

# Proyecto COIL

## VINCULACIÓN INTERNACIONAL COIL-UAEH-UDEA

### **EQUIPO MÉXICO:**

LÓPEZ SÁNCHEZ MARIA JOSELINE

REYES GUTIÉRREZ ALAN

ROMÁN ORTIZ GUSTAVO

SÁNCHEZ CHÁVEZ OJANNA ARISBETH

SEGURA PÁRAMO EDUARDO DANIEL

### **EQUIPO COLOMBIA:**

HENAO ZULUGA RICARDO

## Índice

<b>Tabla de ilustraciones.....</b>	<b>2</b>
<b>Introducción .....</b>	<b>3</b>
<b>Marco teórico.....</b>	<b>4</b>
<b>Autómatas y Compiladores .....</b>	<b>4</b>
<b>Extracción de información de textos, minería de textos (procesamiento del lenguaje natural (PLN)) .....</b>	<b>5</b>
Preprocesamiento del texto .....	5
Tokenización .....	5
Eliminación de palabras vacías (stopwords) .....	6
<b>Objetivo general.....</b>	<b>7</b>
<b>Objetivos específicos .....</b>	<b>7</b>
<b>Desarrollo .....</b>	<b>8</b>
<b>Análisis Léxico: Autómatas empleados para reconocer los tokens requeridos para las resoluciones. ....</b>	<b>8</b>
<b>Análisis Sintáctico: Gramáticas regulares utilizada .....</b>	<b>8</b>
<b>Traducción .....</b>	<b>9</b>
Traducción de modismos de Medellín.....	9
Programa de LEX para el análisis léxico del documento fuente (tokenización).....	11
Programa de LEX para la sustitución de palabras.....	13
<b>Programa de LEX para la eliminación de palabras vacías (stopwords) .....</b>	<b>16</b>
<b>Análisis de Sentimientos .....</b>	<b>18</b>
Análisis de sentimiento basado en reglas.....	19
Análisis de sentimiento mediante aprendizaje automático .....	19
Funcionamiento paso a paso del análisis de sentimientos.....	20
<b>Análisis de sentimientos en el texto procesado (Identificación de palabras) .....</b>	<b>21</b>
<b>Lista de palabras con sentimientos y su análisis.....</b>	<b>24</b>
<b>Resultados obtenidos del Formulario .....</b>	<b>25</b>
<b>Conclusiones .....</b>	<b>27</b>
<b>Referencias .....</b>	<b>28</b>
<b>Anexo A. Documento fuente que incluye regionalismos propios de Colombia. ....</b>	<b>29</b>
<b>Anexo B. Evidencias de la comunicación que se obtuvo con nuestro par Colombiano ....</b>	<b>32</b>

## Tabla de ilustraciones

Figura 1. Diagrama de transición. ....	5
Figura 2. La imagen muestra el resultado de la ejecución del código del listado 1.....	12
Figura 3. La imagen muestra el resultado de la ejecución del código del listado 2.....	15
Figura 4. La imagen muestra el resultado de la ejecución del código del listado 3.....	18
Figura 5. Muestra la captura de pantalla con las preguntas y las respuestas obtenidas ..	25
Figura 6. Captura de pantalla que muestra el total de respuestas obtenidas en Google Forms .....	26
Figura 7. Captura de pantalla de la conversión que se tuvo a través de correo electrónico .....	32
Figura 8. Conversación via WhatsApp donde se acordó horario .....	33
Figura 9. Captura de llamada via WhatsApp con todo el equipo presente .....	33
Figura 10. Ultimo mensaje enviado sin respuesta.....	34

## Introducción

El presente documento es el resultado del proyecto de Vinculación Internacional COIL entre la UAEH y la UDEA, centrado en la intersección de la teoría de compiladores, el procesamiento de lenguaje natural (PLN) y la educación diferenciada. El objetivo principal es explorar cómo las herramientas de análisis léxico computacional pueden emplearse para promover la inclusión educativa, abordando las barreras creadas por las jergas lingüísticas.

A lo largo de este trabajo, el lector encontrará primero el Marco Teórico, que fundamenta el proyecto en tres pilares: los Autómatas Finitos Deterministas (AFD) como base del reconocimiento léxico, los principios del Procesamiento de Lenguaje Natural (PLN) aplicados a textos, y la importancia de la educación diferenciada.

Posteriormente, la sección de Desarrollo detalla la implementación práctica de estos conceptos. Se presenta la construcción de analizadores léxicos utilizando la herramienta Lex para procesar un texto fuente en español con regionalismos colombianos. El lector verá el proceso en tres fases:

1. Un programa para la identificación y conteo de regionalismos específicos.
2. Un segundo programa que realiza la "traducción" o sustitución de dichos regionalismos por términos más neutrales.
3. Un tercer programa dedicado a la limpieza del texto mediante la eliminación de palabras vacías (*stopwords*).

Finalmente, aunque se optó por un análisis manual debido a la naturaleza del texto, se incluye una sección sobre Análisis de Sentimientos, donde se identifican y explican las palabras del texto traducido que connotan una intención o emoción específica. Este proyecto demuestra la viabilidad de usar la teoría de autómatas para la adaptación de contenido educativo.

## Marco teórico

### **Autómatas y Compiladores**

Un **Autómata Finito Determinista (AFD)** se define como una tupla:

$$AFD = (\Sigma, Q, \delta, q_0, F)$$

- $\Sigma$  es el alfabeto de entrada
- $Q$  es el conjunto finito y no vacío de los estados
- $\delta$  es la función de transición,  $\delta: Q \times \Sigma \rightarrow Q$
- $q_0 \in Q$  es el estado inicial
- $F \subset Q$  es un conjunto de estado finales de aceptación ( $F \neq \emptyset$ )

Sea  $A = (\Sigma, Q, \delta, q_0, F)$  un AFD y sea  $w = w_1 w_2 \dots w_n$  una cadena de símbolos donde  $w_i \in \Sigma$ .

Entonces, **A acepta w** si existe una secuencia de estados  $r_0, r_1, \dots, r_n \in Q$  con tres condiciones:

1.  $r_0 = q_0$
2.  $\delta(r_i, w_{i+1}) = r_{i+1}$  para  $i = 0, 1, \dots, n-1$
3.  $r_n \in F$

La condición 1 establece que el AFD comienza en el estado inicial.

La condición 2 establece que el AFD cambia desde un estado hacia otro estado de acuerdo con la función de transición.

La condición 3 establece que el AFD acepta la cadena de entrada si termina en un estado de aceptación. Entonces, **A reconoce el lenguaje L** si  $L = \{w | A \text{ acepta } w\}$ .

Una **tabla de transición de estados** es una tabla que muestra a qué estado se moverá un autómata finito, basándose en el estado actual y las entradas. Una tabla de estados es esencialmente una tabla de verdad en la cual algunas de las entradas son el estado actual, y las salidas incluyen el siguiente estado, junto con otras salidas.

Generalmente se asocia con cada autómata un **diagrama de transición de estados**. Cada nodo del diagrama corresponde a un estado. El estado inicial se indica mediante una flecha que no tiene nodo origen. Los estados finales se representan con un círculo doble. Si existe una transición del estado  $e_i$  al estado  $e_j$  para un símbolo de entrada  $a$ , existe entonces un arco rotulado con  $a$  desde el nodo  $e_i$  al nodo  $e_j$ ; es decir que  $\delta(e_i, a) = e_j$ , se representa en el diagrama como se muestra en la Figura 1.

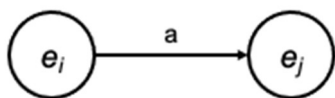


Figura 1. Diagrama de transición.

## ***Extracción de información de textos, minería de textos (procesamiento del lenguaje natural (PLN))***

### **Preprocesamiento del texto**

El Preprocesamiento del Texto es una fase crucial en la Minería de Textos y el PLN. Consiste en limpiar y transformar el texto sin formato (*raw text*) para que sea apto para el análisis computacional. Un texto bien preprocesado asegura la calidad y precisión de los resultados del análisis posterior.

### **Tokenización**

La Tokenización es el proceso de dividir una o varias frases en unidades más pequeñas llamadas *tokens*. Un *token* puede ser una palabra, un número o un signo de puntuación. Este proceso es el primer paso para dar estructura a un texto.

Un ejemplo de tokenización es el siguiente:

**Se tiene la frase: "El sol brilla."**

**Se tokeniza en: \$\$'El', 'sol', 'brilla', '.\$\$**

Como se observa, la frase se dividió en tres unidades donde cada una viene siendo una palabra.

### **Eliminación de palabras vacías (stopwords)**

Las *Palabras Vacías* (*Stopwords*) son términos muy comunes en un idioma (*en el caso del español las preposiciones, artículos, conjunciones, etc., se consideran como palabras vacías*) que generalmente no aportan un significado sustancial al contexto del documento. La eliminación de estos términos reduce el tamaño del vocabulario, el ruido computacional y acelera el procesamiento sin sacrificar el significado esencial del texto.

## **Objetivo general**

Promover la inclusión educativa mediante el diseño de experiencias de aprendizaje diferenciadas que consideren las jergas lingüísticas de los estudiantes, apoyándose en herramientas tecnológicas de reconocimiento y análisis léxico computacional.

### ***Objetivos específicos***

1. Construir el documento fuente siguiendo las especificaciones para la producción del texto.
2. Implementar un analizador léxico utilizando Lex, capaz de identificar, clasificar y validar los regionalismos colombianos.
3. Desarrollar un sistema de procesamiento lingüístico que traduzca el documento fuente a un lenguaje comprensible para estudiantes latinoamericanos.
4. Abordar el tema de análisis de sentimientos para la identificación y posterior explicación de las palabras con sentimientos presentes en el texto.



## Desarrollo

***Análisis Léxico: Autómatas empleados para reconocer los tokens requeridos para las resoluciones.***

$$AFD = (\Sigma, Q, \delta, q_0, F)$$

- $\Sigma$  es el alfabeto de entrada
- $Q$  es el conjunto finito y no vacío de los estados
- $\delta$  es la función de transición,  $\delta: Q \times \Sigma \rightarrow Q$
- $q_0 \in Q$  es el estado inicial
- $F \subset Q$  es un conjunto de estado finales de aceptación ( $F \neq \emptyset$ )

Sea  $A = (\Sigma, Q, \delta, q_0, F)$  un AFD y sea  $w = w_1 w_2 \dots w_n$  una cadena de símbolos donde  $w_i \in \Sigma$ .

***Análisis Sintáctico: Gramáticas regulares utilizada***

$$G = \{\Sigma_T, \Sigma_N, S, P\}$$

donde:

- $\Sigma_T$  es un alfabeto de símbolos terminales
- $\Sigma_N$  es un alfabeto de símbolos no terminales
- $S$  es el símbolo inicial de la gramática
- $P$  es un conjunto de producciones gramaticales

Además, se cumple:

$$S \in \Sigma_N$$

$$\Sigma_T \cap \Sigma_N = \emptyset$$

$$\Sigma = \Sigma_T \cup \Sigma_N$$

La gramática formal  $G$  permite generar un lenguaje  $L = \{x \in \Sigma_T^* | S \xrightarrow{*} x\}$

Las **expresiones regulares** permiten describir con exactitud y sencillez cualquier lenguaje regular.

### ***Traducción***

#### **Traducción de modismos de Medellín**

La siguiente lista de palabras son aquellas que se identificaron como necesarias para la traducción, se pone la palabra y el significado proporcionado por el par colombiano. Palabras extraídas del texto original proporcionado por el par Colombiano y presente en el **Anexo A**.

**Tintico o Tinto:** Café sencillo hecho en casa (Café oscuro o americano)

**Casa de tapia:** Casa de barro (arcilla o adobe, la técnica es diferente).

**Guarapos:** Jugo de caña (bebida de caña de azúcar con limón y azúcar fermentado)(ir a tomar bebidas alcohólicas, aguardiente/pulque).

**Jartera:** Fastidio (expresión de estrés o cansado de lo mismo). Cuando a alguien lo molestan mucho.

**Gritico:** Grito en diminutivo (gritito).

**Ventarrón:** Viento fuerte (Lluvia muy fuerte).

**Pieza:** Cuarto o habitación.

**Ratico:** ratito.

**Berraco:** Muy complicado (significa diversas cosas depende del contexto), también se puede utilizar para expresar que alguien es muy bueno en algo. En este contexto es: *lo más duro/más difícil de la situación*.

**Sabroson:** Persona simpática, relajada (Se cree muy bonito solo en ese contexto).

**Sabroson creyéndose el cuento:** Cree que realmente es muy bonito.

**Pillan:** Atrapar (descubrir a alguien haciendo algo incorrecto / agarrar alguien desprevenido o embobado).

**Enjalman:** Ensillar (Hace alusión a cargar un burro como por ejemplo: con un costal lleno de paja el cual montan encima del burro y no lo sueltan).

**Verraquera:** En este contexto hace alusión que el grito fue con rabia y lo hizo gritó con toda la fuerza que tenía en ese momento.

**Se le sube la bruja:** Se le sube el muerto (parálisis del sueño)

**Maña:** Obsesión.

## Programa de LEX para el análisis léxico del documento fuente (tokenización)

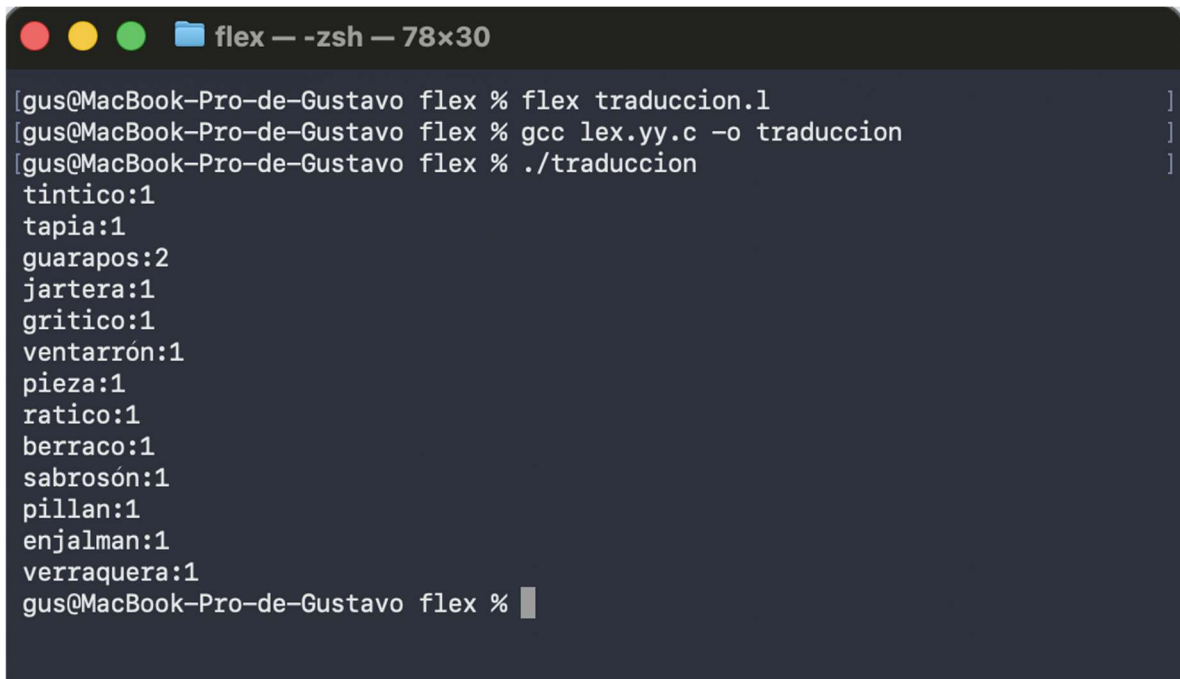
**Listado 1:** Implementación de código LEX para comprobar que el reconocimiento de las regionalismos Colombianos sea correcto.

```
1) %option noyywrap
2) %{
3)     #include <stdio.h>
4)     FILE *yyin;
5)     int tintico = 0, tapia = 0, guarapos = 0, jartera = 0,
    gritico = 0, ventarrón = 0, pieza = 0;
6)     int ratico = 0, berraco = 0, sabrosón = 0, pillan = 0,
    enjalman = 0, verraquera = 0;
7) %}
8) %%
9) tintico { tintico++; }
10)    tapia { tapia++; }
11)    guarapos { guarapos++; }
12)    guarapo { guarapos++; }
13)    jartera { jartera++; }
14)    gritico { gritico++; }
15)    ventarrón { ventarrón++;}
16)    pieza { pieza++;}
17)    ratico { ratico++; }
18)    berraco { berraco++; }
19)    sabrosón { sabrosón++; }
20)    pillan { pillan++; }
21)    enjalman { enjalman++; }
22)    verraquera { verraquera++; }
23)    %%
24)    int main(){
25)        yyin = fopen("./texto.txt", "r");
26)        if(!yyin){
27)            printf("No se pudo abrir el archivo texto.txt");
28)            return 1;
29)        }
30)        yylex();
31)        printf("tintico:%d\n", tintico);
32)        printf("tapia:%d\n", tapia);
33)        printf("guarapos:%d\n", guarapos);
34)        printf("jartera:%d\n", jartera);
35)        printf("gritico:%d\n", gritico);
36)        printf("ventarrón:%d\n", ventarrón);
37)        printf("pieza:%d\n", pieza);
38)        printf("ratico:%d\n", ratico);
39)        printf("berraco:%d\n", berraco);
40)        printf("sabrosón:%d\n", sabrosón);
41)        printf("pillan:%d\n", pillan);
42)        printf("enjalman:%d\n", enjalman);
43)        printf("verraquera:%d\n", verraquera);
44)        fclose(yyin);
```

```
45)          fclose(yyout);  
46)          return 0;  
47)      }
```

### Evidencias de ejecución desde línea de comandos

Se comprueba que el programa funciona correctamente ya que identificó el total de las palabras a traducir, como se muestra en la Figura 2.

A screenshot of a macOS terminal window titled 'flex — -zsh — 78x30'. The terminal shows the execution of a program named 'traduccion'. The output lists 15 words with their corresponding counts: tintico:1, tapia:1, guarapos:2, jartera:1, gritico:1, ventarrón:1, pieza:1, ratico:1, berraco:1, sabrosón:1, pillan:1, enjalman:1, verraquera:1. The prompt 'gus@MacBook-Pro-de-Gustavo flex %' is visible at the bottom.

```
flex % flex traduccion.1  
flex % gcc lex.yy.c -o traduccion  
flex % ./traduccion  
tintico:1  
tapia:1  
guarapos:2  
jartera:1  
gritico:1  
ventarrón:1  
pieza:1  
ratico:1  
berraco:1  
sabrosón:1  
pillan:1  
enjalman:1  
verraquera:1  
gus@MacBook-Pro-de-Gustavo flex %
```

Figura 2. La imagen muestra el resultado de la ejecución del código del listado 1.

## Programa de LEX para la sustitución de palabras

**Listado 2:** Código del segundo programa de LEX encargado de la sustitución de los regionalismos colombianos con palabras neutrales.

```
1) %option noyywrap
2) %{
3)     #include <stdio.h>
4)     FILE *yyin;
5)     FILE *yyout;
6)     int tintico = 0, tapia = 0, guarapos = 0, jartera = 0, gritico
   = 0, ventarrón = 0, pieza = 0;
7)     int ratico = 0, berraco = 0, sabrosón = 0, pillan = 0, enjalman
   = 0, verreaquera = 0;
8) %}
9) %%
10)    tintico { fprintf(yyout, "café"); tintico++;}
11)    tapia { fprintf(yyout, "arcilla"); tapia++;}
12)    guarapos { fprintf(yyout, "tragos"); guarapos++;}
13)    guarapo { fprintf(yyout, "alcohol"); guarapos++;}
14)    jartera { fprintf(yyout, "fastidio"); jartera++;}
15)    gritico { fprintf(yyout, "gritito"); gritico++;}
16)    ventarrón { fprintf(yyout, "viento fuerte"); ventarrón++;}
17)    pieza { fprintf(yyout, "habitación"); pieza++;}
18)    ratico { fprintf(yyout, "ratito"); ratico++;}
19)    berraco { fprintf(yyout, "complicado"); berraco++;}
20)    sabrosón { fprintf(yyout, "guapo"); sabrosón++;}
21)    pillan { fprintf(yyout, "agarran desprevenido"); pillan++;}
22)    enjalman { fprintf(yyout, "montan"); enjalman++;}
23)    verreaquera { fprintf(yyout, "fuerza"); verreaquera++;}
24)    %%
25)    int main(){
26)        yyin = fopen("./texto.txt", "r");
27)        yyout = fopen("./textoT.txt", "w");
28)        if(!yyin){
29)            printf("No se pudo abrir el archivo texto.txt");
30)            return 1;
31)        } else if(!yyout){
32)            printf("No se pudo abrir el archivo textoT.txt");
33)            return 1;
34)        }
35)        yylex();
36)        printf("Iniciando proceso de analisis\n");
37)        printf("Analizando texto...\n");
38)        printf("Proceso de analisis exitosamente completado\n");
39)        fclose(yyin);
40)        fclose(yyout);
41)        return 0; }
```

## Texto procesado y con las palabras sustituidas:

"La noche en que se me subió la bruja"

Leyenda contada por Ricardo Henao

Vea pues, miijo, siéntese y pídase un café, que le voy a echar un cuento que no me lo invento yo, eso me pasó de verdad, aquí mismito en Santa Fe de Antioquia, una noche de esas en que el calor se pega a la piel y ni el abanico da tregua. Yo vivía en la casa vieja de mi abuela, una casa de arcilla y teja, con un patio lleno de matas de ruda, sábila y albahaca, y con ese olor a tierra caliente que uno no olvida ni queriendo.

Esa noche, después de haberme tomado unos tragos con los compas en la esquina, llegué medio cansado, con ganas de caer redondo y de mirar pa' dentro. El pueblo estaba callado, sólo se oía el canto de los grillos y un perro que ladraba a lo lejos, como si hubiera visto al mismísimo diablo. Me acosté, rezando de carrerita el Padre Nuestro —por si las moscas—, y cerré los ojos. Pero vea, no habían pasado ni cinco minutos cuando sentí una pesadez en el pecho, como si me hubieran echado una piedra encima. Quise moverme, pero ni pío pude. Ni un dedo. Ahí fue cuando me di cuenta de que se me había subido la bruja.

Hermano, yo sentía el aire helado en el cuello, y ese silencio raro, como si el mundo se hubiera quedado sin sonido. Quise gritar, pero la voz no me salía. De reojo, vi una sombra encima mío, con una risita chiquita, burlona, y un olor a azufre mezclado con perfume barato. Yo pensé: "Ave María Purísima, esto no es sueño, esto es brujería del monte".

Mi abuela siempre decía: "Cuando le pase eso, diga: mañana venga por sal, que ahí se le quita la maña". Pero dígame usted eso a alguien que no puede ni respirar, ¡qué fastidio! Me tocó hacer fuerza con el alma. En eso, sentí que la sombra se me pegó más al pecho, y me zumbó en el oído algo que no entendí bien, pero que me erizó hasta el alma. Entonces, como pude, solté un gritito ahogado:

—¡Mañana venga por sal!

Y vea, miijo, fue decir eso y la cosa esa pegó un brinco, como si le hubieran echado agua bendita. Sentí un viento fuerte por la habitación y un alarido que hizo temblar las tejas. Me quedé quieto, empapado en sudor, temblando como un flan. Cuando por fin pude moverme, me levanté y encendí la luz, pero no había nadie, sólo el olor raro y un mechón de pelo negro en la almohada. ¡Ave María! Más asustado que gallina en patio ajeno, eso sí.

Al otro día amanecí con los ojos como un sapo, sin dormir un ratito. Pero vea lo más complicado: como a las nueve de la mañana, tocaron la puerta. Yo salí todavía medio zombi, y ahí estaba una señora del barrio, una que casi nunca hablaba con nadie.

—Vecino —me dijo con esa vocecita empalagosa—, ¿no me regala un poquito de sal?

Y vea, miijo, yo me quedé frío. Sentí que se me heló la sangre. Era ella, la mismita.

Le pasé la sal temblando y le dije:

—Tenga, vecina, llévese toda si quiere, pero no vuelva por acá ni en sueños.

Ella soltó una risita, de esas que uno no sabe si son de burla o de agradecimiento, y se fue despacio, arrastrando las chanclas por la calle empedrada. Desde entonces, cada vez que paso por su casa, los pelos se me paran como alambre pelado.

Dicen los viejos que esas brujas se suben por envidia, o porque uno anda muy guapo, creyéndose el cuento. Que, si lo agarran desprevenido desprevenido, lo montan como mula y no lo sueltan hasta que uno diga la frase mágica. Y vea, no sé si será verdad o no, pero desde esa noche duermo con un escapulario en el cuello, una rama de ruda bajo la almohada y la luz prendida. Por si las moscas, pues.

Y si le soy sincero, uno se ríe ahora, pero esa noche casi me da un patatús. Dicen que “el que no ha visto el diablo, no cree en él”, y yo le digo: a mí no me lo contaron, a mí me pasó.

Aunque bueno, no hay mal que por bien no venga, porque desde ese día dejé de trasnocharme en la esquina y de beber tanto alcohol, que eso, dicen, atrae los malos vientos.

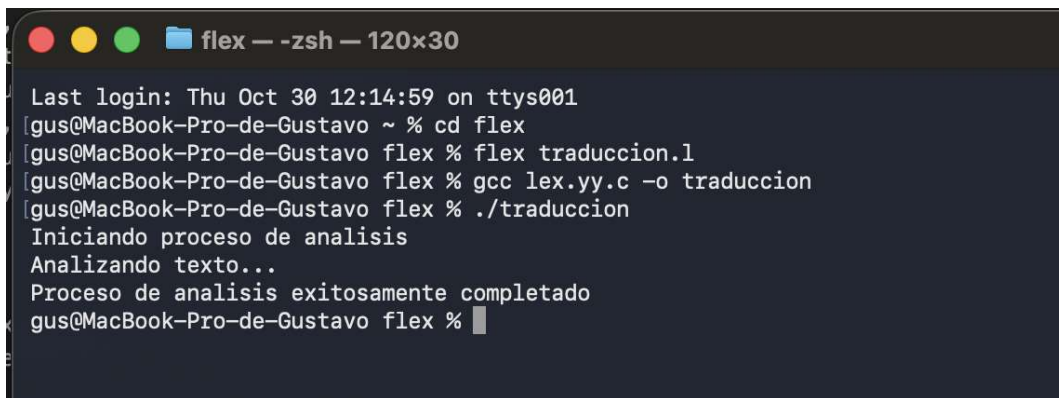
Así que ya sabe, miijo: si alguna vez se le sube la bruja, no se me quede callado ni tieso, que ahí sí fue. Saque fuerzas del alma y grite con toda la fuerza:

—¡Mañana venga por sal!

Y si al otro día le tocan la puerta... pues, no diga que no le advertí.

## Evidencias de ejecución desde línea de comandos

Se comprueba que el programa funciona correctamente para la traducción del texto, como se muestra en la Figura 3.



```
flex — -zsh — 120x30

Last login: Thu Oct 30 12:14:59 on ttys001
gus@MacBook-Pro-de-Gustavo ~ % cd flex
gus@MacBook-Pro-de-Gustavo flex % flex traduccion.1
gus@MacBook-Pro-de-Gustavo flex % gcc lex.yy.c -o traduccion
gus@MacBook-Pro-de-Gustavo flex % ./traduccion
Iniciando proceso de analisis
Analizando texto...
Proceso de analisis exitosamente completado
gus@MacBook-Pro-de-Gustavo flex %
```

Figura 3. La imagen muestra el resultado de la ejecución del código del listado 2.



## ***Programa de LEX para la eliminación de palabras vacías (stopwords)***

**Listado 3:** Código de LEX para el análisis léxico del documento fuente y la eliminación de palabras vacías (sin un significado propio).

```
1)      %option noyywrap case-insensitive
2) %{
3)      #include<stdio.h>
4)      FILE *yyin;
5)      FILE *yyout;
6) %}
7) %%
8) ([[:space:]]|^)(La|le|a|al|algo|algunas|alguno|algunos|ante|antes
   |como|con|contra|cual|cuales|cualquier|cualquiera|cuando|de|del|d
   esde|donde|durante|e|el|ella|ellas|ello|ellos|en|entre|era|erais|
   eran|eras|es|esa|esas|ese|eso|esos|esta|estaba|estaban|estabas|es
   tado|estamos|están|estar|este|estos|fue|fueron|fui|fuimos|ha|habé
   is|habían|habías|haber|había|hago|hasta|la|las|le|les|lo|los|me|m
   i|mis|mucha|muchas|mucho|muchos|muy|nada|ni|no|nos|nosotras|nosot
   ros|o|os|otra|otras|otro|otros|para|pero|poca|pocas|poco|pocos|po
   r|porque|que|quién|quiénes|se|ser|será|serán|serás|sería|serían|s
   í|sí|sin|sobre|sois|somos|son|soy|su|sus|te|ti|tiene|tienen|todo|
   todos|tu|tus|un|una|unas|uno|unos|usted|ustedes|y|ya|yo)/([[:spac
   e:]]|$) { fprintf(yyout, " "); }
9) \n { fprintf(yyout, "\n"); }
10)     . { fprintf(yyout, "%s", yytext); }
11)     %%
12)     int main() {
13)         yyin = fopen("texto.txt", "r");
14)         yyout = fopen("textoV.txt", "w");
15)
16)         if (!yyin) {
17)             printf("No se pudo abrir el archivo texto.txt\n");
18)             return 1;
19)         }
20)         if (!yyout) {
21)             printf("No se pudo abrir el archivo textoV.txt\n");
22)             return 1;
23)         }
24)         yylex();
25)         printf("Iniciando proceso de analisis\n");
26)         printf("Analizando texto...\n");
27)         printf("Eliminacion de palabras vacias completado con
   exito\n");
28)         fclose(yyin);
29)         fclose(yyout);
30)         return 0;
31)     }
```

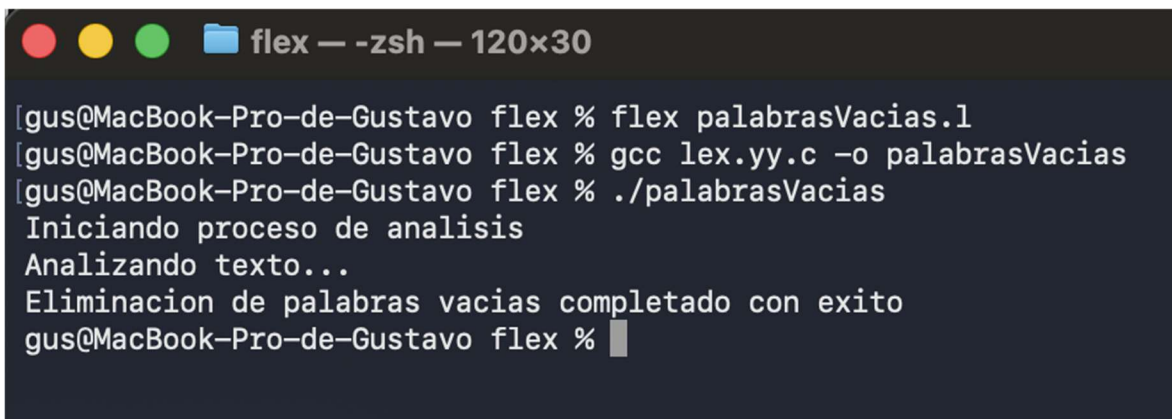
**Texto procesado sin las palabras vacías:**

"La noche subió bruja"  
Leyenda contada Ricardo Henao  
Vea pues, mijo, siéntese pídase tintico, voy echar cuento  
invento yo, pasó verdad, aquí mismito Santa Fe Antioquia,  
noche calor pega piel abanico da tregua. vivía casa  
vieja abuela, casa tapia teja, patio lleno matas ruda,  
sábila albahaca, olor tierra caliente olvida queriendo.  
noche, después haberme tomado guarapos compas esquina,  
llegué medio cansado, ganas caer redondo mirar pa' dentro.  
pueblo callado, sólo oía canto grillos perro ladraba  
lejos, hubiera visto mismísimo diablo. acosté, rezando  
carrerita Padre Nuestro -por moscas-, cerré ojos. vea,  
pasado cinco minutos sentí pesadez pecho, hubieran echado  
piedra encima. Quise moverme, pío pude. dedo.  
Ahí di cuenta subido bruja.  
Hermano, sentía aire helado cuello, silencio raro, mundo  
hubiera quedado sonido. Quise gritar, voz salía. reajo, vi  
sombra encima mío, risita chiquita, burlona, olor azufre  
mezclado perfume barato. pensé: "Ave María Purísima, esto  
sueño, esto brujería monte". abuela siempre decía: "Cuando  
pase eso, diga: mañana venga sal, ahí quita maña". dígame  
alguien puede respirar, ¡qué jartera! tocó hacer fuerza  
alma. eso, sentí sombra pegó más pecho, zumbó oído  
entendí bien, erizó alma. Entonces, pude, solté gritico  
ahogado:  
-¡Mañana venga sal! vea, mijo, decir cosa pegó brinco,  
hubieran echado agua bendita. Sentí ventarrón pieza alarido  
hizo temblar tejas. quedé quieto, empapado sudor, temblando  
flan. fin pude moverme, levanté encendí luz, nadie, sólo  
olor raro mechón pelo negro almohada. ¡Ave María! Más  
asustado gallina patio ajeno, sí. día amanecí ojos sapo,  
dormir ratico. vea más berraco: nueve mañana, tocaron  
puerta. salí todavía medio zombi, ahí señora barrio, casi  
nunca hablaba nadie.  
-Vecino -me dijo vocecita empalagosa-, ¿no regala poquito  
sal? vea, mijo, quedé frío. Sentí heló sangre. ella,  
mismita. pasé sal temblando dije:  
-Tenga, vecina, llévese toda quiere, vuelva acá sueños.  
soltó risita, sabe burla agradecimiento, despacio,  
arrastrando chanclas calle empedrada. entonces, cada vez  
paso casa, pelos paran alambre pelado.  
Dicen viejos brujas suben envidia, anda sabrosón,  
creyéndose cuento. Que, pillan desprevenido, enjalman mula  
sueltan diga frase mágica. vea, sé verdad no, noche  
duermo escapulario cuello, rama ruda bajo almohada luz  
prendida. moscas, pues. sincero, ríe ahora, noche casi  
da patatús. Dicen "el visto diablo, cree él", digo:  
mí contaron, mí pasó.  
Aunque bueno, hay mal bien venga, día dejé trasnocharme

```
esquina beber tanto guarapo, eso, dicen, atrae malos vientos.  
Así sabe, miijo: alguna vez sube bruja, quede callado  
tieso, ahí fue. Saque fuerzas alma grite toda verraquera:  
-¡Mañana venga sal! día tocan puerta... pues, diga  
advertí.
```

## Evidencias de ejecución desde línea de comandos

Se comprueba que el programa funciona correctamente para la eliminación de las palabras vacías, como se muestra en la Figura 4.



```
flex — -zsh — 120x30  
[gus@MacBook-Pro-de-Gustavo flex % flex palabrasVacias.l  
[gus@MacBook-Pro-de-Gustavo flex % gcc lex.yy.c -o palabrasVacias  
[gus@MacBook-Pro-de-Gustavo flex % ./palabrasVacias  
Iniciando proceso de analisis  
Analizando texto...  
Eliminacion de palabras vacias completado con exito  
gus@MacBook-Pro-de-Gustavo flex %
```

Figura 4. La imagen muestra el resultado de la ejecución del código del listado 3.

## Análisis de Sentimientos

El análisis de sentimiento es una técnica de procesamiento del lenguaje natural (NLP) que utiliza la lingüística computacional y el machine learning para detectar el tono emocional detrás de los datos de texto. Esto permite a las organizaciones identificar un sentimiento positivo, neutral o negativo hacia tu marca, productos, servicios o ideas.

Las tecnologías centrales incluyen:

- **Procesamiento de lenguaje natural (NLP):** Permite que las máquinas procesen e interpreten el lenguaje humano

- **Lingüística computacional:** proporciona marcos de trabajo lingüísticos para el análisis de texto
- **Machine learning (ML):** Los modelos aprenden patrones de texto etiquetado para clasificar sentimientos

### **Análisis de sentimiento basado en reglas**

En el enfoque basado en reglas, el software está entrenado para clasificar ciertas palabras clave en un bloque de texto basado en grupos de palabras, o léxicos, que describen la intención del autor. Por ejemplo, las palabras en un léxico positivo pueden incluir "asequible", "rápido" y "bien hecho", mientras que las palabras en un léxico negativo podrían mostrar "caro", "lento" y "mal hecho". A continuación, el software busca en el clasificador las palabras del léxico positivo o negativo y calcula una puntuación total del sentimiento basada en el volumen de palabras utilizadas y la puntuación del sentimiento de cada categoría.

### **Análisis de sentimiento mediante aprendizaje automático**

Con un enfoque de aprendizaje automático (ML), se utiliza un algoritmo para entrenar al software para medir el sentimiento en un bloque de texto utilizando palabras que aparecen en el texto, así como el orden en que aparecen. Los desarrolladores utilizan algoritmos de análisis de sentimiento para enseñar al software a identificar las emociones en el texto de forma similar a como lo hacen los humanos. Los modelos de ML siguen "aprendiendo" de los datos que se les suministran, de ahí el nombre de "machine learning" (aprendizaje automático). Estos son algunos de los algoritmos de clasificación más utilizados:

- **Regresión lineal:** un algoritmo de estadísticas que describe un valor (Y) basado en un conjunto de características (X).
- **Naive Bayes:** Algoritmo que utiliza el teorema de Bayes para categorizar palabras en un bloque de texto.

- Máquinas de vectores de soporte: algoritmo de clasificación rápido y eficiente utilizado para resolver problemas de clasificación de dos grupos.
- Aprendizaje profundo (DL): también conocido como red neuronal artificial, el aprendizaje profundo es una técnica avanzada de machine learning que vincula varios algoritmos para imitar la función cerebral humana.

### **Funcionamiento paso a paso del análisis de sentimientos**

1. **Ingesta de texto:** datos de texto sin procesar se recopilan de una variedad de fuentes, incluidos correos electrónicos, tickets de soporte, logs de chat, redes sociales y reseñas de clientes.
2. **Preprocesamiento de texto:** El texto sin procesar se limpia y se normaliza:
  1. Tokenización: los datos (texto) se dividen en palabras o frases
  2. Minúsculas: la entrada está estandarizada
  3. Eliminación de palabras vacías: se filtran las palabras comunes no informativas
  4. Stemming/lematización: Descompone las palabras en sus formas más básicas
  5. Reconocimiento de entidades con nombre (NER): El proceso de reconocer nombres propios y entidades
3. **Extracción de características:** Se crean representaciones numéricas estructuradas a partir del texto:
  1. Bolsa de palabras (BoW) o TF-IDF para modelos de vectores dispersos
  2. Embeddings de palabras (Word2Vec, GloVe, BERT, etc.) para contexto semántico.
  3. Vectores contextuales (p. ej., de modelos basados en transformers)
4. **Clasificación de sentimientos:** El texto se categoriza en categorías de sentimiento utilizando modelos de machine learning o de aprendizaje profundo:
  1. Modelos basados en reglas (usando léxicos de sentimientos y heurísticas lingüísticas)

2. Modelos de ML tradicionales (Naïve Bayes, SVM, regresión logística)
3. Modelos neuronales (LSTM, CNN, transformers)
5. **Salida y puntuación:** Cada texto de entrada tiene una puntuación a lo largo de un continuo de sentimiento (por ejemplo, de -1 a +1) o se etiqueta como positivo, negativo o neutral. A continuación, estos metadatos de sentimiento son:
  1. Indexados para filtrado o búsqueda
  2. Combinados para analíticas y dashboards
  3. Usados para activar advertencias (por ejemplo, cuando el sentimiento negativo aumenta)
6. **Comentarios + actualizaciones del modelo:** El modelo se puede ajustar o reentrenar utilizando salidas etiquetadas.

*Debido a la limitada cantidad de palabras clave pertinentes en el documento fuente, se consideró inviable la implementación de un programa para un análisis de sentimientos automatizado y extenso. Por consiguiente, se optó por omitir el desarrollo de dicho componente. En su lugar, el análisis se realizó manualmente, operando directamente sobre el texto ya traducido. A continuación, se presenta el texto resultante de este proceso, con las palabras de interés resaltadas y la lista correspondiente.*

### **Análisis de sentimientos en el texto procesado (Identificación de palabras)**

#### **“La noche en que se me subió la bruja”**

Leyenda contada por Ricardo Henao

Vea pues, miijo, siéntese y pídase un café, que le voy a echar un cuento que no me lo invento yo, eso me pasó de verdad, aquí mismito en Santa Fe de Antioquia, una noche de esas en que el calor se pega a la piel y ni el abanico da tregua. Yo vivía en la casa vieja de mi abuela, una casa de arcilla y teja, con un patio lleno de matas

de ruda, sábila y albahaca, y con ese olor a tierra caliente que uno no olvida ni queriendo.

Esa noche, después de haberme tomado unos tragos con los compas en la esquina, llegué medio cansado, con ganas de caer redondo y de mirar pa' dentro. El pueblo estaba callado, sólo se oía el canto de los grillos y un perro que ladraba a lo lejos, como si hubiera visto al mismísimo diablo. Me acosté, rezando de carrerita el Padre Nuestro —por si las moscas—, y cerré los ojos. Pero vea, no habían pasado ni cinco minutos cuando sentí una pesadez en el pecho, como si me hubieran echado una piedra encima. Quise moverme, pero ni pío pude. Ni un dedo.

Ahí fue cuando me di cuenta de que se me había subido la bruja.

Hermano, yo sentía el aire helado en el cuello, y ese silencio raro, como si el mundo se hubiera quedado sin sonido. Quise gritar, pero la voz no me salía. De reojo, vi una sombra encima mío, con una risita chiquita, burlona, y un olor a azufre mezclado con perfume barato. Yo pensé: “Ave María Purísima, esto no es sueño, esto es brujería del monte”.

Mi abuela siempre decía: “Cuando le pase eso, diga: mañana venga por sal, que ahí se le quita la maña”. Pero dígame usted eso a alguien que no puede ni respirar, ¡qué fastidio! Me tocó hacer fuerza con el alma. En eso, sentí que la sombra se me pegó más al pecho, y me zumbó en el oído algo que no entendí bien, pero que me erizó hasta el alma. Entonces, como pude, solté un gritito ahogado:

—¡Mañana venga por sal!

Y vea, mijo, fue decir eso y la cosa esa pegó un brinco, como si le hubieran echado agua bendita. Sentí un viento fuerte por la habitación y un alarido que hizo temblar las tejas. Me quedé quieto, empapado en sudor, temblando como un flan. Cuando por fin pude moverme, me levanté y encendí la luz, pero no había nadie, sólo el olor raro y un mechón de pelo negro en la almohada. ¡Ave María! Más asustado que gallina en patio ajeno, eso sí.

Al otro día amanecí con los ojos como un sapo, sin dormir un **ratito**. Pero vea lo más complicado: como a las nueve de la mañana, tocaron la puerta. Yo salí todavía medio zombi, y ahí estaba una señora del barrio, una que casi nunca hablaba con nadie.

—Vecino —me dijo con esa **vocecita** empalagosa—, ¿no me regala un **poquito** de sal?

Y vea, mijo, yo me quedé frío. Sentí que se me heló la sangre. Era ella, la mismita.

Le pasé la sal temblando y le dije:

—Tenga, vecina, llévese toda si quiere, pero no vuelva por acá ni en sueños.

Ella soltó una risita, de esas que uno no sabe si son de burla o de agradecimiento, y se fue despacio, arrastrando las chancas por la calle empedrada. Desde entonces, cada vez que paso por su casa, los pelos se me paran como alambre pelado.

Dicen los viejos que esas brujas se suben por envidia, o porque uno anda muy guapo, creyéndose el cuento. Que, si lo agarran desprevenido, lo montan como mula y no lo sueltan hasta que uno diga la frase mágica. Y vea, no sé si será verdad o no, pero desde esa noche duermo con un escapulario en el cuello, una rama de ruda bajo la almohada y la luz prendida. Por si las moscas, pues.

Y si le soy sincero, uno se ríe ahora, pero esa noche casi me da un patatús. Dicen que “el que no ha visto el diablo, no cree en él”, y yo le digo: a mí no me lo contaron, a mí me pasó.

Aunque bueno, no hay mal que por bien no venga, porque desde ese día dejé de trasnocharme en la esquina y de beber tanto alcohol, que eso, dicen, atrae los malos vientos.

Así que ya sabe, mijo: si alguna vez se le sube la bruja, no se me quede callado ni tieso, que ahí sí fue. Saque fuerzas del alma y grite con toda la fuerza:



—¡Mañana venga por sal!

Y si al otro día le tocan la puerta... pues, no diga que no le advertí.

### ***Lista de palabras con sentimientos y su análisis***

**carrerita** (rezando de carrerita ): El sufijo '-ita' no indica que la oración sea corta, sino la *intención* de hacerla rápido, casi por trámite, denotando un ligero nerviosismo ("por si las moscas" ).

**risita** (vi una sombra... con una risita; Ella soltó una risita ): El sufijo '-ita' es clave. No es una "risa", es una "risita", lo que la vuelve burlona, inquietante y de intención oculta.

**chiquita** (risita chiquita ): Usado para intensificar la cualidad siniestra de la "risita".

**gritito** (solté un gritito ahogado ): El sufijo '-ito' describe un grito débil, que apenas salió, mostrando el sentimiento de miedo extremo y parálisis.

**ratito** (sin dormir un ratito ): Intensifica la falta total de descanso; ni siquiera pudo dormir "un poco", mostrando la fatiga y el tormento.

**vocecita** (con esa vocecita empalagosa ): Describe la voz de la vecina. El diminutivo aquí implica falsedad, una voz que finge ser dulce.

**poquito** (un poquito de sal ): Usado por la bruja. La intención del diminutivo es minimizar la petición, haciéndola sonar inocente cuando es todo lo contrario.

**pesadez** (sentí una pesadez en el pecho ): El sufijo '-ez' transforma la cualidad "pesado" en el sustantivo que describe el *sentimiento* físico de opresión y parálisis.

**mismísimo** (al mismísimo diablo ): El sufijo '-ísimo' es un intensificador superlativo. No solo vio al diablo, sino "al mismísimo", elevando el sentimiento de terror o asombro.

**Purísima** (¡Ave María Purísima! ): Usado en la exclamación de terror. El sufijo '-ísima' lleva la cualidad de "pura" al máximo, usándose como una súplica desesperada.

## Resultados obtenidos del Formulario

Con el propósito de evaluar la calidad y pertinencia de la traducción propuesta por la Universidad Autónoma del Estado de Hidalgo respecto al cuento elaborado por un estudiante de la Universidad de Antioquia, se diseñó y aplicó una encuesta dirigida a los pares académicos colombianos. No obstante, ante la ausencia de una respuesta oportuna por parte de dichos colaboradores, se optó por que los integrantes del equipo de trabajo completaran el instrumento, a fin de contar con una retroalimentación preliminar que permitiera avanzar en el análisis y ajuste de la traducción.

A continuación se presentan imágenes de los resultados obtenidos por medio de las encuestas (Figura 5 y Figura 6).

¿Crees que la esencia del relato cambio con las palabras que "tradujimos" ?  
5 respuestas

- si
- no
- No
- De manera general pienso que la traducción mantuvo la idea general de la historia
- No, la esencia e historia del relato se conserva

¿Cambiarías alguna palabra que sientes que no transmite la idea original?  
5 respuestas

- no
- Tal vez, por algunas mas efusivas
- No, pienso que transmiten la idea correcta
- Ninguna

¿Crees que se realizo un buen trabajo de traducción de regionalismos?  
5 respuestas

- Si
- si
- muy buen trabajo
- Si, la comunicación hizo que las palabras cambiadas a los regionalismos mexicanos transmitieran lo mismo

*Figura 5. Muestra la captura de pantalla con las preguntas y las respuestas obtenidas*

La conclusión a la que se llegó con las encuestas es que de manera general se logró el objetivo de la “traducción” de regionalismos específicos de Colombia a palabras neutrales o comprensivas para un público hispanohablante en general sin afectar o cambiar el sentimiento e intención del relato original permitiendo la resolución de los objetivos planteados al principio.

Form_Responses			
Marca temporal	¿Crees que la esencia del relato cambio con l	¿Cambiarías alguna palabra que sientes que	¿Crees que se realizo un buen trabajo de trad
10/11/2025 10:34:55	si	no	si
11/11/2025 22:37:51	no	no	muy buen trabajo
11/11/2025 22:38:12	No	Tal vez, por algunas mas efusivas	Si
11/11/2025 22:38:52	De manera general pienso que la traducción mant	No, pienso que transmiten la idea correcta	Si
11/11/2025 22:39:45	No, la esencia e historia del relato se conserva	Ninguna	Si, la comunicación hizo que las palabras cambia

Figura 6. Captura de pantalla que muestra el total de respuestas obtenidas en Google Forms

## Conclusiones

Este proyecto COIL ha representado una valiosa experiencia de aprendizaje interdisciplinario y colaborativo. A través de su desarrollo, logramos aplicar exitosamente los fundamentos teóricos de los lenguajes formales y autómatas en un problema práctico de Procesamiento de Lenguaje Natural (PLN).

Los objetivos planteados se cumplieron mediante la implementación de tres analizadores léxicos funcionales con la herramienta Lex. Estos programas demostraron ser eficaces para:

1. **Reconocer** los *tokens* correspondientes a regionalismos colombianos en el documento fuente.
2. **Traducir** el texto sustituyendo estos regionalismos por un léxico neutral, haciéndolo más accesible para otros estudiantes iberoamericanos.
3. **Preprocesar** el texto mediante la eliminación de *stopwords*.

Si bien el análisis de sentimientos no se automatizó programáticamente, el análisis manual realizado permitió identificar cómo los sufijos y prefijos en palabras como "risita", "gritito" o "pesadez" son cruciales para transmitir la intención y la emoción del texto, lo cual es vital para la correcta comprensión.

Este trabajo sirve como un prototipo funcional que valida el uso del análisis léxico como una herramienta potente para la inclusión educativa. Demuestra que la tecnología de compiladores puede ser un puente efectivo para superar las barreras dialectales, facilitando la creación de experiencias de aprendizaje diferenciadas en contextos interculturales.

## Referencias

- Aho, A. V., Ravi, S. A. V., Ullman, J. D. (1998). *Compiladores: Principios, técnicas y herramientas*. Addison Wesley Longman.
- Giró, J., Vázquez, J., Meloni, B., Constable, L. (2015). *Lenguajes formales y teoría de autómatas*. Editorial Alfaomega. Argentina.
- Gutú, O. (2013). Primer curso en teoría de autómatas y lenguajes formales. Pearson Educación. México.
- Hopcroft, J. E. Motwani, R. & Ullman, J. D. (2007). Introducción a la teoría de autómatas, lenguajes y computación. Pearson Educación. México.
- ¿Qué es el análisis de sentimiento? Guía técnica exhaustiva. (2025, 10 noviembre). Elastic. <https://www.elastic.co/es/what-is/sentiment-analysis>
- Ibm. (2023, 26 diciembre). Análisis de sentimientos. ¿Qué es el análisis de sentimientos? <https://www.ibm.com/mx-es/think/topics/sentiment-analysis>

## Anexo A. Documento fuente que incluye regionalismos propios de Colombia.

### “La noche en que se me subió la bruja”

*Leyenda contada por Ricardo Henao*

Vea pues, mijo, siéntese y pídase un **tintico**, que le voy a echar un cuento que no me lo invento yo, eso me pasó de verdad, aquí mismito en **Santa Fe de Antioquia**, una noche de esas en que el calor se pega a la piel y ni el abanico da tregua. Yo vivía en la casa vieja de mi abuela, una casa de **tapia** y teja, con un patio lleno de matas de ruda, sábila y albahaca, y con ese olor a tierra caliente que uno no olvida ni queriendo.

Esa noche, después de haberme tomado unos **guarapos** con los compas en la esquina, llegué medio cansado, con ganas de caer redondo y de mirar pa’ dentro. El pueblo estaba callado, sólo se oía el canto de los grillos y un perro que ladraba a lo lejos, como si hubiera visto al mismísimo diablo. *Me acosté, rezando de carrerita el Padre Nuestro* —por si las moscas—, y cerré los ojos. Pero vea, no habían pasado ni cinco minutos cuando sentí una pesadez en el pecho, como si me hubieran echado una piedra encima. Quise moverme, pero ni pío pude. Ni un dedo. Ahí fue cuando me di cuenta de que **se me había subido la bruja**.

Hermano, yo sentía el aire helado en el cuello, y ese silencio raro, como si el mundo se hubiera quedado sin sonido. Quise gritar, pero la voz no me salía. De reojo, vi una sombra encima mío, con una risita chiquita, burlona, y un olor a azufre mezclado con perfume barato. Yo pensé: “*Ave María Purísima, esto no es sueño, esto es brujería* del *monte*”.

Mi abuela siempre decía: “*Cuando le pase eso, diga: mañana venga por sal, que ahí se le quita la **maña***”. Pero dígame usted eso a alguien que no puede ni respirar, ¡qué jartera! Me tocó hacer fuerza con el alma. En eso, sentí que la sombra se me pegó más al pecho, y me zumbó en el oído algo que no entendí bien, pero que me

erizó hasta el alma. Entonces, como pude, solté un **grítico** ahogado: —¡Mañana venga por sal!

Y vea, mijo, fue decir eso y la cosa esa pegó un brinco, como si le hubieran echado agua bendita. Sentí un **ventarrón** por la **pieza** y un alarido que hizo temblar las tejas. Me quedé quieto, empapado en sudor, temblando como un flan. Cuando por fin pude moverme, me levanté y encendí la luz, pero no había nadie, sólo el olor raro y un mechón de pelo negro en la almohada. ¡Ave María! Más asustado que gallina en patio ajeno, eso sí.

Al otro día amanecí con los ojos como un sapo, sin dormir un **ratito**. Pero vea lo más **berraco**: como a las nueve de la mañana, tocaron la puerta. Yo salí todavía medio zombi, y ahí estaba una señora del barrio, una que casi nunca hablaba con nadie.

—Vecino —me dijo con esa vocecita empalagosa—, ¿no me regala un poquito de sal?

Y vea, mijo, yo me quedé frío. Sentí que se me heló la sangre. Era ella, la mismita. Le pasé la sal temblando y le dije: —Tenga, vecina, llévese toda si quiere, pero no vuelva por acá ni en sueños. Ella soltó una risita, de esas que uno no sabe si son de burla o de agradecimiento, y se fue despacio, arrastrando las chanclas por la calle empedrada. Desde entonces, cada vez que paso por su casa, los pelos se me paran como alambre pelado.

Dicen los viejos que esas brujas se suben por envidia, o porque uno anda muy **sabrosón**, creyéndose el cuento. Que, si lo **pillan** desprevenido, lo **enjalman** como mula y no lo sueltan hasta que uno diga la frase mágica. Y vea, no sé si será verdad o no, pero desde esa noche duermo con un escapulario en el cuello, una rama de ruda bajo la almohada y la luz prendida. Por si las moscas, pues.

Y si le soy sincero, uno se ríe ahora, pero esa noche casi me da un patatús. Dicen que “el que no ha visto el diablo, no cree en él”, y yo le digo: a mí no me lo contaron, a mí me pasó.

Aunque bueno, no hay mal que por bien no venga, porque desde ese día dejé de trasnocharme en la esquina y de beber tanto **guarapo**, que eso, dicen, atrae los malos vientos.

Así que ya sabe, mijo: si alguna vez **se le sube la bruja**, no se me quede callado ni tieso, que ahí sí fue. Saque fuerzas del alma y grite con toda la **verraquera**:

**—¡Mañana vengas por mí!**

Y si al otro día le tocan la puerta... pues, no diga que no le advertí.



## Anexo B. Evidencias de la comunicación que se obtuvo con nuestro par Colombiano

El primer acercamiento que se tuvo fue por medio de correo electrónico, donde se le contacto para saber la disponibilidad que tenía para poder establecer una conversación y platicar acerca de los regionalismos de Colombia, la respuesta que se obtuvo fue rápida y mostro accesibilidad. Véase Figura 7.

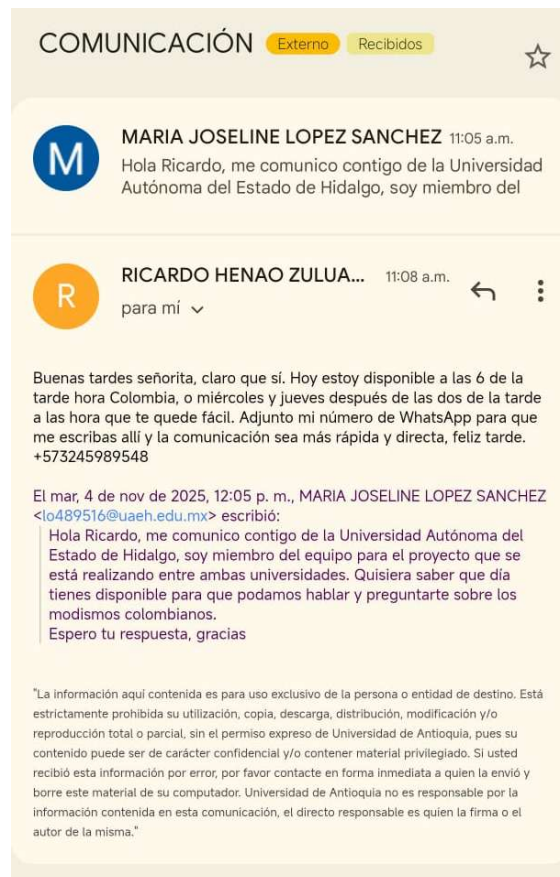


Figura 7. Captura de pantalla de la conversación que se tuvo a través de correo electrónico

Ese mismo día se siguió la comunicación vía WhatsApp donde se acordó el día y la hora (Figura 8) para posteriormente realizar una llamada donde se encontró todo el equipo (Figura 9)

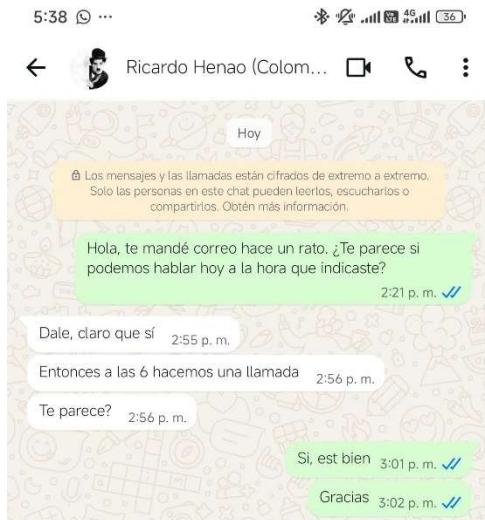


Figura 8. Conversación via WhatsApp donde se acordó horario

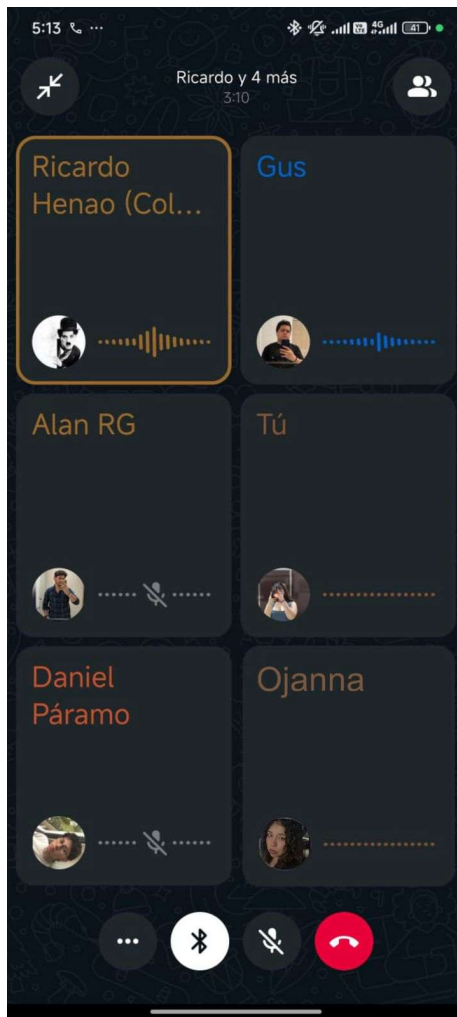


Figura 9. Captura de llamada via WhatsApp con todo el equipo presente

Finalmente, se intento volver a tener comunicación con el compañero colombiano con el fin de mostrarle la traducción del texto y poder obtener una opinión y retroalimentación de su parte, sin embargo, no se obtuvo respuesta. (Figura 10).

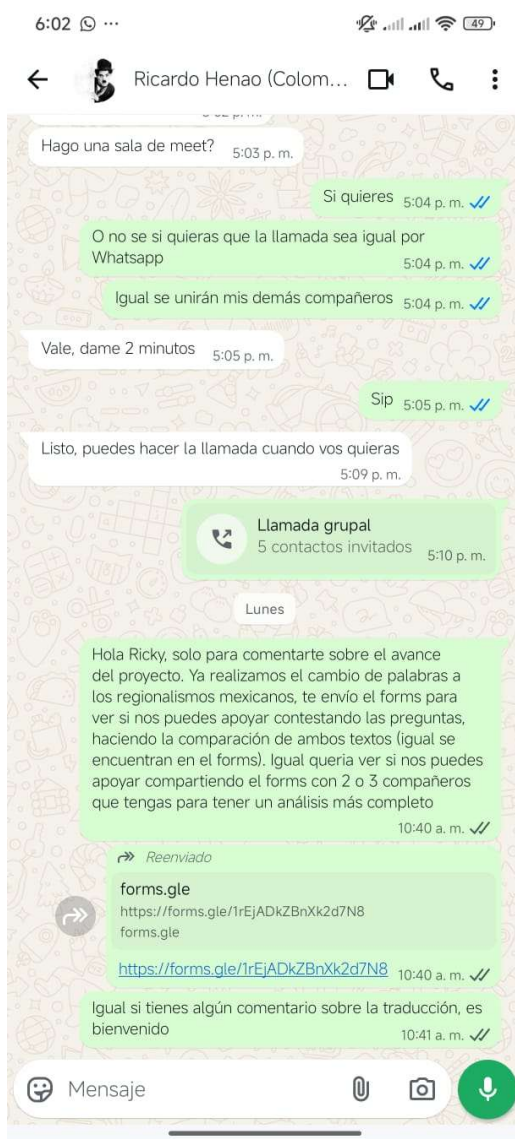


Figura 10. Ultimo mensaje enviado sin respuesta