokens and sentences in the training, validation, and test subsets for both Spanish and Mixtec languages.

In the input file format for processing, the Mixtec language file contains one sentence per line, separated by line breaks. Similarly, the Spanish language text input follows the same format, with one sentence per line, separated by line breaks. Both text files are encoded in UTF-8 for better processing in the translation module.

Table 7 provides an example of the Mixtec and Spanish texts contained in the corpus.

Table 7. Comparison between Spanish and Mixtec parallel corpus.

Spanish	Mixtec
Las cuatro de la tarde	Ka kumi xkuaà
A las diez de la noche	Ka uxi te ñuú
A las tres de la mañana	Ka uni te na a

After we had split the corpus into training and test sets, we developed and implemented a corpus processing module. This module performed several key activities, including the removal of punctuation marks (such as periods, commas, and accents) and converting all text to lowercase to facilitate more effective processing in translation tasks. We outline these activities in pseudocode within Algorithm 1. Our process began with a UTF-8 encoded text file as input and produced a cleaned text file as output. We utilized regular expressions, specified in lines 5 and 10 of the algorithm, for text cleaning. These expressions were designed to remove a variety of characters, including parentheses, brackets, dashes, quotes, numbers, periods, and commas. We applied this text cleaning approach consistently to both the Spanish parallel texts and the Mixtec texts, ensuring an equivalent standard of preprocessing was applied across languages.

Algorithm 1 Preprocessing of Mixtec and Spanish corpus

```
Start
Input: Text file F in Mixtec or Spanish in utf8
Output: Processed Mixtec text file S
function READFILE(F)
    Open file F to read its content
end function
End Read File
function CLEANTEXT(F)
    Define special characters \specialCharacters=«»()[]—“”-;
    Replace the characters in file F
end function
End Method CleanText
Call and store in TextFile = READFILE(F)
Call and store in CleanText = CLEANTEXT(TextFile)
Define regular expression to delete numbers: [0–9]+
Replace and store the processed text in ProcessedText P
function WRITEFILE(F, P)
    Open output file S to write the content P
    Convert the text to lowercase
end function
End Method WriteFile
End
```

Table 8 showcases examples of text that have been processed using Algorithm 1. The “Input” column contains the original, unprocessed texts, while the “Output” column displays the texts after they have undergone processing.

Table 8. Examples of processed texts in Mixtec and Spanish.

Languages	Input	Output
Mixtec	yoòó ña vaxi (yóo)	yoòó ña vaxi yóo
Spanish	(este) mes que viene	este mes que viene
Mixtec	kuiya ña vaxi (yóo)	kuiya ña vaxi yóo
Spanish	(este) año que viene	este año que viene

4.3. Phase 3: Semi-Automatic Alignment of Mixtec–Spanish Texts

Alignment entails linking a sentence in a Spanish text with its equivalent Mixtec translation, utilizing separators such as tabs, hashtags, or vertical bars, or by creating distinct files for the source and target languages.

In this project, our approach centered on aligning sentences by generating separate files for the source and target languages. Given the nature of the available texts, where a Mixtec sentence might be immediately followed by its Spanish translation within a single file, we devised an algorithm to extract these sentences. This process resulted in two distinct files: one for Mixtec and another for Spanish, with each line in the Mixtec file corresponding directly to a line in the Spanish file.

Following the alignment process, a meticulous manual review was performed by an expert in the Mixtec language to verify the accuracy of the sentence pairings.

4.4. Phase 4: Development and Application of the Translation Algorithm

In this phase, we developed and applied the algorithm for the automatic translation of texts written in Mixtec–Spanish.

According to Sennrich et al. [38] and Tonja et al. [39], the procedures to improve translation with monolingual data consist of the following steps:

1. The translation model was trained in the Spanish-to-Mixtec direction with the authentic parallel corpus to generate synthetic data in Mixtec.
2. After training the model with the authentic corpus, the model was then trained with synthetic data in Mixtec.
3. Finally, the synthetic and authentic data were combined to train the model in the Spanish–Mixtec and Mixtec–Spanish directions.

4.4.1. Vocabulary Generation

After cleansing the corpus, we proceeded with tokenization, a process that segments sentences into individual words or subwords. This study specifically concentrated on word-level tokenization, which, as our experiments demonstrated, outperforms other tokenization strategies, especially in languages with scarce digital resources. We implemented word-level tokenization across both recurrent neural network (RNN) and transformer network architectures.

Tokenization results in a collection of words, each associated with its respective tensors or vectors. This conversion of words in a sentence into vectors is referred to as word embedding. Word embedding facilitates the representation of tokens in a multidimensional space, allowing for the clustering of semantically similar terms. We denote the collection of segmented words and their corresponding tensors as vocabularies. In the following sections, we elaborate on the creation of vocabularies for both Mixtec and Spanish languages.

The vocabulary is composed of two files, one for Mixtec and another for Spanish. The format of the vocabulary file is as follows: Token+space+tensor identifier, where *token* corresponds to a word from the vocabulary followed by a blank space and the numerical value of the token. Table 9 shows examples of the vocabulary in Mixtec and Spanish. The format for generating vocabularies is applied in recurrent neural networks and transformers.

Table 9. Examples of vocabularies in Mixtec and Spanish.

Languages	Word Token	Token Value
Mixtec	cha	534
Spanish	de	436
Mixtec	cua	319
Spanish	que	251
Mixtec	ña	287
Spanish	la	333

Table 10 shows the tokenization of words by language.

Table 10. Word tokenization.

Languages	Words
Spanish	5066
Mixtec	3965

4.4.2. Model Architectures

Next, we detail the proposed architectures of the recurrent neural network and transformer network, in addition to the tokenization and vectorization of words.

Recurrent Neural Network

Figure 6 shows the architecture of the proposed recurrent neural network. Table 11 shows the number of layers and hyperparameters.

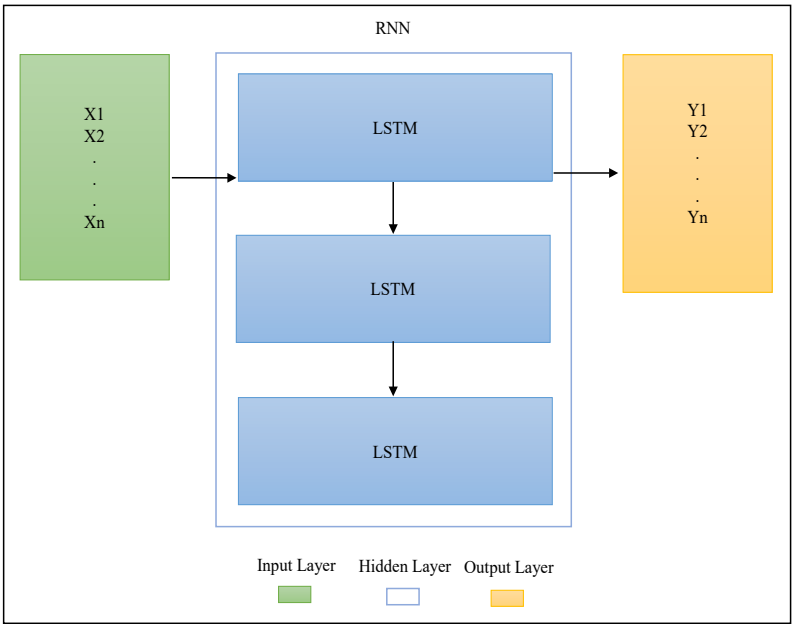

Figure 6. Architecture of the recurrent neural network. Described as X are the input sentences in Mixtec and Spanish, followed by LSTM cells for the recurrent neural network, while Y denotes the target or output text of the architecture.

Each element X_i indicates the sentence in Mixtec and Spanish in the input layer. In the recurrent neural network, we assigned a word embedding length of 128.

Transformer Network

Figure 7 shows the proposed transformer architecture. Table 11 shows the number of layers and hyperparameters.

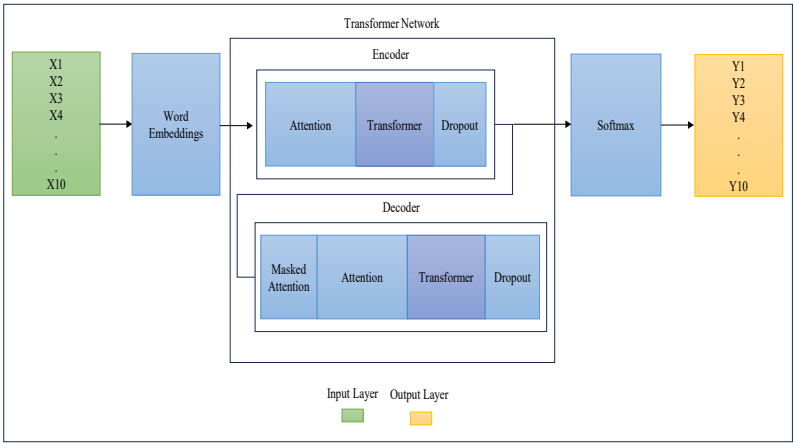

Figure 7. Architecture of the Transformer network. X denotes the corpus in Mixtec and Spanish. Next, we use the Transformer with encoder and decoder layers. The variable Y denotes the output of the Transformer.

Each X_i in the input layer denotes a sentence in Mixtec and Spanish, and subsequently, each word of the sentence was converted into an embedding. For the Transfomer model, we use word embeddings of size 512.

We utilized the OpenNMT-py library due to its capabilities in sequence generation, support for transformer models, recurrent neural networks, and features for automatic evaluation. Our work was conducted within the Google Colab environment.

Algorithm 2 presents the pseudocode for the training process. The instructions on lines 2, 3, and 4 utilize commands provided by OpenNMT-py, with the input parameters specified in a configuration file.

Algorithm 2 Training algorithm

1: Start
2: **Input:** Text file in Mixtec and Spanish in utf8, YAML configuration file
3: **Output:** Meta text file
4: Specify the path of the YAML configuration file.
5: Call *onmt build vocab* to generate vocabularies in Mixtec and Spanish.
6: Call *onmt train* to initiate the training process.
7: Call *onmt translate* to conduct translation tests.
8: End

The parameters listed in Table 11 are specified within a YAML configuration file. This file includes paths for the training and validation corpora, along with paths for logs that detail the training process.

Table 11. Input parameters of the neural network.

No.	Parameter	RNN and BRNN	Transformer
1	Batch size	16	4096
2	Train steps	1000	700
3	Valid steps	500	350
4	Optimizer	Adam	Adam
5	Word vector size	128	512
6	Layers	1, 3	6
7	Learning rate	0.15	2
8	Dropout	0.0, 0.1	0.1, 0.3
9	Hidden size	512	512
10	Heads	0	8
11	Activation function	relu	relu

Figure 8 shows in more detail the architecture of the translation training algorithm. The result is called the target text or translated text. The target text is generated in a plain text file.

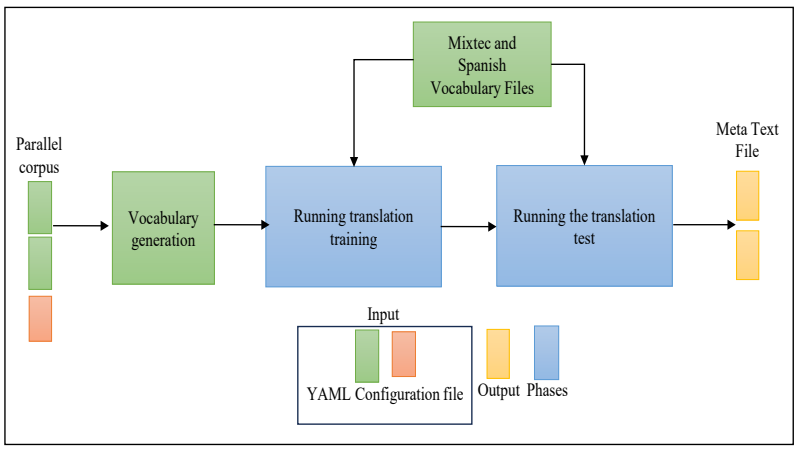

Figure 8. The architecture of the translation training algorithm involves several key steps. Initially, vocabularies for both Mixtec and Spanish are created. This is followed by the execution of the training command. Finally, the process concludes with the execution of the command to perform testing.

5. Results

In this section, we explore the results of the Mixtec–Spanish translation experiments. The evaluation is twofold; it includes an automatic assessment utilizing the BLEU metric and a manual review conducted by an expert in the Mixtec language.

As detailed by Papineni et al. [12], the BLEU metric facilitates the automatic comparison of a translation system's output against a reference translation, ideally crafted by a human expert. BLEU scores range from 0 to 1, with 0 indicating no resemblance between the candidate and reference translations and 1 signifying a perfect match. On the other hand, the resulting TER score represents the edit distance between the translated text and the reference text. A lower TER score indicates a translation that is closer to the reference and thus generally of higher quality. Conversely, a higher TER score suggests that more edits are needed to match the reference translation, indicating a lower-quality translation.

In the experiments conducted, a total of 4568 Mixtec–Spanish sentence pairs were utilized. The distribution was as follows: 70% (3197 pairs) were allocated for training, 15% (685 pairs) for validation, and the remaining 15% (685 pairs) for testing. The focus was specifically on educational texts, which included materials such as the Mexican national anthem, everyday dialogues, descriptions of objects, and sentences in the past tense.

Table 12 shows the outcomes of the automatic evaluation, employing both the BLEU and TER metrics, for translations from Spanish to Mixtec. The highest scores obtained in the experiments are highlighted in bold.

Table 12. Spanish-to-Mixtec evaluation using BLEU and TER metrics. The table compares different model architectures—such as RNN, BRNN with Dropout, and Transformer with Dropout—across varying layers. The highest-scoring results are highlighted in bold for easy identification.

Approaches	BLEU × 100	TER
RNN + Dropout 0.0 + 1 Layer + Batch Size 16	0	100
RNN + Dropout 0.1 + 3 Layer + Batch Size 16	1.44	98.86
BRNN + Dropout 0.0 + 1 Layer + Batch Size 16	0.2	99.35
BRNN + Dropout 0.1 + 3 Layer + Batch Size 16	0	100
Transformer + Dropout 0.1 + 6 Layer + Batch size 4096	**99.87**	0.5
Transformer + Dropout 0.2 + 6 Layer + Batch size 4096	99.80	**0.10**

Table 13 shows the results of the automatic evaluation with the Mixtec–Spanish BLEU and TER metrics.

Table 13. Mixtec-to-Spanish evaluation using BLEU and TER metrics. The table compares different model architectures—such as RNN, BRNN with Dropout, and Transformer with Dropout—across varying layers. The highest-scoring results are highlighted in bold for easy identification.

Approaches	BLEU × 100	TER
RNN + Dropout 0.0 + 1 Layer + Batch Size 16	5.08	84.21
RNN + Dropout 0.1 + 3 Layer + Batch Size 16	0.50	90.44
BRNN + Dropout 0.0 + 1 Layer + Batch size 16	0.73	90.32
BRNN + Dropout 0.1 + 3 Layer + Batch size 16	1.46	95.31
Transformer + Dropout 0.1 + 6 Layer + Batch size 4096	**95.66**	**16.5**
Transformer + Dropout 0.2 + 6 Layer + Batch size 4096	95.59	16.62

Table 14 provides example predictions from Mixtec to Spanish. Each sentence from the first column represents the input to the model to predict the output shown in the second column.

Table 14. Mixtec–Spanish translation examples.

Mixtec	Spanish
Ka´anu	Es grande
¿A ii ra ixatata va?	¿Dónde está él ahora?
Jdjatu yúù va, iyo va'a va yúù	También yo estoy bien
¿Ndia mii kúù tu yóò tu?	¿Dónde vas también tú?
¿A iyo tixú'u djan a kù ú va?	¿Cuántos son tus chivos?
Ñu´u	Tierra buena
Koó ñöú va'a	Tierra mala
¿Ndee iyo do?	¿Dónde vives?
Libiní ña no ñú'u itai	Esa tierra es muy buena
Va'a in ti	Es un buen animal

The best Spanish-to-Mixtec translation model was the Transformer with six layers and Dropout of 0.1, achieving a BLEU score of 99.87 and a TER of 0.5.

For the Mixtec-to-Spanish translation, the best model was the same Transformer architecture with six layers and a Dropout of 0.1, achieving a BLEU score of 95.66 and TER of 16.5.

In terms of training durations, the RNN and BRNN models completed their training in an average of 30 min, whereas the training time for the Transformer models extended to four hours.

Table 15 offers a comparative analysis between our translation approach for Mixtec–Spanish and related low-resource language work highlighted in the literature review. These results demonstrate a significant improvement, showing a 28% increase in BLEU score with respect to scores obtained in related studies.

Table 15. Comparison with respect to related works in low-digital-resource languages.

References	BLEU
Zacarias et al. [15]	+5
Mager et al. [17]	23.47
Knowles et al. [18]	16.2
Dione et al. [22]	37.5
Rubino et al. [23]	26.8
Our approach	52.39 *

* Average BLEU score between BRNN, RNN, and Transformer.

6. Discussion

Our review of the state of the art in machine translation revealed several challenges in translating low-resource languages, including the absence of natural language processing tools, limited text availability, and the complexity of tonal variations. The following points detail these challenges:

- Lack of Natural Language Processing Tools: Resources such as grammatical taggers, normalizers, and automatic tokenizers, crucial for processing these languages, are often unavailable.
- Restricted Corpus Size: Corpora in low-resource languages are typically small, leading to a lack of sufficient texts for experimentation and validation in translation processes.
- Tonal Complexity: The diversity of tonal variations in low-resource languages adds complexity to the application of natural language processing algorithms, including machine translation and grammatical tagging.
- Inconsistency in Texts: A notable challenge is the inconsistency in texts within parallel corpora used for training. Such inconsistencies can adversely affect machine translation efficiency, as highlighted by Mager et al. [40] and Matos Veliz et al. [41] in their research on neural network-based normalization for low-resource language environments.

- Lack of keyboards: In low-resource languages, the vast majority of them do not have the characters defined in a keyboard that allows writing texts in these languages.
- Lack of optical character recognition: The vast majority of languages with low digital resources are not incorporated into the language catalogs of the recognition tools, which allow identifying and correcting the characters in these languages.

7. Conclusions

In this study, we successfully developed the first methodology for Mixtec–Spanish and Spanish–Mixtec translation in Mexico. Our approach achieved a remarkable 28% improvement in BLEU scores compared to existing works. This enhancement was possible through the integration of neural network techniques, corpus processing, synthetic data generation, and back-translation. We found that preprocessing the corpus by normalizing it significantly enhanced translation accuracy and improved BLEU scores. During this phase, we encountered and overcame challenges with the transformer network in decoding Mixtec texts. We innovated an algorithm to effectively remove or replace unique characters in Mixtec syntax, which was problematic for translation.

Furthermore, the use of synthetic data generation strategies significantly expanded our parallel corpus. The methodology we developed holds excellent promise for adapting text translations in other indigenous languages such as Amuzgo, Tlapaneco, and Zoque. We also created Mixtec–Spanish translation memories, which will be an asset for the creation of voice recognition, grammatical tagging, linguistic studies, multilingual translation involving other Mexican indigenous languages, automatic summary generation, speech-to-text conversion, sentiment analysis, and language detection.

In future work, we plan to expand the parallel corpus and refine the results further by incorporating synthetic data generation with more advanced models such as GPT-4 [42], Gemini Ultra [43], or Claude 3 Opus [44], as well as adding other augmentation techniques such as noise injection or transfer learning from high-resource language models.

Author Contributions: Conceptualization, H.S.-B., D.-M.C.-E. and N.-A.C.-S.; methodology H.S.-B. and D.-M.C.-E.; software, H.S.-B.; validation, H.S.-B., D.-M.C.-E. and N.-A.C.-S.; formal analysis, H.S.-B., D.-M.C.-E. and N.-A.C.-S.; investigation, H.S.-B., D.-M.C.-E. and N.-A.C.-S.; resources H.S.-B., D.-M.C.-E., N.-A.C.-S., J.-A.R.-G., T.G.-R. and J.T.; writing—original draft preparation, H.S.-B., D.-M.C.-E. and N.-A.C.-S.; writing—review and editing, H.S.-B., D.-M.C.-E., N.-A.C.-S., J.-A.R.-G., T.G.-R. and J.T.; visualization, H.S.-B., D.-M.C.-E., N.-A.C.-S., J.-A.R.-G., T.G.-R. and J.T.; supervision, D.-M.C.-E. and N.-A.C.-S.; project administration, H.S.-B., D.-M.C.-E., N.-A.C.-S., J.-A.R.-G., T.G.-R. and J.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: We thank the Autonomous University of Querétaro and the National Council of Humanities, Sciences, and Technologies (CONAHCYT) for their support. Additionally, we acknowledge the use of two AI tools: Grammarly Assistant for improving the grammar, clarity, and overall readability of the manuscript, and GPT-4 to help with the wording, formatting, and styling of the manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Oliver, A. MTUOC: Easy and free integration of NMT systems in professional translation environments. In Proceedings of the 22nd Annual Conference of the European Association for Machine Translation, Lisboa, Portugal, 3–5 November 2020; pp. 467–468.
2. Vázquez, A.; Pinto, D.; Lavalle, J.; Jiménez, H.; Vilariño, D. Grammatical Inference of Semantic Components in Dialogues. *Comput. Sist.* **2020**, *24*, 715–718. [CrossRef]
3. Gutierrez-Vasques, X. Bilingual lexicon extraction for a distant language pair using a small parallel corpus. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop, Denver, CO, USA, 31 May–5 June 2015; Inkpen, D., Muresan, S., Lahiri, S., Mazidi, K., Zhila, A., Eds.; pp. 154–160. [CrossRef]
4. Turki Khemakhem, I.; Jamoussi, S.; Ben Hamadou, A. POS tagging without a tagger: Using aligned corpora for transferring knowledge to under-resourced languages. *Comput. Y Sist.* **2016**, *20*, 667–679. [CrossRef]
5. Gutierrez-Vasques, X.; Sierra, G.; Pompa, I.H. Axolotl: A Web Accessible Parallel Corpus for Spanish-Nahuatl. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia, 23–28 May 2016; Calzolari, N., Choukri, K., Declerck, T., Goggi, S., Grobelnik, M., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., et al., Eds.; pp. 4210–4214.
6. Ebrahimi, A.; Mager, M.; Oncevay, A.; Chaudhary, V.; Chiruzzo, L.; Fan, A.; Ortega, J.; Ramos, R.; Rios, A.; Meza Ruiz, I.V.; et al. AmericasNLI: Evaluating Zero-shot Natural Language Understanding of Pretrained Multilingual Models in Truly Low-resource Languages. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland, 22–27 May 2022; Muresan, S., Nakov, P., Villavicencio, A., Eds.; pp. 6279–6299. [CrossRef]
7. INALI. CATÁLOGO DE LAS LENGUAS INDÍGENAS NACIONALES. 2008. Available online: https://www.inali.gob.mx/clin-inali/ (accessed on 27 May 2023).
8. INEGI. Hablantes de Lengua Indígena en México. 2020. Available online: https://cuentame.inegi.org.mx/poblacion/lindigena.aspx (accessed on 27 May 2023).
9. INPI. Mixtecos—Lengua, Atlas de los Pueblos Indígenas de México. 2008. Available online: http://atlas.inpi.gob.mx/4964-2/ (accessed on 27 May 2023).
10. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
11. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All You Need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Red Hook, NY, USA, 4–9 December 2017; NIPS'17; pp. 6000–6010.
12. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: A Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 6–12 July 2002; pp. 311–318. [CrossRef]
13. Snover, M.; Dorr, B.; Schwartz, R.; Micciulla, L.; Makhoul, J. A Study of Translation Edit Rate with Targeted Human Annotation. In Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers, Cambridge, MA, USA, 8–12 August 2006; pp. 223–231.
14. Mager Hois, J.M.; Barrón Romero, C.; Meza Ruiz, I.V. Traductor estadístico wixarika—Español usando descomposición morfológica. *Comtel* **2016**, *2016*, 63–68
15. Zacarías Márquez, D.; Meza Ruiz, I.V. Ayuuk-Spanish Neural Machine Translator. In Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas, Online, 11 June 2021; pp. 168–172. [CrossRef]
16. Mager, M.; Oncevay, A.; Ebrahimi, A.; Ortega, J.; Rios, A.; Fan, A.; Gutierrez-Vasques, X.; Chiruzzo, L.; Giménez-Lugo, G.; Ramos, R.; et al. Findings of the AmericasNLP 2021 Shared Task on Open Machine Translation for Indigenous Languages of the Americas. In Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas, Online, 11 June 2021; pp. 202–217. [CrossRef]
17. Mager, M.; Meza, I. Retos en construcción de traductores automáticos para lenguas indígenas de México. *Digit. Scholarsh. Humanit.* **2021**, *36*, i43–i48. [CrossRef]
18. Knowles, R.; Littell, P. Translation Memories as Baselines for Low-Resource Machine Translation. In Proceedings of the Thirteenth Language Resources and Evaluation Conference, Marseille, France, 20–25 June 2022; pp. 6759–6767.
19. Gezmu, A.M.; Nürnberger, A.; Bati, T.B. Extended Parallel Corpus for Amharic-English Machine Translation. In Proceedings of the Thirteenth Language Resources and Evaluation Conference, Marseille, France, 20–25 June 2022; pp. 6644–6653.
20. Stanojević, M.; Sima'an, K. Fitting Sentence Level Translation Evaluation with Many Dense Features. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 202–206. [CrossRef]
21. Wang, W.; Peter, J.T.; Rosendahl, H.; Ney, H. CharacTer: Translation Edit Rate on Character Level. In Proceedings of the First Conference on Machine Translation, 11–12 August 2016; Volume 2, pp. 505–510. [CrossRef]
22. Dione, C.M.B.; Lo, A.; Nguer, E.M.; Ba, S. Low-Resource Neural Machine Translation: Benchmarking State-of-the-Art Transformer for Wolof<->French. In Proceedings of the Thirteenth Language Resources and Evaluation Conference, Marseille, France, 20–25 June 2022; pp. 6654–6661.
23. Rubino, R.; Marie, B.; Dabre, R.; Fujita, A.; Utiyama, M.; Sumita, E. Extremely Low-Resource Neural Machine Translation for Asian Languages. *Mach. Transl.* **2020**, *34*, 347–382. [CrossRef]

24. Kumar, A.; Mundotiya, R.K.; Pratap, A.; Singh, A.K. TLSPG: Transfer learning-based semi-supervised pseudo-corpus generation approach for zero-shot translation. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 6552–6563. [CrossRef]
25. Imankulova, A.; Sato, T.; Komachi, M. Filtered Pseudo-Parallel Corpus Improves Low-Resource Neural Machine Translation. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2019**, *19*, 1–16. [CrossRef]
26. Hlaing, Z.Z.; Thu, Y.K.; Supnithi, T.; Netisopakul, P. Improving neural machine translation with POS-tag features for low-resource language pairs. *Heliyon* **2022**, *8*, e10375. [CrossRef]
27. Hujon, A.V.; Singh, T.D.; Amitab, K. Transfer Learning Based Neural Machine Translation of English-Khasi on Low-Resource Settings. *Procedia Comput. Sci.* **2023**, *218*, 1–8. [CrossRef]
28. Meetei, L.S.; Singh, S.M.; Singh, A.; Das, R.; Singh, T.D.; Bandyopadhyay, S. Hindi to English Multimodal Machine Translation on News Dataset in Low Resource Setting. *Procedia Comput. Sci.* **2023**, *218*, 2102–2109. [CrossRef]
29. Maimaiti, M.; Liu, Y.; Luan, H.; Sun, M. Enriching the transfer learning with pre-trained lexicon embedding for low-resource neural machine translation. *Tsinghua Sci. Technol.* **2022**, *27*, 150–163. [CrossRef]
30. Sethi, N.; Dev, A.; Bansal, P.; Sharma, D.K.; Gupta, D. Hybridization Based Machine Translations for Low-Resource Language with Language Divergence. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 2022, *just accepted*. [CrossRef]
31. Pirinen, F.; Wiechetek, L. Building an Extremely Low Resource Language to High Resource Language Machine Translation System from Scratch. In Proceedings of the 18th Conference on Natural Language Processing (KONVENS 2022), Potsdam, Germany, 12–15 September 2022; pp. 150–155.
32. Karakanta, A.; Dehdari, J.; Genabith, J. Neural Machine Translation for Low-Resource Languages without Parallel Corpora. *Mach. Transl.* **2018**. *32*, 167–189. [CrossRef]
33. Tonja, A.; Kolesnikova, O.; Arif, M.; Gelbukh, A.; Sidorov, G. Improving Neural Machine Translation for Low Resource Languages Using Mixed Training: The Case of Ethiopian Languages. In *Advances in Computational Intelligence—21st Mexican International Conference on Artificial Intelligence, MICAI 2022, Monterrey, Mexico, 24–29 October 2022, Proceedings, Part I*; Pichardo Lagunas, O., Martínez Seis, B., Martínez-Miranda, J., Eds.; Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer: Berlin/Heidelberg, Germany, 2022; pp. 30–40. [CrossRef]
34. Singh, S.M.; Singh, T.D. Low Resource Machine Translation of English—Manipuri: A Semi-Supervised Approach. *Expert Syst. Appl.* **2022**, *209*, 118187. [CrossRef]
35. Gutiérrez-Vasques, X.; Vilchis Vargas, E.C.; Cerbón Ynclán, R. Recopilación de un corpus paralelo electrónico para una lengua minoritoria: El caso del nahuatl-español. In Proceedings of the Primer Congreso Internacional el Patrimonio Cultural y las Nuevas Tecnologías, Ciudad de Mexico, Mexico, 3–5 November 2014.
36. Tran, P.; Nguyen, T.; Vu, D.H.; Tran, H.A.; Vo, B. A Method of Chinese-Vietnamese Bilingual Corpus Construction for Machine Translation. *IEEE Access* **2022**, *10*, 78928–78938. [CrossRef]
37. Arora, K.K.; Agrawal, S.S. Pre-Processing of English-Hindi Corpus for Statistical Machine Translation. *Comput. Y Sist.* **2017**, *21*, 725–737. [CrossRef]
38. Sennrich, R.; Haddow, B.; Birch, A. Improving Neural Machine Translation Models with Monolingual Data. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; Volume 1, pp. 86–96. [CrossRef]
39. Tonja, A.L.; Kolesnikova, O.; Gelbukh, A.; Sidorov, G. Low-Resource Neural Machine Translation Improvement Using Source-Side Monolingual Data. *Appl. Sci.* **2023**, *13*, 1201. [CrossRef]
40. Mager, M.; Rosales, M.J.; Çetinoğlu, Ö.; Meza, I. Low-resource neural character-based noisy text normalization. *J. Intell. Fuzzy Syst.* **2019**, *36*, 4921–4929. [CrossRef]
41. Matos Veliz, C.; De Clercq, O.; Hoste, V. Is neural always better? SMT versus NMT for Dutch text normalization. *Expert Syst. Appl.* **2021**, *170*, 114500. [CrossRef]
42. Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. GPT-4 technical report. *arXiv* **2023**, arXiv:2303.08774.
43. Team, G.; Anil, R.; Borgeaud, S.; Wu, Y.; Alayrac, J.B.; Yu, J.; Soricut, R.; Schalkwyk, J.; Dai, A.M.; Hauth, A.; et al. Gemini: A family of highly capable multimodal models. *arXiv* **2023**, arXiv:2312.11805.
44. Anthropic. Introducing the Next Generation of Claude. 2024. Available online: https://www.anthropic.com/news/claude-3-family (accessed on 28 March 2024).