RECONOCIMIENTO CONTINUO Y TRADUCCIÓN DE LA LENGUA DE SEÑAS EMPLEANDO UNA ARQUITECTURA *TRANSFORMER*

CHRISTIAN EDUARDO RUIZ LAGOS

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2023

RECONOCIMIENTO CONTINUO Y TRADUCCIÓN DE LA LENGUA DE SEÑAS EMPLEANDO UNA ARQUITECTURA *TRANSFORMER*

CHRISTIAN EDUARDO RUIZ LAGOS

Trabajo de Grado para optar al título de:
Ingeniero de Sistemas

Director:
Fabio Martínez Carrillo
Doctor en Ingeniería de sistemas y computación

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2023

AGRADECIMIENTOS

A mi madre y abuela, quienes son todo lo que tengo y han sido las personas que siempre han estado ahí para mí de forma incondicional, que me han apoyado en todas y cada una de mis decisiones.

A mi director de tesis, el profesor Fabio Martínez, por alentarme a no rendirme en este proceso y por aconsejarme en momentos donde realmente lo necesité.

A la Universidad Industrial de Santander, por ofrecerme una educación de alta calidad y por enseñarme que hay que trabajar duro para lograr nuestros objetivos.

CONTENIDO

LISTA DE FIGURAS

pág.

LISTA DE TABLAS

pág.

Resumen

Titulo: Reconocimiento continuo y traducción de la lengua de señas empleando una arquitectura *Transformer*. [*]

Autor: Christian Eduardo Ruiz Lagos. [**]

Palabras Clave: traducción de la lengua de señas, codificador-decodificador, Transformer.

Descripción: Los sistemas de traducción de la lengua de señas (SLT, por su denominación en inglés) apoyan la comunicación de personas con discapacidad auditiva al encontrar equivalencias entre las lenguas de señas y el lenguaje hablado. Sin embargo, esta tarea es desafiante debido a las múltiples variaciones presentes en las señas, la complejidad del lenguaje y la inherente riqueza de expresiones. Los enfoques computacionales basados en visión por computador han demostrado ser capaces de apoyar la traducción de la lengua de señas. No obstante, estos enfoques todavía presentan limitaciones para abarcar la variabilidad de los gestos y traducir secuencias largas. Este trabajo presenta una arquitectura basada en *Transformers* que codifica parámetros espacio-temporales presentes en los gestos, preservando información espacial local y de largo plazo mediante el uso de convoluciones y múltiples mecanismos de atención. El enfoque propuesto se validó en el conjunto de datos de Lengua de Señas Colombiana (CoL-SLTD), superando los enfoques base y logrando un puntaje BLEU4 del 51,37 %. Además, el enfoque propuesto se validó en el conjunto de datos RWTH-PHOENIX-Weather-2014T (PHOENIX14T), logrando un puntaje BLEU4 de 15,24 %, lo que demuestra su robustez y efectividad para manejar escenarios más realistas.

---

[*] Trabajo de grado

[**] Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: Fabio Martínez Carrillo, Ph.D.

Abstract

Title: Continuous sign language recognition and translation using a Transformer architecture. [*]

Author: Christian Eduardo Ruiz Lagos [**]

Keywords: sign language translation, encoder-decoder, Transformer.

Description: Sign Language Translation (SLT) systems support hearing-impaired people communication by finding equivalences between signed and spoken languages. This task is however challenging due to multiple sign variations, complexity in language and inherent richness of expressions. Computational approaches have evidenced capabilities to support SLT. Nonetheless, typical approaches remain limited to cover gestures variability and support long sequence translations. This work introduces a Transformer-based architecture that encodes spatio-temporal motion gestures, preserving both local and long-range spatial information through the use of multiple convolutional and attention mechanisms. The proposed approach was validated on the Colombian Sign Language Translation Dataset (CoL-SLTD) outperforming baseline approaches, and achieving a BLEU4 of 51.37 %. Additionally, the proposed approach was validated on the RWTH-PHOENIX-Weather-2014T (PHOENIX14T), achieving a BLEU4 score of 15,24 %, demonstrating its robustness and effectiveness in handling real-world variations.

---

[*] Degree work

[**] School of Physical-Mechanical Engineering. School of Systems and Computer Engineering. Advisor: Fabio Martínez Carrillo Ph.D.

Introducción

La lengua de señas es la principal alternativa de comunicación para aproximadamente 466 millones de personas con discapacidad auditiva en todo el mundo[1]. Sin embargo, la comunicación efectiva entre personas sordas puede resultar desafiante debido a las diversas diferencias culturales, regionales y socio-lingüísticas presentes en la lengua de señas[2]. Además, la comunicación con el resto de la sociedad está severamente limitada debido a la falta de conocimiento de las lenguas de señas, creando un obstáculo significativo para la inmersión completa de las personas sordas en actividades diarias[3]. Por lo tanto, es crucial contar con alternativas de soporte y traducción que faciliten la comunicación entre personas con discapacidades auditivas y el resto de la sociedad. Como consecuencia, modelar, caracterizar y desarrollar métodos computacionales que puedan permitir una traducción continua de la lengua de señas sigue siendo una tarea desafiante.

Las lenguas de señas son lenguajes visuales complejos que involucran parámetros espacio-temporales presentes en la articulación de las señas, como el movimiento, la posición, la forma y la orientación de las manos y dedos. En esta lengua también se incluyen componentes no manuales correspondientes a la posición corporal y el

---

1 World Health Organization. *Estimates of Hearing Loss 2021*. Jul. de 2018. URL: https://www.who.int/deafness/estimates/en/ (visitado 26-09-2023).

2 Wendy Sandler y Diane Lillo-Martin. *Sign Language and Linguistic Universals*. Cambridge University Press, 2006.

3 Bencie Woll, RL Sutton-Spence y Frances Elton. “Multilingualism: The global approach to sign languages”. En: *The sociolinguistics of sign languages*. Cambridge University Press, 2001, págs. 8-32.

movimiento de los ojos y la boca[4]. Por lo tanto, su modelamiento y caracterización computacional resulta desafiante incluso para estrategias del estado del arte.

Actualmente, los enfoques más avanzados para la traducción de la lengua de señas se fundamentan principalmente sobre modelos secuencia a secuencia con arquitecturas codificador-decodificador[5]. Estos enfoques suelen basarse en redes neuronales recurrentes (RNN), las cuales pueden limitar la capacidad de la red para capturar contextos temporales a largo plazo propios del lenguaje[6]. Para abordar estos desafíos, se han propuesto enfoques más avanzados, como los *Transformers*, los cuales permiten capturar de manera más eficiente las dependencias temporales a largo plazo gracias a sus múltiples mecanismos de atención. Esto les permite ponderar la importancia de los gestos clave presentes en una conversación[7]. Sin embargo, los enfoques actuales basados en *Transformers* para la traducción de la lengua de señas tienden a perder información espacio-temporal importante presente en las señas, ya que típicamente utilizan representaciones de embebidos unidimen-

---

4 Pamela Perniss, Jenny Lu y Gary Morgan. "The visual complexity of sign languages". En: *Cognitive Science* 42.S3 (2018), págs. 911-939. DOI: 10.1111/cogs.12566.

5 Necati Cihan Camgoz et al. "Neural machine translation for sign languages: A survey". En: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics. 2018, págs. 47-53; Xiaoying Hu et al. "Sign language translation: A deep learning-based approach". En: *2020 IEEE International Conference on Signal and Image Processing (ICSIP)*. IEEE. 2020, págs. 131-136. DOI: 10.1109/ICSIP50750.2020.9376157.

6 Huan Liu et al. "Sign language recognition and translation with recurrent neural networks and visual attention". En: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2018, págs. 7266-7273. DOI: 10.1109/ICRA.2018.8460515; Zhaopeng Zhang et al. "Sign language translation using multimodal recurrent neural networks with visual attention". En: *2019 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2019, págs. 1368-1374. DOI: 10.1109/ICRA.2019.8793686.

7 Necati Cihan Camgoz et al. *Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation*. 2020. arXiv: 2003.13830 [cs.CV].

sionales que pierden la información espacial. Además, durante la comunicación, la información relacionada al movimiento es relevante para proporcionar información contextual adicional. Por lo tanto, se requieren nuevos esquemas que aprovechen de mejor forma la información espacio-temporal relacionada con la pose, los gestos y los movimientos cinemáticos primitivos.

La principal contribución de este trabajo es una arquitectura basada en *Transformers* que utiliza una representación espacial de las señas más adecuada para modelar dependencias espaciales locales y de largo plazo. Para lograr esto, mejoramos la estrategia de codificación realizando cambios en la representación de entrada, introduciendo imágenes en flujo óptico, lo cual ayuda a la red a enfocarse en patrones cinemáticos de movimiento. A partir de esto, se extraen representaciones bidimensionales (2D) utilizando convoluciones para obtener mapas de características que busquen preservar información espacial local relevante en los gestos. En este trabajo también exploramos codificadores posicionales en 2D (2D positional encodings) que permiten aprovechar las composiciones geométricas y estructurales de cada imagen. Además, incluimos un mecanismo de auto-atención en 2D que toma los mapas de características previamente extraídos y calcula una atención a nivel de píxel para resaltar las dependencias a largo plazo y actuar como un complemento a las convoluciones. Nuestro enfoque propuesto sugiere una mejora a la hora de detectar variaciones relacionadas con los articuladores manuales y no manuales, además de entrenarse utilizando un esquema secuencia a secuencia.

1. Fundamentos y Trabajo Previo

En la literatura reciente, los modelos de traducción secuencia-a-secuencia han marcado un notable avance en la tarea de traducción automática de señas[8]. En esencia, estos modelos utilizan arquitecturas de tipo codificador − decodificador, que de manera natural encuentran correspondencias entre dos secuencias. Dentro de sus componentes −para el problema de traducción−, estas arquitecturas usan unidades recurrentes (RNN) para obtener representaciones compactas con respecto a la coherencia temporal, así como también unidades convolucionales (CNN) para obtener representaciones visuales robustas. Hoy en día, estas arquitecturas han sido fortalecidas con mecanismos de atención, lo cual manifiesta un interés latente en la comunidad por proponer nuevas representaciones de este problema que permitan capturar relaciones más complejas. Teniendo en cuenta el estado del arte, al igual que los intereses del proyecto, a continuación, nos permitimos ampliar la información sobre cada una de estas estrategias computacionales.

1.1. Arquitecturas Codificador - Decodificador

Dentro del dominio de los modelos de traducción de secuencias, cuyas entradas y salidas suelen ser secuencias de longitud variable, una herramienta comúnmente utilizada son los modelos codificador − decodificador[9]. En la actualidad, una gran cantidad de modelos están construidos bajo este tipo de arquitecturas, haciendo posible tareas tales como la traducción automática, generación de texto, modelado del

---

[8] Camgoz et al., “Neural machine translation for sign languages: A survey”.

[9] Kyunghyun Cho et al. “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation”. En: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014.

lenguaje, entre otras[10]. Los modelos codificador − decodificador son arquitecturas relativamente nuevas; propuesta inicialmente por Ilya Sutskever et. al. (2014)[11] para tareas de traducción automática y de forma más general, aprendizaje secuencia a secuencia (seq2seq, por su denominación en Inglés).

De forma general, el codificador extrae una representación (embebido) de longitud fija de una secuencia de entrada de longitud variable y el decodificador genera una traducción de longitud también variable de esta representación. Dentro del estado del arte en esquemas secuencia a secuencia para el modelado del lenguaje y la traducción, tanto el codificador como el decodificador normalmente corresponden a redes neuronales recurrentes (RNN). Estas unidades operan sobre la coherencia temporal, aprendiendo índices de correspondencia que permiten explorar la memoria de las secuencias y operar temporalmente como filtros no lineales. Sin embargo, existen otros enfoques dentro de los cuales se observa el uso de redes neuronales convolucionales (CNN) que también han demostrado tener un buen desempeño en el modelado de secuencias, específicamente en tareas como síntesis de audio y traducción automática. Estas arquitecturas tienen ventajas relacionadas con rapidez y capacidad de paralelismo debido a su procesamiento no-secuencial[12].

---

[10] Dzmitry Bahdanau, Kyunghyun Cho y Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. 2016. arXiv: 1409.0473 [cs.CL]. URL: https://arxiv.org/abs/1409.0473; Alex Graves. "Generating Sequences with Recurrent Neural Networks". En: *arXiv preprint arXiv:1308.0850* (2014); Nal Kalchbrenner, Edward Grefenstette y Phil Blunsom. "Recurrent Continuous Translation Models". En: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2013.

[11] Ilya Sutskever, Oriol Vinyals y Quoc V Le. "Sequence to sequence learning with neural networks". En: *Advances in neural information processing systems*. 2014, págs. 3104-3112.

[12] Shaojie Bai, J. Zico Kolter y Vladlen Koltun. *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*. 2018. arXiv: 1803.01271 [cs.LG].

1.2. Esquemas Recurrentes

Las redes neuronales recurrentes (RNN)[13], así como las redes de gran memoria de corto plazo (LSTM)[14] y las redes neuronales recurrentes con puertas (GRUs)[15], se han establecido firmemente como enfoques del estado del arte en el modelado de secuencias y problemas de transducción como la traducción automática[16]. Los modelos recurrentes siguen la propiedad de Márkov de primer orden, donde cada estado depende del estado visto anteriormente. En particular, las RNN generan una secuencia de estados ocultos tal que:

$$h_t = f_W(h_{t-1}, x_t)$$
$$h_t = tanh(W_{hh}h_{t-1} + W_{hx}x_t + b)$$

donde h_{t-1} corresponde al estado oculto (activación) anterior, x_t a la entrada para la posición t y f_W a aplicar una transformación ($WX + b$), seguido de una función de activación (no lineal) a los datos de entrada. En particular, las LSTM se constituyen como la opción más plausible dentro de las distintas arquitecturas de RNN, al gozar de un tipo de memoria adicional (*cell memory*) que les permite aprender dependencias (correlaciones) a largo plazo, las cuales son un factor clave en tareas de traducción[17]. Estas dependencias permiten una introducción del contexto

---

[13] Jeffrey L. Elman. "Finding Structure in Time". En: *Cognitive Science* 14.2 (1990), págs. 179-211.

[14] Sepp Hochreiter y Jürgen Schmidhuber. "Long Short-Term Memory". En: *Neural Computation* 9.8 (1997), págs. 1735-1780.

[15] Cho et al., "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation".

[16] Minh-Thang Luong, Hieu Pham y Christopher D. Manning. "Effective Approaches to Attention-based Neural Machine Translation". En: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2015, págs. 1412-1421.

[17] Felix A. Gers, Jürgen Schmidhuber y Fred Cummins. "Learning to Forget: Continual Predic-

en las secuencias generadas, dado que, en general, el lenguaje se da por medio de asociaciones, por lo cual, para dar un sentido y ser entendido, debe asumirse de forma holística. Es decir, en una oración, una palabra puede complementar su sentido con otra u otras palabras presentes más adelante o con anterioridad dentro de la misma. No obstante, debido a su naturaleza recurrente, la capacidad de aprender estas dependencias puede aún verse acotada por factores como la longitud de las secuencias (entradas o salidas largas). Asimismo, los modelos recurrentes no pueden ser paralelizados, lo cual se traduce en entrenamientos más lentos y, por tanto, costosos[18]. Esto último resulta en un factor a considerar a la hora de tratar con modelos de aprendizaje profundo, que requieren de grandes cantidades de datos para su correcto aprendizaje.

Dentro de los modelos del estado del arte, es común combinar arquitecturas recurrentes y convolucionales con el fin de abarcar un mayor dominio en tareas de traducción. Ejemplos de esto son la introducción de las Redes Convolucionales Recurrentes (CRNN), las cuales han tenido buen desempeño en tareas de reconocimiento visual[19], así como segmentación y detección de acciones en vídeo; y las Redes Convolucionales Recurrentes Gráficas (GCRN)[20], para estructuras de datos no-regulares (secuencias de datos basadas en grafos).

Del mismo modo, la incorporación de mecanismos de atención ha resultado en mejoras notables en el desempeño de los modelos, por lo que resulta ser un compo-

---

tion with LSTM". En: *Neural Computation* 12.10 (1999), págs. 2451-2471. DOI: 10.1162/089976699300016629.

[18] Razvan Pascanu, Tomas Mikolov y Yoshua Bengio. "On the Difficulty of Training Recurrent Neural Networks". En: *International Conference on Machine Learning (ICML)*. 2013, págs. 1310-1318.

[19] Jeff Donahue et al. *Long-term Recurrent Convolutional Networks for Visual Recognition and Description*. 2016. arXiv: 1411.4389 [cs.CV]. URL: https://arxiv.org/abs/1411.4389.

[20] Youngjoo Seo et al. *Structured Sequence Modeling with Graph Convolutional Recurrent Networks*. 2016. arXiv: 1612.07659 [stat.ML]. URL: https://arxiv.org/abs/1612.07659.

nente fundamental a la hora de consolidar esquemas que puedan hacer frente al estado del arte.

1.3. Mecanismos de Atención y arquitecturas *Transformer*

Un enfoque más reciente resulta ser la arquitectura *Transformer*[21], que se basa completamente en mecanismos de atención, prescindiendo de redes RNN y CNN. Esta arquitectura está diseñada para trabajar con secuencias y exhibe ventajas tanto en procesamiento como en desempeño. Por lo tanto, se ha consolidado fuertemente en el estado del arte de los modelos codificador-decodificador en el ámbito del procesamiento del lenguaje natural (*NLP*), superando a los modelos secuencia a secuencia basados en esquemas recurrentes. El mecanismo de atención en Aprendizaje Profundo modela estrategias para dar mayor importancia o "prestar mayor atención" a la información más relevante, es decir, los datos clave para lograr una mayor correlación con la salida esperada. Dzmitry Bahdanau et al. (2016)[22] introdujeron este concepto al cuestionar la manera en que los modelos de traducción automática de la familia codificador-decodificador codificaban una secuencia de entrada en un vector de longitud fija. En este contexto, el decodificador generaba una traducción, lo que insinuaba un posible cuello de botella en el rendimiento de estos modelos. Propusieron, por lo tanto, un modelo que automáticamente busca partes relevantes de una secuencia de entrada para predecir una palabra. El uso de estos módulos puede impactar en diferentes niveles de procesamiento y puede contribuir al desempeño en un problema de traducción automática. Soportado en este concepto, Ashish Vaswani et al. (2017)[23] propusieron la arquitectura *Transformer*

---

[21] Ashish Vaswani et al. *Attention Is All You Need*. 2017. arXiv: 1706.03762 [cs.CL].

[22] Bahdanau, Cho y Bengio, *Neural Machine Translation by Jointly Learning to Align and Translate*.

[23] Vaswani et al., *Attention Is All You Need*.

basada enteramente en mecanismos de atención, prescindiendo en su totalidad de redes neuronales convolucionales y recurrentes. La premisa del artículo es que la atención es el único bloque necesario a la hora de construir modelos de aprendizaje profundo para tareas de procesamiento del lenguaje natural (NLP) bajo arquitecturas codificador-decodificador. A continuación, se detallan los componentes de una arquitectura *Transformer*.

1.3.1. Embebidos Posicionales Los *Transformer* procesan la información de forma no secuencial (procesamiento en paralelo). Este enfoque, en principio, carece de información sobre la posición de las palabras (o unidades de la secuencia), la cual es importante para crear un sentido lingüístico. Sin embargo, se han propuesto diferentes estrategias para calcular esta información mediante frecuencias de onda de diferentes fases, formando así los embebidos posicionales.

$$PE_(pos, 2i) = sin(pos/10000^{2i/d_{embebido}})$$
$$PE_(pos, 2i+1) = cos(pos/10000^{2i/d_{embebido}})$$

Donde *pos* corresponde a la posición de la palabra dentro de la secuencia, *i* a los índices de cada una de las dimensiones del embebido posicional y $d_{embebido}$ a la dimensión del embebido (valor fijo). El uso del seno y el coseno es dado en función de la dimensión del embebido posicional; esto es, para $i = 0$ se usará el seno, para $i = 1$ el coseno y así sucesivamente. Estos descriptores posicionales pretenden no interferir en la información contenida en cada posición.

1.3.2. Auto-atención y Atención de Múltiples Cabezas (Self Attention - Multi-Head Attention) Los componentes de procesamiento de autoatención (Self-Attention, por su denominación en inglés), son la parte central de las arquitecturas *Transformer*, permitiendo resaltar las principales correlaciones dispuestas no solo

en vecindarios locales, sino también a largo plazo de las secuencias codificadas de entrada. Los mecanismos SA se definen de la siguiente manera:

$$SA(Q, K, V) = softmax(QK^T/\sqrt{d_k})V$$

donde se calcula el producto punto entre Q (*Query*) y K (*Key*) con el fin de identificar la similitud entre ambas matrices, esto a su vez dividido por la raíz cuadrada de la dimensión del vector k (con propósitos de escalado). Una vez realizadas estas operaciones, la salida conforma una matriz de similitud con los "puntajes de atención" los cuales pueden requerir un proceso de enmascaramiento dependiendo del módulo de atención. Luego, se aplica una *softmax* para representar la similitud en términos de probabilidad de correlación conjunta en los valores construidos. El resultado obtenido corresponde a un "filtro de atención", el cual al multiplicarse por la matriz V (*Value*) produce una matriz de "valor filtrado" que asigna alta prioridad a características que son más importantes dentro de un determinado contexto.

En esta misma línea, se pueden definir módulos especiales de atención múltiple, denominados módulos de atención con múltiples cabeceras (Multi-Head Attention, por su denominación en inglés). Estos módulos codifican múltiples mecanismos de atención de forma simultánea, donde cada unidad recibirá una porción de todo el embebido y se enfocará en identificar distintas combinaciones de características lingüísticas, concatenando al final todas las salidas. Por lo anterior, el número de cabeceras deberá corresponder a un número que permita la división entera de la dimensión del embebido. Por definición, estos módulos MHA corresponden a:

$$MultiHead(Q, K, V) = \text{Concatenación}(head_1, ..., head_h)$$
$$\text{donde } head_i = SA(QW_i, KW_i, VW_i)$$

siendo, Q, K y V (*Query*, *Key* y *Value* respectivamente) matrices que corresponden a tres copias exactas del embebido, las cuales inicialmente pasan por una capa

lineal (cada una con su propio conjunto de pesos) y posteriormente a los mecanismos SA. Una vez son concatenadas las salidas de cada cabecera, estas pasan nuevamente a una capa lineal conformando la salida final del módulo MHA. Debido al procesamiento holístico de los módulos de atención, donde toda la información correspondiente a las entradas (x) y las etiquetas (y) es embebida al mismo tiempo, es preciso poder codificar las etiquetas o traducciones correctas en función de la secuencia generada hasta el momento. Esta codificación es procesada como la proyección a un espacio embebido oculto, que corresponde a "enmascarar" la información que aún no ha sido generada por el decodificador. De esta manera, el modelo presta atención solamente a lo que ha generado hasta el momento para continuar con la traducción, evitando así un aprendizaje incorrecto o sobreentrenamiento. Este mecanismo de "enmascarar" partes de una secuencia se conoce como módulos "*Masked Multi Head Attention*" (MMHA)[24]. En esencia, los módulos MMHA tienen el mismo comportamiento de los módulos MHA, con la diferencia de que en el mecanismo SA, existe una capa de enmascaramiento después del proceso de escalado. Es decir, una vez se tienen los puntajes de atención y antes de pasar a la capa *softmax* se realiza la operación de enmascaramiento. Esta operación se expresa como una matriz filtro (filtro de enmascaramiento) que se aplica sobre el filtro de atención, donde todas las palabras futuras son procesadas con un puntaje de $-Inf$. En la capa softmax, todos los puntajes procesados con $-Inf$ pasan a ser ceros; de esta manera, el modelo sólo presta atención a las palabras previamente generadas.

---

[24] Vaswani et al., *Attention Is All You Need*.

1.4. Modelos propuestos en la tarea de traducción continua de la lengua de señas

Dentro de los enfoques propuestos en el estado del arte para la traducción de la lengua de señas, es común identificar dos componentes principales: el reconocimiento de las señas (*Sign Language Recognition*) y el posterior módulo de traducción al lenguaje hablado (*Sign Language Translation*). Recientemente, se han introducido enfoques computacionales para mejorar el reconocimiento continuo de instancias de lengua de señas. Hao Zhou *et al*., 2020[25], proponen una arquitectura para abordar la tarea de Reconocimiento Continuo de Lenguaje de Señas (SLR, por sus siglas en inglés). La idea principal de este enfoque es identificar la relevancia de los parámetros involucrados en la producción de señas mediante un proceso de extracción de características utilizando una red neuronal convolucional (CNN) como estructura central. Luego, se lleva a cabo un modelado temporal seguido de un proceso de "aprendizaje de secuencias" con el fin de obtener una secuencia de *glosas* [26] a partir de un video de lengua de señas. Sin embargo, este enfoque se centra únicamente en el reconocimiento de los gestos, dejando de lado la tarea de traducción.
Kayo Yin *et al*., 2020[27], exploran el mapeo entre la secuencia de glosas previamente identificadas por Hao Zhou *et al*.[28] y su correspondencia en lenguaje hablado como una tarea de traducción texto a texto. Se utiliza una arquitectura basada en *Trans-*

---

[25] Hao Zhou et al. "Spatial-Temporal Multi-Cue Network for Sign Language Recognition and Translation". En: *IEEE Transactions on Multimedia* 24.4 (2022), págs. 768-779. DOI: 10.1109/TMM.2021.3059098.

[26] Las glosas son palabras en el lenguaje hablado que tienen correspondencia con las señas

[27] Kayo Yin y Jesse Read. *Better Sign Language Translation with STMC-Transformer*. 2020. arXiv: 2004.00588 [cs.CL].

[28] Zhou et al., "Spatial-Temporal Multi-Cue Network for Sign Language Recognition and Translation".

formers[29] para codificar una representación adecuada de las glosas, que luego se decodifica aprovechando los mecanismos de autoatención de los *Transformers* para identificar las correspondencias temporales entre ambas secuencias. Sin embargo, este enfoque depende en gran medida de la calidad del proceso de reconocimiento de las glosas, lo que implica modularizar la tarea de traducción y no aprovechar completamente el potencial de las arquitecturas codificador-decodificador para realizar la tarea de manera conjunta (end-to-end).

Por otro lado, Rodriguez Jefferson *et al.*, 2021[30], presentan una arquitectura codificador-decodificador para abordar la traducción de la lengua de señas con un enfoque extremo a extremo, al mismo tiempo que se explora el flujo óptico como una representación más adecuada de las características espaciales de las señas. Las representaciones de flujo óptico permiten resaltar patrones cinemáticos espaciales que luego son procesados por unidades compuestas por redes neuronales recurrentes bidireccionales (BRNN) en conjunto con módulos de atención para aprovechar mejor las relaciones temporales complejas en los descriptores del video. No obstante, este enfoque se basa principalmente en redes neuronales recurrentes, las cuales actualmente se han visto opacadas con la introducción de las redes *Transformers*, que han demostrado ser más eficientes en el procesamiento de la información y más poderosas para identificar relaciones contextuales complejas.

En la misma línea, Camgoz Necati *et al.*, 2020[31], introducen una red *Transformer* para realizar la traducción de la lengua de señas, al mismo tiempo que se introduce un sistema de reconocimiento continuo dentro del codificador a través de una Clasi-

---

[29] Vaswani et al., *Attention Is All You Need*.

[30] Jefferson Rodriguez y Fabio Martínez. “How important is motion in sign language translation?” En: *IET Computer Vision* 15.3 (2021), págs. 224-234.

[31] Camgoz et al., *Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation*.

ficación Temporal Conexionista (CTC). Aunque este enfoque representa un avance significativo en la introducción de sistemas de traducción continua de la lengua de señas en escenarios de la vida real, este trabajo, al igual que la mayoría de los enfoques mencionados anteriormente, genera una representación unidimensional de las imágenes que puede carecer de información espacial relevante asociada a la reducción de dimensionalidad realizada durante el proceso de extracción de características.

2. Problema de Investigación

La lengua de señas, al igual que cualquier otra forma de comunicación, tiene componentes intrínsecamente variables debido a diversos factores sociales, culturales y geográficos. Esto genera una riqueza lingüística que plantea desafíos para su estandarización y modelamiento. Además, la representación gesto-visual de la lengua de señas es compleja, ya que involucra múltiples canales de información asincrónica conocidos como articuladores. Dichos articuladores comprenden aspectos como la expresión facial, la posición corporal, la velocidad de ejecución de las señas, así como el movimiento y la posición de las manos y dedos, entre otros.

Actualmente, se utilizan unidades lingüísticas de representación intermedia conocidas como “glosas“ para establecer una correspondencia entre la lengua de señas y el texto, que difiere de la traducción al lenguaje hablado. Sin embargo, estas representaciones intermedias suelen restringir la información densa y rica en términos gramaticales de los múltiples articuladores, lo que implica la pérdida de detalles locales y contextos complementarios que podrían ser clave para los modelos computacionales. Por lo tanto, la inclusión estricta de estas representaciones intermedias es actualmente tema de debate.

Los modelos propuestos para el reconocimiento de la lengua de señas (SLR), en gran parte abordan la tarea como un reconocimiento aislado de los gestos, asumiendo una relación entre las secuencias. Por otra parte, los mecanismos basados en recurrencia sólo adoptan ventanas de tiempo limitadas para definir correspondencias del lenguaje. Además, debido a la alta variabilidad en la ejecución de las señas y en el significado de las expresiones, la lengua de señas se considera un canal de comunicación altamente variable, incluso a nivel local (geográficamente hablando). Esto plantea desafíos para los modelos recurrentes de bajo orden, que pueden tener dificultades para capturar la coherencia semántica de las expresiones,

la cual está determinada en la frase completa. En la literatura existen herramientas de aprendizaje profundo que exploran relaciones no lineales entre secuencias para explorar la correspondencia entre la lengua de señas y el texto. Sin embargo, estas metodologías deben adaptarse y analizarse en el contexto de la traducción de la lengua de señas y su correspondencia con el texto.

3. Objetivos

Objetivo general

Desarrollar una arquitectura de aprendizaje profundo de tipo *Transformer* para la traducción de la lengua de señas, ponderando representaciones de atención.

Objetivos específicos

- Seleccionar un conjunto de datos debidamente etiquetado con las traducciones de la lengua de señas.
- Definir los principales mecanismos computacionales de atención en el problema de reconocimiento de gestos.
- Implementar una arquitectura *Transformer* para la traducción continua de la lengua de señas.
- Validar la arquitectura *Transformer* en la tarea de traducción de señas en secuencias de vídeo a texto.

4. Método Propuesto

En este trabajo se desarrolló e implementó un *Transformer* para la codificación y caracterización de gestos en el contexto de la traducción de videos en lengua de señas. Para esto, el enfoque propuesto recibe como entrada secuencias de video de lengua de señas en una representación de flujo óptico (Sección 4.1). Luego, se convolucionan los fotogramas para obtener mapas de características que preserven sus dimensiones espaciales (Sección 4.2). Posteriormente, codificadores posicionales bidimensionales (positional encodings 2D) aprovechan las composiciones geométricas y estructurales de cada mapa de características (Sección 4.3). A continuación, se calcula un mecanismo de auto-atención bidimensional (Sección 4.4). Como complemento, se utiliza un decodificador para relacionar las salidas del codificador con las secuencias de texto mediante mecanismos de atención (Sección 4.5). El modelo propuesto tiene la capacidad de estimar la correspondencia en lenguaje hablado utilizando un enfoque auto-regresivo. El proceso general de implementación se ilustra en la Figura 1.

4.1. De RGB a una representación en flujo óptico

Las lenguas de señas son expresiones visuales complejas que incorporan patrones espacio-temporales intrínsecos, tales como el movimiento, la posición, la forma y orientación de las manos y dedos[32]. En otras palabras, la comunicación en estas lenguas se compone de dos elementos principales: la geometría y el movimiento[33].

---

[32] Bencie Woll, RL Sutton-Spence y Frances Elton. “Multilingualism: The global approach to sign languages”. En: *The sociolinguistics of sign languages*. Cambridge University Press, 2001, págs. 8-32.

[33] Perniss, Lu y Morgan, “The visual complexity of sign languages”.

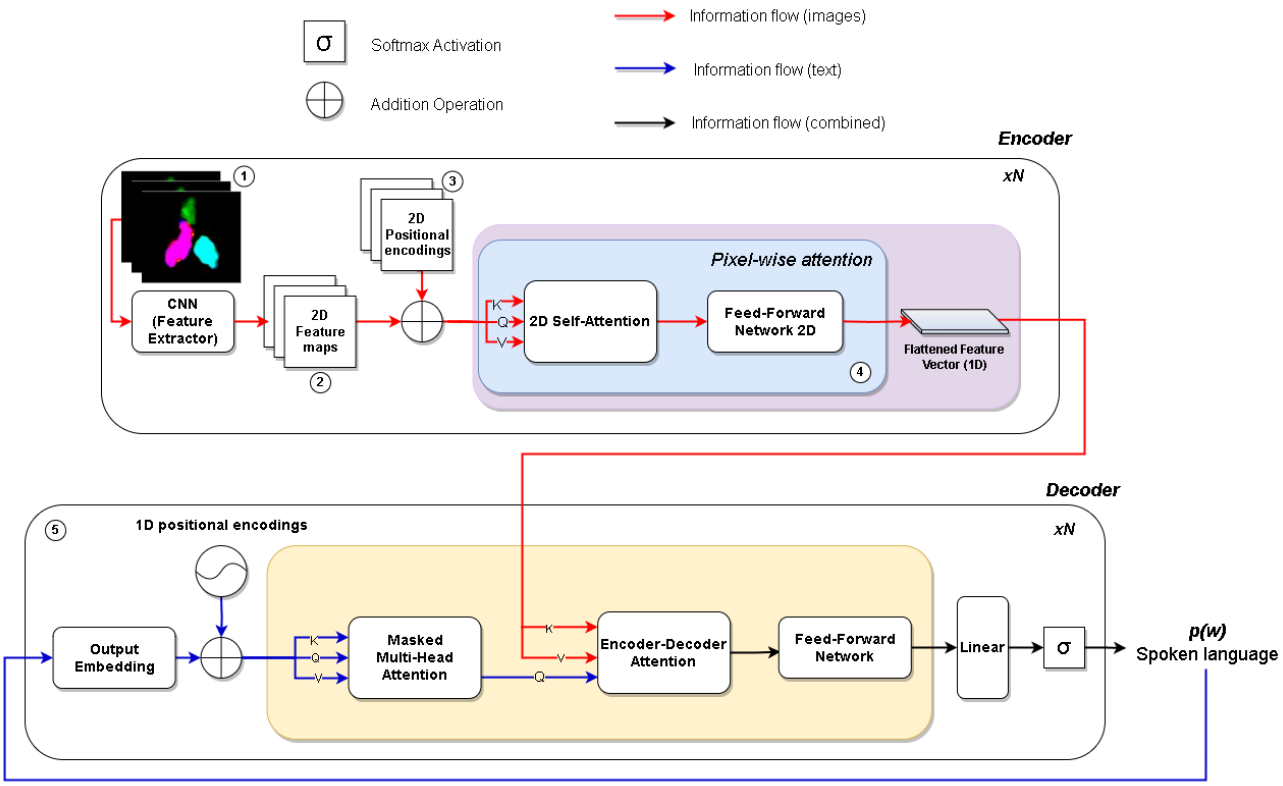

Figura 1. Enfoque propuesto. La arquitectura recibe una secuencia en flujo óptico, sobre la cual se realiza un proceso de extracción de características. Posteriormente, se añaden codificadores posicionales bidimensionales (positional encodings 2D). Se emplea un mecanismo de Auto-Atención 2D. El decodificador genera la traducción mediante proyecciones temporales entre los vídeos y el lenguaje escrito, siguiendo un enfoque auto-regresivo.

Por lo tanto, es de particular interés crear una representación de los vídeos que considere factores espacio-temporales, capturando no sólo los cambios de forma sino también las cinemáticas de las lenguas de señas. Para capturar la dinámica espacio-temporal de los videos de señas, se calculó un flujo óptico denso, que agrega información cinemática mientras preserva la forma de los articuladores durante la comunicación. Las secuencias de video V se representan como $V = \{v_1, v_2, \ldots, v_t\}$, donde $V \in \mathbb{N}^{T \times W \times H \times C}$. Aquí, T corresponde a la longitud temporal, $W \times H$ es la dimensión espacial de cada fotograma y C conforma el número de canales en cada

fotograma. El uso del flujo óptico en la traducción continua de la lengua de señas, como alternativa para transformar una secuencia de vídeo $V = \{v_1, v_2, \ldots, v_t\}$ en una representación cinemática $F = \{f_1, f_2, \ldots, f_t\}$, proporciona una ventaja significativa al capturar y explotar los patrones de movimiento presentes en los gestos. Esta representación corresponde a una alternativa visual menos densa, al eliminar información del fondo. Es por esto que el enfoque propuesto realiza un cambio en la representación de entrada, mapeando las imágenes en formato RGB a una representación de movimiento aparente de V en flujo óptico.

La transformación de videos a una representación espacio-temporal de movimiento ($V \rightarrow F$) se logra calculando un flujo óptico denso a lo largo de cada secuencia de fotogramas en RGB. En particular, en este trabajo se implementó el flujo óptico de Brox[34], que tiene la capacidad de manejar grandes desplazamientos entre los fotogramas de una secuencia de imágenes. Aquí, "grandes desplazamientos"se refiere a movimientos considerables de los objetos en la escena entre fotogramas consecutivos, siendo esta una característica fundamental para modelar adecuadamente la lengua de señas. El cálculo del flujo óptico se basa en un enfoque variacional. En la Figura 2 se muestra el resultado de cálcular una representación F utilizando el flujo óptico de Brox en una serie de fotogramas consecutivos (v_t, v_{t+1}).

4.2. Arquitectura Convolucional 2D como Extractor de Características

En este trabajo, se utilizó la arquitectura convolucional 2D *ResNet-18*[35] como extractor de características a partir de imágenes en flujo óptico (ver Figura 3). Esto

---

[34] Thomas Brox y Jitendra Malik. “Large displacement optical flow: descriptor matching in variational motion estimation”. En: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2009, págs. 41-48.

[35] Kaiming He et al. “Deep residual learning for image recognition”. En: *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), págs. 770-778.

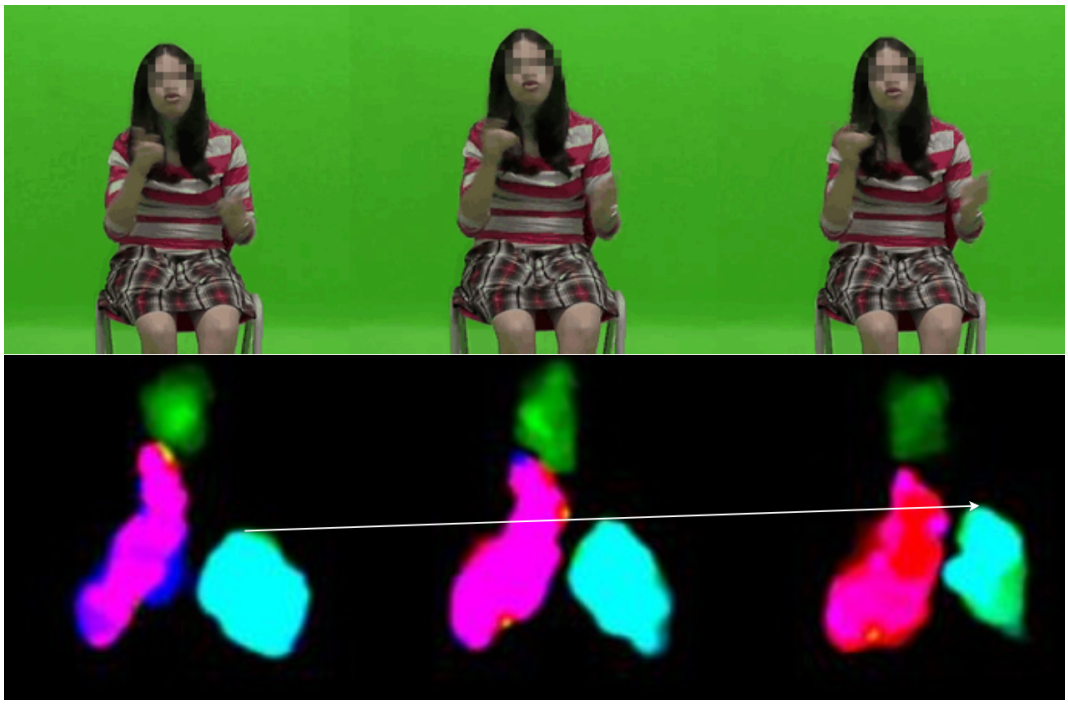

Figura 2. La representación en flujo óptico de Brox resalta patrones de movimiento primitivos relevantes y tiene la capacidad de manejar grandes desplazamientos entre los fotogramas, lo que resulta en un modelamiento más apropiado de la lengua de señas.

permitió obtener una representación que se basa en un conjunto de activaciones espacio-temporales que describen las formas de las manos, los movimientos y las expresiones faciales, elementos esenciales en la comunicación.

El enfoque distintivo de la arquitectura *ResNet-18* se basa en el uso de conexiones residuales, que permiten arquitecturas más profundas evitando el desvanecimiento del gradiente. Cada bloque residual de la arquitectura *ResNet-18* comprende capas de convolución seguidas de una función de activación ReLU, lo que introduce no linealidad en la red. Matemáticamente, para una capa específica i en un bloque residual, la salida se calcula mediante la ecuación:

$$\text{Salida}_i = \text{ReLU}(\text{BN}(\text{Conv}(\text{Salida}_{i-1}, \Theta_{\text{conv}})) + \text{Salida}_{i-1}), \quad (1)$$

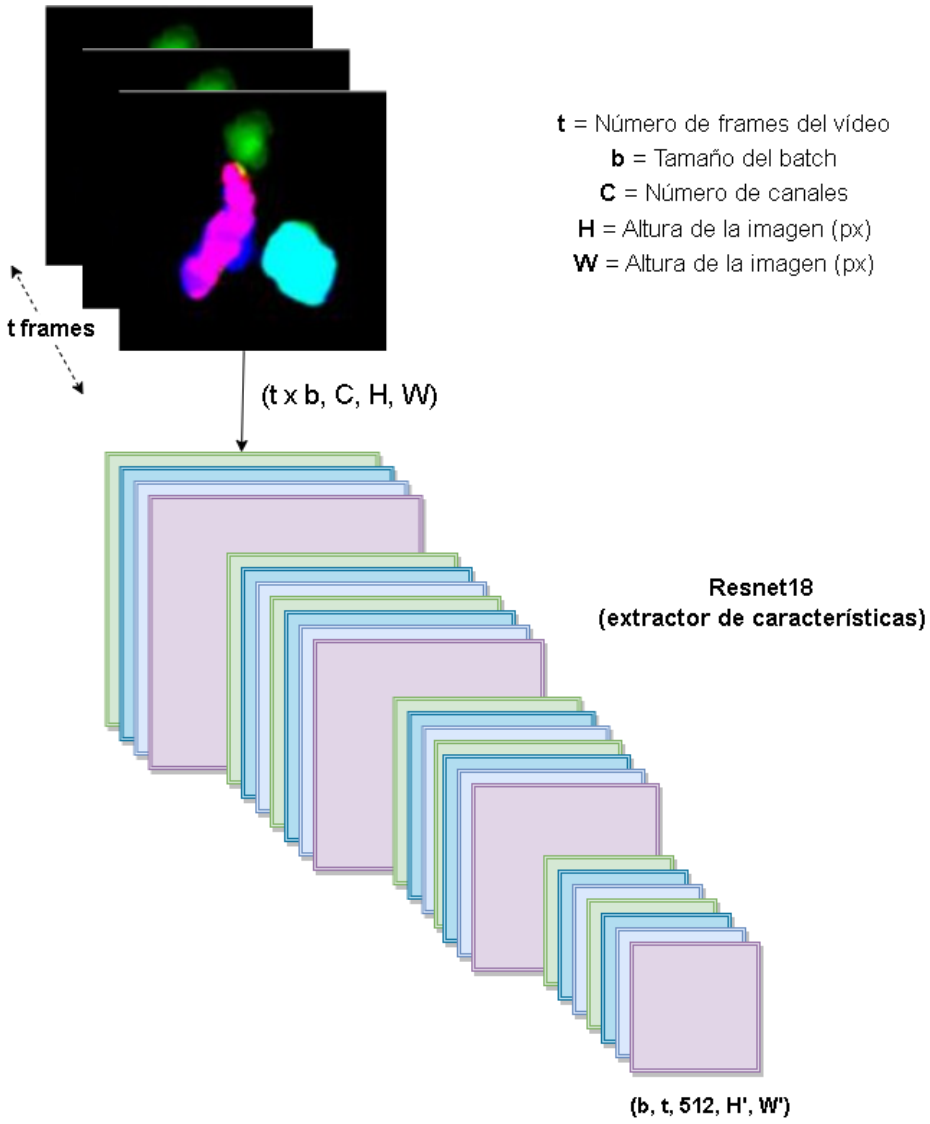

Figura 3. Proceso de extracción de características a partir de una arquitectura convolucional 2D.

donde Salida$_{i-1}$ es la salida de la capa anterior en el mismo bloque, Conv es la operación de convolución, BN es la normalización por lotes y Θ_{conv} son los parámetros de la capa convolucional. La integración de conexiones residuales en la arquitectura *ResNet-18* facilita la optimización y el aprendizaje de características discriminativas. El enfoque propuesto evita tomar las activaciones expresadas como un vector

de características aplanado debido a la pérdida significativa de información espacial local presente en las imágenes. En su lugar, el método propuesto selecciona las activaciones de un bloque convolucional anterior, donde dichas activaciones preservan sus dimensiones espaciales. Entonces, el flujo V es mapeado a la arquitectura convolucional, obteniendo una representación $X \in \mathbb{N}^{T \times W' \times H' \times D^{\mathsf{L}}}$, donde $W' \times H'$ son las dimensiones espaciales y D el conjunto de activaciones, calculado en la capa L.

4.3. Codificadores Posicionales Bidimensionales

Cuando se trata de modelar computacionalmente la lengua de señas, es importante que los modelos puedan aprender a reconocer patrones espaciales y las relaciones entre los elementos de una misma seña (articuladores). Sin embargo, las redes neuronales típicamente hacen proyecciones a vectores embebidos, perdiendo la relación espacial entre los píxeles de una imagen, lo que puede representar una limitación para el modelamiento estructural de las señas. Por esta razón, el enfoque propuesto utiliza codificaciones posicionales bidimensionales (positional encodings 2D)[36], las cuales se agregan a los mapas de características profundas X. Estas codificaciones constituyen una diferencia importante con las arquitecturas *Transformer* típicas, que usan codificaciones posicionales unidimensionales. Estas codificaciones posicionales 2D resultan ser más adecuadas para datos de entrada estructurados, como las imágenes, ya que permiten conservar la estructura de la imagen a nivel espacial, lo cual puede contener semántica crítica. Para esto, se añaden a cada píxel un par único de coordenadas (i, j), donde i representa el índice de fila y j representa el índice de columna. Las ecuaciones para las codificaciones posicionales 2D son las siguientes:

---

[36] Rui Xu et al. "Positional encoding as spatial inductive bias in gans". En: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, págs. 13569-13578.

$$\mathsf{PE}(x, y, 2i) = \sin\left(\frac{x}{10000^{4i/D}}\right)$$
$$\mathsf{PE}(x, y, 2i+1) = \cos\left(\frac{x}{10000^{4i/D}}\right)$$
$$\mathsf{PE}(x, y, 2j + \frac{D}{2}) = \sin\left(\frac{y}{10000^{4j/D}}\right)$$
$$\mathsf{PE}(x, y, 2j + 1 + \frac{D}{2}) = \cos\left(\frac{y}{10000^{4j/D}}\right)$$

Aquí, (x, y) es un punto en un espacio 2D, y i, j son enteros en $[0, D/4)$, donde D es el número de dimensiones del embebido. Estas codificaciones posicionales tienen el mismo tamaño y dimensión que los mapas de características extraídos previamente, y permiten codificar la posición relativa de cada píxel en relación con otros píxeles en la imagen. Entonces, a la representación X se le suma el vector posicional PE, obteniendo $X = X + PE$ una representación mejorada que incluye información posicional espacial.

4.4. Codificador

El codificador es una parte esencial de la arquitectura ya que permite la extracción y procesamiento de características significativas de las imágenes de lengua de señas (Ver Figura 4). El bloque codificador está compuesto por dos componentes clave: el mecanismo de auto-atención 2D y la red neuronal prealimentada, que trabajan en conjunto para capturar y transformar la información visual en representaciones útiles y compactas.

En este trabajo se implementó un módulo de auto-atención bidimensional[37]. Este módulo permite modelar de manera más eficiente las relaciones entre regiones es-

[37] Han Zhang et al. *Self-Attention Generative Adversarial Networks*. 2019. arXiv: 1805.08318 [stat.ML].

paciales ampliamente separadas, lo cual resulta crucial para distinguir gestos que involucran tanto articuladores manuales como no manuales. La Figura 4 muestra el módulo de auto-atención 2D utilizado en el enfoque propuesto.

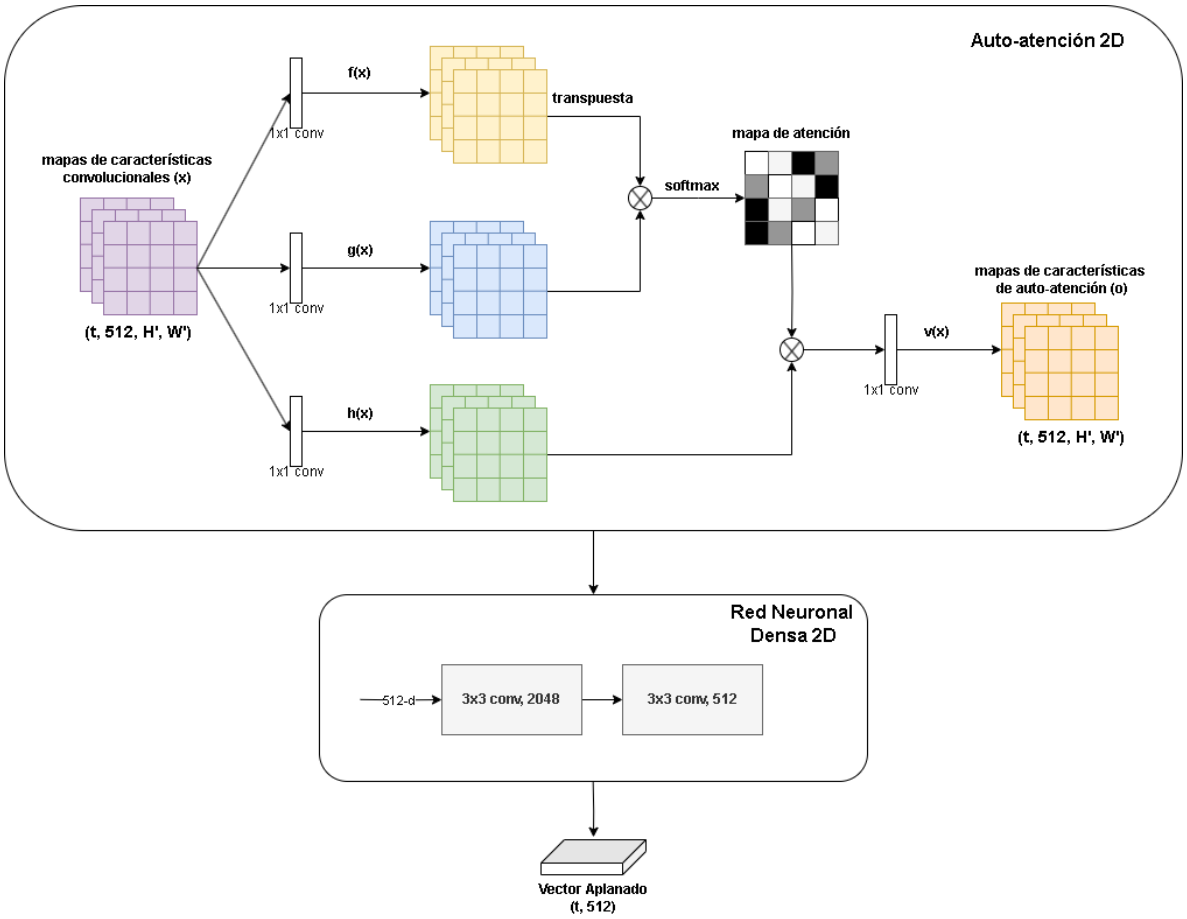

Figura 4. Codificador. Está compuesto por dos componentes clave: el mecanismo de auto-atención 2D[38] y la red neuronal prealimentada, que trabajan en conjunto para capturar y transformar la información visual en representaciones útiles y compactas.

Particularmente, dado un conjunto de mapas de características X con dimensiones espaciales $H \times W$ y C canales, se puede calcular el mapa de auto-atención A mediante el cálculo de tres conjuntos de mapas de características Q, K y V. Estos conjuntos se obtienen aplicando tres transformaciones lineales diferentes a los mapas de características originales. Cada conjunto tiene su propia matriz de pesos

aprendidos W_Q, W_K y W_V y se calcula de la siguiente manera:

$$Q = W_Q \cdot X; \qquad K = W_K \cdot X; \qquad V = W_V \cdot X;$$

A continuación, se calcula el mapa de atención A mediante el cálculo de la similitud entre los conjuntos Q y K, donde $A = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)$.
Aquí, d_k es la dimensión de los mapas de características, y softmax es una función que normaliza los pesos de atención calculados, produciendo una matriz de probabilidades normalizadas. El mapa de atención A es una matriz con dimensiones $H \times W \times H \times W$, y representa la importancia de cada posición espacial en los mapas de características.
Finalmente, se utliza el mapa de atención A para calcular una suma ponderada de los mapas de características V, tal que $O = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V$. Aquí, O es la salida del mecanismo de auto-atención, y representa una combinación ponderada de los mapas de características originales, en la cual se han aplicado los pesos de atención determinados por el mapa de atención A. El mecanismo de auto-atención 2D permite que la red se enfoque en las partes más importantes de los mapas de características de entrada, lo que resulta en una mejor identificación de los componentes clave de las señas al aprender relaciones de dependencia espacial entre diferentes regiones de una imagen.
Una vez que la representación de entrada ha sido codificada a través del mecanismo de auto-atención 2D, esta sirve de entrada a una variante adaptada de la red neuronal prealimentada (feed-forward network) presente en el *Transformer* estándar. Esta versión modificada de la red consta de dos capas convolucionales, lo que permite al modelo capturar representaciones más complejas y abordar relaciones espaciales locales de manera efectiva. Adicionalmente, se realiza un mapeo de dimensionalidad, temporalmente incrementando la capacidad de la red con el propó-

sito de capturar características más enriquecedoras en los datos. Para potenciar la estabilidad y el rendimiento, se implementa la normalización por lotes (batch normalization) en las salidas de cada capa. Esta técnica evita que las salidas se desvíen significativamente de la distribución de entrada y puede prevenir problemas como el desvanecimiento del gradiente durante el proceso de entrenamiento. Finalmente, la salida se transforma en un vector de características aplanado mediante una operación de *Average Pooling*, con el fin de crear una representación más compacta de la secuencia de entrada. Esta representación compacta resulta más óptima para su posterior procesamiento en el decodificador.

4.5. Decodificador

El decodificador toma la salida del codificador, una secuencia de estados ocultos y genera una secuencia de salida en lenguaje hablado, siguiendo un enfoque autoregresivo basado en la probabilidad condicional $P(W_t|V, W_{t-1})$. El decodificador está compuesto por tres componentes principales: la capa de auto-atención enmascarada de múltiples cabezas, la capa de auto-atención de múltiples cabezas y la capa neuronal densa. A continuación, se detallan cada uno de los módulos que componen el decodificador *Transformer*.

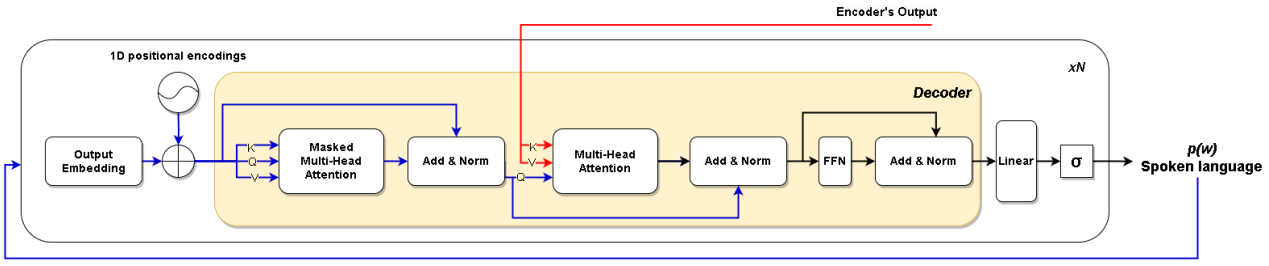

Figura 5. Decodificador del modelo propuesto basado en la arquitectura *Transformer* estándar[39]

.

4.5.1. Capa Word Embedding La capa de *word embedding* en el decodificador del Transformer juega un papel crucial en la generación del texto objetivo. Su objetivo es mapear las palabras del vocabulario a un espacio continuo y denso, donde las relaciones semánticas y sintácticas entre las palabras se reflejan de manera eficiente. Esta proyección de palabras permite que el modelo capture las relaciones, correlacionando las palabras durante la generación del texto. Específicamente, dado un vocabulario de tamaño V y una secuencia de palabras objetivo $Y = (y_1, y_2, \ldots, y_T)$, donde y_t es el índice de la palabra en el vocabulario en el paso de tiempo t, la capa de *word embedding* transforma cada palabra en un vector denso $e_t \in \mathbb{R}^d$, donde d es la dimensión del espacio de *embedding*: $e_t = \text{Embed}(y_t)$. Donde Embed es la función a vectores embebidos que proyecta el índice de la palabra en el espacio de *embedding*: $\text{Embed}(y_t) = \mathbf{E}_{y_t}$. Aquí, $\mathbf{E} \in \mathbb{R}^{V \times d}$ es la matriz de vectores embebidos, donde cada fila corresponde al vector de *embedding* de una palabra en el vocabulario. Cada palabra se representa como un vector en el espacio embebido, lo que permite que el modelo capture las similitudes semánticas entre las palabras y genere un texto coherente y significativo en la tarea de traducción.

4.5.2. Módulos de Auto-Atención Este componente captura relaciones de dependencia entre las palabras en las secuencias de entrada y salida. Estos módulos permiten que el modelo procese cada palabra teniendo en cuenta las relaciones con otras palabras en la misma secuencia, lo que resulta en una representación contextualizada y rica de las palabras.

Específicamente, al considerar un conjunto de secuencias de entrada $X = (x_1, x_2, \ldots, x_T)$, donde x_t es el vector de características en el paso de tiempo t, el módulo de auto-atención calcula una nueva representación contextual z_t para cada palabra: $z_t = \sum_{j=1}^{T} \alpha_{tj} x_j$. Donde α_{tj} es el peso de atención entre las palabras x_t y x_j, y se calcula como: $\alpha_{tj} = \frac{\exp(e_{tj})}{\sum_{k=1}^{T} \exp(e_{tk})}$, y donde $e_{tj} = \text{Score}(x_t, x_j)$ calcula la afinidad

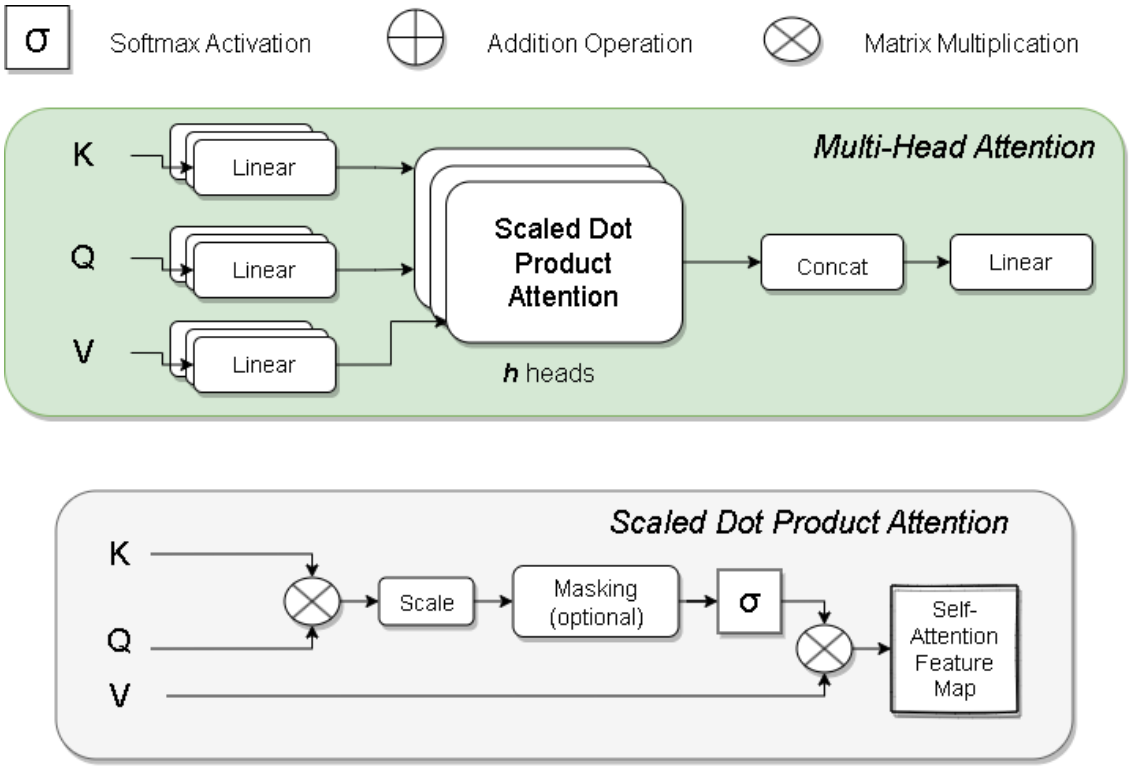

Figura 6. Módulo de Auto-Atención empleado en el Decodificador.

entre las palabras x_t y x_j. La regla de afinidad puede ser implementada utilizando productos internos, redes neuronales u otras métricas de similitud.

Como proceso adicional, se utiliza la auto-atención enmascarada para asegurar que las palabras futuras no influyan en las palabras actuales durante la generación del texto objetivo. Esto se logra aplicando una máscara triangular superior a los pesos de atención:

$$\alpha_{tj} = \begin{cases} \frac{\exp(e_{tj})}{\sum_{k=1}^{T} \exp(e_{tk})}, & \text{si } j \leq t \\ 0, & \text{si } j > t \end{cases}$$

La auto-atención enmascarada garantiza que cada palabra solo tenga acceso a información de palabras previamente generadas en la secuencia de salida, evitando

así el acceso a la información futura. En la Figura 6 se ilustra el funcionamiento en general de estos módulos de atención utilizados en el decodificador.

4.5.3. Red Neuronal Densa Esta red se utiliza para procesar y transformar las representaciones contextuales, capturando relaciones más complejas, lo que contribuye significativamente a su capacidad de generación y traducción. Dada una representación contextual z_t obtenida del módulo de auto-atención, la red neuronal prealimentada realiza una serie de transformaciones lineales y no lineales para generar una nueva representación procesada $h_t = \text{FFN}(z_t)$. Donde FFN es una función que consta de dos capas totalmente conectadas (capa lineal seguida de una función de activación no lineal) y se calcula como: $a_t = \text{ReLU}(W_1 z_t + b_1)$ y posteriormente $h_t = W_2 a_t + b_2$. Aquí, W_1, W_2, b_1, b_2 son los pesos y sesgos de las capas lineales, y ReLU es la función de activación Rectified Linear Unit, que introduce la no linealidad en la red.

5. CONFIGURACIÓN EXPERIMENTAL

Para validar el modelo propuesto, utilizamos videos de lengua de señas representados como una secuencia de fotogramas en RGB y los convertimos en representaciones de flujo óptico. Cada fotograma fue procesado utilizando una arquitectura *ResNet-18* pre-entrenada en el conjunto de imágenes naturales ImageNet. Las imágenes de entrada se redimensionaron a una resolución de 224×224 píxeles. Durante el proceso de extracción de características, se tomaron las activaciones correspondientes a la capa convolucional 5, que tiene dimensiones de $[longitudSecuencia \times dimensiónEmbebido \times 7 \times 7]$. A estas características, agregamos codificaciones posicionales bidimensionales. Para la red neuronal densa, aplicamos dos capas convolucionales con un tamaño de kernel de 3×3 y utilizamos la normalización por lotes. Durante el entrenamiento, utilizamos anotaciones de glosas en el codificador con una función de pérdida conocida como *Connectionist Temporal Classification* (CTC) para mapear cada representación de fotograma a una glosa, alineando así las glosas repetidas en el tiempo. El decodificador emplea cuatro cabezales para los mecanismos de atención de múltiples cabezas. El *Transformer* consta de una sola capa. El modelo fue entrenado durante 10 épocas en el conjunto de datos de lengua de señas Colombiano (LSCD) y durante 20 épocas en el conjunto de datos Alemán (RWTH-PHOENIX-2014T). Utilizamos un tamaño de lote de 1 vídeo, el optimizador Adam y una función de pérdida de entropía cruzada (*Cross Entropy*). El entrenamiento se realizó de manera *end-to-end*. En la tabla 1 se resume la configuración de la metodología implementada.

Tabla 1. Detalles del Entrenamiento y Configuración del Modelo

Configuración	**Valor**
Representación de Entrada	RGB y Flujo Óptico
Modelo de CNN	ResNet18 pre-entrenada en ImageNet
Resolución de Imagen de Entrada	224×224 píxeles
Capa Convolucional de Salida	5
Dimensiones de las Activaciones	$[longitudSecuencia \times dimensiónEmbebido \times 7 \times 7]$
Codificaciones Posicionales Codificador	Bidimensionales (2D)
Tipo de mecanismo de Atención Codificador	Auto-Atención 2D
Parámetro de Mecanismo de Auto-atención 2D	*Gamma* (aprendible)
Conexiones de Atajo (skip connections)	Sí
Red Neuronal Densa Codificador	2 capas con kernel 3×3 y normalización por lotes
Función de Pérdida Codificador	*Connectionist Temporal Classification* (CTC)
Codificaciones Posicionales Decodificador	Unidimensionales (1D)
Tipo de mecanismo de Atención Decodificador	Auto-Atención y Auto-Atención Enmascarada
Número de Cabezas en Mecanismos de Atención Decodificador	4 cabezas
Número de Capas del *Transformer*	1 capa
Épocas de Entrenamiento en col-LSCD	10 épocas
Épocas de Entrenamiento en RWTH-PHOENIX-2014T	20 épocas
Tamaño de Lote	1 vídeo para ambos conjuntos de datos
Optimizador	Adam
Función de Pérdida	Entropía Cruzada (*Cross Entropy*)
Tipo de Entrenamiento	Extremo a extremo (*End-to-End*)

6. EVALUACIÓN Y RESULTADOS

Para validar el método propuesto, primero se evaluó la capacidad de la representación profunda para modelar entradas que incluyen información de movimiento, así como la contribución de las variables cinemáticas en la representación de los gestos durante la tarea de traducción de la lengua de señas. La tabla 2 resume los resultados obtenidos por el método propuesto, utilizando secuencias de entrada crudas en RGB y secuencias de movimiento aparente calculadas a partir del flujo óptico de largos desplazamientos. Como se puede observar, los resultados expuestos en la tabla 2 indican que la información cinemática contribuye significativamente a la descripción y caracterización de los gestos. De hecho, la representación con flujo óptico obtuvo una mejora del 17.04 % en BLEU1 y un aumento del 20.6 % en BLEU4. Esto demuestra que el uso de información de movimiento, junto con la información estructural de la seña, conduce a mejoras tanto en las predicciones de intervalos cortos (BLEU1) como en las codificaciones a largo plazo (BLEU4).

	RGB	Flujo óptico
BLEU1	0.4858	**0.6562**
BLEU2	0.4191	**0.5991**
BLEU3	0.3425	**0.5444**
BLEU4	0.3077	**0.5137**

Tabla 2. Comparación del enfoque propuesto en imágenes RGB contra imágenes en flujo óptico

Una de las principales contribuciones de este trabajo fue la inclusión de codificadores posicionales en 2D (*positional encodings 2D*) que puedan ayudar a mantener la informacion de espacialidad de la seña. En este caso, estos codificadores posicionales pueden contribuir a mantener la estructura de la seña mientras expresan una frase, registrado en un vídeo. En caso contrario, los codificadores posicionales

desde vectores embebidos pierden esta información estructural de la seña, lo cual puede representar una desventaja en la traducción de señas. La Figura 7 ilustra un análisis comparativo entre la representación propuesta, que utiliza codificaciones espaciales, y un *Transformer* estándar que emplea codificaciones posicionales unidimensionales.

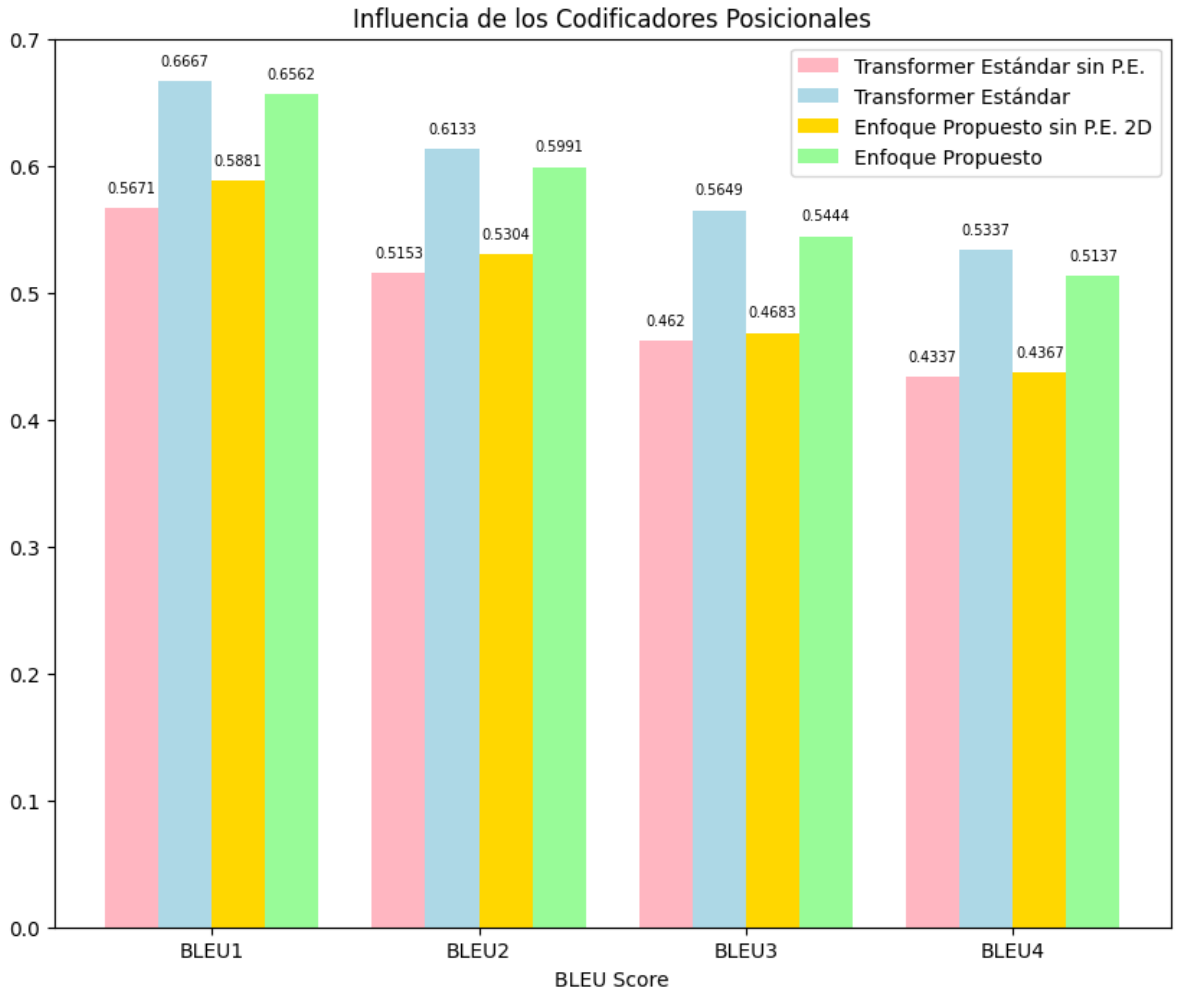

Figura 7. Influencia de los Codificadores Posicionales en la arquitectura *Transformer* estándar vs Enfoque Propuesto.

A partir de esto, se encontró que el desempeño del *Transformer* estándar depende considerablemente de las codificaciones posicionales unidimensionales (P.E.) mos-

trando un decrecimiento en su rendimiento de un 9.96 % BLEU1 y un 10 % BLEU4 al ser eliminadas. Estas codificaciones posiciones unidimensionales embeben la información a nivel de orden de los *tokens* dentro de la secuencia, ya que el *Transformer* por si sólo, debido a su procesamiento en paralelo, es invariante a las permutaciones. Lo anterior resulta en una perdida de contextos temporales que dan sentido al lenguaje. Aunque estos contextos temporales son esenciales para el procesamiento del lenguaje natural, depender únicamente de ellos limita la capacidad del modelo para incorporar de manera efectiva información espacial en el contexto del reconocimiento de gestos. Por otra parte, el enfoque propuesto experimenta una disminución del 6.81 % en BLEU1 y del 7.7 % en BLEU4 al eliminar las codificaciones posicionales bidimensionales (P.E. 2D), lo que sugiere que, si bien este componente es importante para la arquitectura, estas codificaciones funcionan principalmente como un complemento de los módulos orientados a la explotación de la información espacial. De hecho, el enfoque propuesto no hace uso de codificaciones posicionales unidimensionales, es decir, no embebe información relacionada con el orden dentro de la secuencia, centrándose únicamente en la información posicional espacial presente en los mapas de características. Además, se puede observar que, a pesar de la clara disminución en el rendimiento al eliminar ambas codificaciones posicionales, en estas condiciones, el enfoque propuesto supera a la arquitectura estándar en todas las métricas reportadas.

Para complementar este análisis, en este trabajo también se estudió cada uno de los componentes de la arquitectura para estimar la contribución de cada módulo con el fin de identificar estrategias de configuración que respalden la traducción de la lengua de señas. Para ello, el estudio comparó la arquitectura con todos los componentes (EP), la arquitectura sin el módulo posicional 2D (EP SIN P.E 2D), la arquitectura sin el módulo de auto-atención 2D (EP SIN ATTN 2D), la arquitectura sin

la red neuronal densa 2D (EP SIN FFN 2D), la arquitectura sin el módulo de auto-atención 2D y la red neuronal densa 2D (EP SIN ATTN 2D, FFN 2D), y también, la arquitectura sin anotaciones textuales en glosas (EP SIN GLOSAS). La tabla 3 recopila los resultados obtenidos del análisis de componentes. Cabe resaltar que la mejor configuración resultante es la arquitectura propuesta que incluye todos los componentes, junto con anotaciones en glosas.

Método	BLEU1	BLEU2	BLEU3	BLEU4
E.P.	**0.6562**	**0.5991**	**0.5444**	**0.5137**
E.P. - SIN P.E 2D	0.5881	0.5304	0.4683	0.4367
E.P. - SIN ATTN 2D	0.6179	0.5556	0.4957	0.4624
E.P. - SIN FFN 2D	0.6155	0.5555	0.4938	0.4620
E.P. - SIN ATTN 2D, FFN 2D	0.4791	0.4072	0.3292	0.2952
E.P. - SIN GLOSAS	0.6048	0.5510	0.4969	0.4699

Tabla 3. Enfoque Propuesto - Estudio de análisis de componentes

De la tabla 3, se observa que el enfoque propuesto (EP) logra la puntuación BLEU4 más alta, con un 51.37 %. No obstante, la eliminación de las codificaciones posicionales bidimensionales (P.E 2D) provoca una disminución significativa en el rendimiento, reduciendo la puntuación BLEU4 a 43.67 %. Este resultado puede indicar la importancia de la codificación espacial en el enfoque propuesto. En contraste, la eliminación del mecanismo de auto-atención (ATTN 2D) o de la red neuronal densa (FFN 2D) solo ocasiona una leve disminución en el rendimiento. Sin embargo, de forma interesante, la exclusión de ambos módulos ATTN 2D y FFN 2D resulta en una disminución sustancial del rendimiento, con una reducción del 21.85 % en BLEU4. Además, la incorporación de glosas al enfoque propuesto mejora todas las métricas BLEU. Por lo tanto, este análisis sugiere que cada uno de los componentes principales del enfoque propuesto contribuye de manera significativa al rendimiento general, y la adición de glosas mejora notablemente el rendimiento.

Para complementar este análisis, a continuación, se presentan algunas de las sali-

das del modelo. En la Figura 4, se muestran algunas de las traducciones del lenguaje hablado obtenidas en comparación con sus correspondientes traducciones de referencia. Se puede observar cómo, aunque la red puede ser bastante precisa, todavía existen casos en los que no logra traducir correctamente la secuencia de entrada. Estos resultados sugieren que se pueden realizar mejoras adicionales en el modelo para aumentar su rendimiento general.

Oración real	**Oración predicha**
nosotros somos felices	nosotros somos cansones que compró una casa
carlos viaja a bogotá hoy	carlos viaja a bogotá hoy
juan comprará un carro en el futuro	juan no comprará una casa en el futuro
tu hermano tiene hambre	tu hermano tiene hambre
a juan le gusta el chocolate	a juan le gusta esto y eso
mary le cuenta a juan que compró una casa	mary le cuenta a juan que compró una casa

Tabla 4. Predicciones en el conjunto de datos de lengua de señas Colombiana (LCSD)

Además, como parte de un análisis cualitativo, se recopilaron algunas de las salidas visuales más relevantes que pueden ayudar a comprender mejor el comportamiento del modelo. En la Figura 8, se pueden apreciar los mapas de atención de las tres primeras cabezas en el mecanismo de auto-atención del decodificador del *Transformer*. A partir de esto, se puede observar cómo la red identifica diferentes relaciones entre cada una de las palabras de la oración, lo que permite que la red aprenda contextos relevantes del lenguaje. Adicionalmente, se puede notar que por encima de la diagonal superior, no se tiene en cuenta ninguna palabra; esto se debe al proceso de enmascaramiento que se realiza en relación a las palabras futuras, con el fin de prevenir un sobreentrenamiento del modelo.

El trabajo propuesto también se comparó con dos arquitecturas propuestas en el

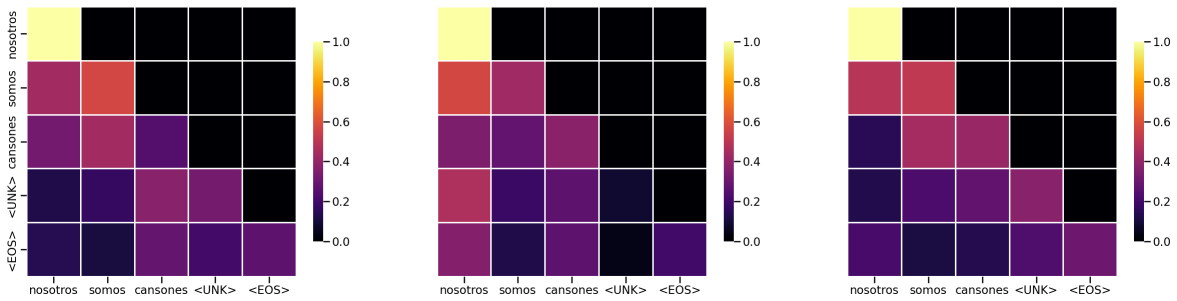

Figura 8. Mapas de atención generados por el mecanismo de auto-atención en el decodificador del *Transformer*. La secuencia de entrada $x_1, x_2, \ldots, x_n$ pasa a través de un mecanismo de auto-atención, el cual produce mapas de atención que ponderan cada elemento de la secuencia de entrada según su relevancia con respecto a los demás elementos.

estado del arte: una arquitectura con módulos recurrentes[40] y una arquitectura de tipo *Transformer* estándar adaptada para la traducción de señas. Este *Transformer* de la línea base sigue una configuración clásica ([41]), pero fue adaptado para la tarea de traducción de señas. De manera interesante, el método propuesto logra un BLEU4 de 51.37 %, mientras que el *Transformer* estándar logra un BLEU4 de 53.37 %. En un experimento extra se eliminó la información posicional de ambas arquitecturas obteniendo un BLEU4 de 43.67 % para el método propuesto y un BLEU4 de 43.37 % para el *Transformer* estándar. Esto nos permite deducir que la arquitectura estándar está limitándose al aprendizaje posicional a nivel de orden dentro de la secuencia de entrada, realizando un sobre-aprendizaje del conjunto de datos propuesto y siendo limitado su uso en esquemas generalizados para otros conjuntos de datos. Por otra parte, en la misma tarea, el modelo recurrente logró un BLEU4 de 58.69 % en su mejor configuración; sin embargo, el método propuesto obtiene resultados superiores en comparación con el resto de configuraciones reportadas, tanto

---

[40] Rodriguez y Martínez, “How important is motion in sign language translation?”

[41] Vaswani et al., *Attention Is All You Need*.

en flujo óptico como en imágenes en RGB. Esto sugiere que el modelo recurrente tiene marcadas limitaciones para realizar el trabajo de traducción. De hecho, en este caso, el método propuesto superó al método recurrente en un 11.21 % en el dominio de imágenes en RGB. Para validar la capacidad de generalización del modelo, se ajustó la mejor arquitectura para realizar la traducción en el conjunto de datos RWTH-PHOENIX-2014T. En este caso, el método propuesto obtuvo un BLEU4 del 15.24 %, demostrando un rendimiento competitivo en comparación con el estado del arte. Adicionalmente, el mejor trabajo del estado del arte en traducción de video a texto obtiene un BLEU4 del 21.32 %, mientras que la arquitectura recurrente logra un BLEU4 del 4.56 %.

7. CONCLUSIONES Y PERSPECTIVAS

En este trabajo se desarrolló una arquitectura de aprendizaje profundo de tipo *Transformer* para la traducción de la lengua de señas, ponderando representaciones de atención. Entre las principales contribuciones, se propuso un cambio en la representación de entrada de imágenes crudas en RGB a representaciones en flujo óptico de grandes desplazamientos. Esta representación espacio-temporal demostró ser más efectiva para recuperar la forma del gesto, además de lograr una caracterización de las cinemáticas de las señas modeladas. Además, se ajustó el extractor de características conformado por una arquitectura convolucional ResNet18 para tomar las activaciones correspondientes a una capa convolucional, en lugar del vector aplanado. En este sentido, la representación de entrada mantiene la información estructural de los gestos.

Adicionalmente, se introdujeron codificadores posicionales bidimensionales, siendo esta la primera arquitectura *Transformer* que mantiene la estructura de los gestos en la tarea de traducción continua de la lengua de señas. Como complemento, se utilizó un mecanismo de auto-atención 2D y una red neuronal densa que busca explotar aún más la información espacial presente en las señas.

El enfoque propuesto se evaluó principalmente utilizando el conjunto de datos de Lengua de Señas Colombiano (col-LSCD), en comparación con dos enfoques de línea base como referencia. Los resultados obtenidos fueron competitivos y, en condiciones específicas, incluso superaron a estos enfoques. De hecho, nuestro enfoque logró un BLEU4 del 30.77 % en RGB, superando las métricas previamente reportadas en el estado del arte. Además, alcanzó un BLEU4 del 51.37 % en imágenes en flujo óptico, obteniendo un puntaje que supera a la mayoría de las configuraciones previamente reportadas. Por otra parte, se validó la capacidad de generalización del modelo evaluándolo en el conjunto de datos RWTH-PHOENIX-2014T, donde obtu-

vo un BLEU4 del 15.24 %, un puntaje igualmente competitivo con respecto a las arquitecturas del estado del arte. Sin embargo, el enfoque propuesto presenta algunas limitaciones. Específicamente, es computacionalmente costoso, lo que conduce a tiempos de entrenamiento prolongados. Además, debido a las limitaciones de los conjuntos de datos disponibles, puede haber problemas de sobreentrenamiento. Como trabajo futuro, se busca explorar nuevos mecanismos de atención que capturen relaciones espacio-temporales de las señas sin perder la generalización de la representación. Además, se deben investigar estrategias de entrenamiento que permitan reducir el costo computacional de la arquitectura, especialmente al trabajar con conjuntos de datos más grandes. También se podrían explorar diferentes arquitecturas convolucionales, como convoluciones 3D, para mejorar la representación de entrada y capturar características espacio-temporales de manera más efectiva. El módulo de atención 2D podría mejorarse mediante la implementación de un enfoque de atención de múltiples cabezas, lo que permitiría que el modelo atienda diferentes partes de la entrada simultáneamente. Futuros trabajos también incluyen la evaluación del método propuesto en *corpus* mucho más grandes que incorporen desafíos adicionales típicamente observados en la lengua de señas.

BIBLIOGRAPHY

Bahdanau, Dzmitry, Kyunghyun Cho y Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. 2016. arXiv: 1409.0473 [cs.CL]. URL: https://arxiv.org/abs/1409.0473.

Bai, Shaojie, J. Zico Kolter y Vladlen Koltun. *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*. 2018. arXiv: 1803.01271 [cs.LG].

Brox, Thomas y Jitendra Malik. "Large displacement optical flow: descriptor matching in variational motion estimation". En: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2009, págs. 41-48.

Camgoz, Necati Cihan et al. "Neural machine translation for sign languages: A survey". En: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics. 2018, págs. 47-53.

Camgoz, Necati Cihan et al. *Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation*. 2020. arXiv: 2003.13830 [cs.CV].

Cho, Kyunghyun et al. "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation". En: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014.

Donahue, Jeff et al. *Long-term Recurrent Convolutional Networks for Visual Recognition and Description*. 2016. arXiv: 1411.4389 [cs.CV]. URL: https://arxiv.org/abs/1411.4389.

Elman, Jeffrey L. "Finding Structure in Time". En: *Cognitive Science* 14.2 (1990), págs. 179-211.

Gers, Felix A., Jürgen Schmidhuber y Fred Cummins. “Learning to Forget: Continual Prediction with LSTM”. En: *Neural Computation* 12.10 (1999), págs. 2451-2471. DOI: 10.1162/089976699300016629.

Graves, Alex. “Generating Sequences with Recurrent Neural Networks”. En: *arXiv preprint arXiv:1308.0850* (2014).

He, Kaiming et al. “Deep residual learning for image recognition”. En: *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), págs. 770-778.

Hochreiter, Sepp y Jürgen Schmidhuber. “Long Short-Term Memory”. En: *Neural Computation* 9.8 (1997), págs. 1735-1780.

Hu, Xiaoying et al. “Sign language translation: A deep learning-based approach”. En: *2020 IEEE International Conference on Signal and Image Processing (ICSIP)*. IEEE. 2020, págs. 131-136. DOI: 10.1109/ICSIP50750.2020.9376157.

Kalchbrenner, Nal, Edward Grefenstette y Phil Blunsom. “Recurrent Continuous Translation Models”. En: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2013.

Liu, Huan et al. “Sign language recognition and translation with recurrent neural networks and visual attention”. En: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2018, págs. 7266-7273. DOI: 10.1109/ICRA.2018.8460515.

Luong, Minh-Thang, Hieu Pham y Christopher D. Manning. “Effective Approaches to Attention-based Neural Machine Translation”. En: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2015, págs. 1412-1421.

Organization, World Health. *Estimates of Hearing Loss 2021*. Jul. de 2018. URL: https://www.who.int/deafness/estimates/en/ (visitado 26-09-2023).

Pascanu, Razvan, Tomas Mikolov y Yoshua Bengio. “On the Difficulty of Training Recurrent Neural Networks”. En: *International Conference on Machine Learning (ICML)*. 2013, págs. 1310-1318.

Perniss, Pamela, Jenny Lu y Gary Morgan. “The visual complexity of sign languages”. En: *Cognitive Science* 42.S3 (2018), págs. 911-939. DOI: 10.1111/cogs.12566.

Rodriguez, Jefferson y Fabio Martínez. “How important is motion in sign language translation?” En: *IET Computer Vision* 15.3 (2021), págs. 224-234.

Sandler, Wendy y Diane Lillo-Martin. *Sign Language and Linguistic Universals*. Cambridge University Press, 2006.

Seo, Youngjoo et al. *Structured Sequence Modeling with Graph Convolutional Recurrent Networks*. 2016. arXiv: 1612.07659 [stat.ML]. URL: https://arxiv.org/abs/1612.07659.

Sutskever, Ilya, Oriol Vinyals y Quoc V Le. “Sequence to sequence learning with neural networks”. En: *Advances in neural information processing systems*. 2014, págs. 3104-3112.

Vaswani, Ashish et al. *Attention Is All You Need*. 2017. arXiv: 1706.03762 [cs.CL].

Woll, Bencie, RL Sutton-Spence y Frances Elton. “Multilingualism: The global approach to sign languages”. En: *The sociolinguistics of sign languages*. Cambridge University Press, 2001, págs. 8-32.

– “Multilingualism: The global approach to sign languages”. En: *The sociolinguistics of sign languages*. Cambridge University Press, 2001, págs. 8-32.

Xu, Rui et al. “Positional encoding as spatial inductive bias in gans”. En: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, págs. 13569-13578.

Yin, Kayo y Jesse Read. *Better Sign Language Translation with STMC-Transformer*. 2020. arXiv: 2004.00588 [cs.CL].

Zhang, Han et al. *Self-Attention Generative Adversarial Networks*. 2019. arXiv: 1805.08318 [stat.ML].

Zhang, Zhaopeng et al. “Sign language translation using multimodal recurrent neural networks with visual attention”. En: *2019 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2019, págs. 1368-1374. DOI: 10.1109/ICRA.2019.8793686.

Zhou, Hao et al. “Spatial-Temporal Multi-Cue Network for Sign Language Recognition and Translation”. En: *IEEE Transactions on Multimedia* 24.4 (2022), págs. 768-779. DOI: 10.1109/TMM.2021.3059098.