



Análisis difuso para el cálculo de sentimientos de publicaciones en redes sociales.

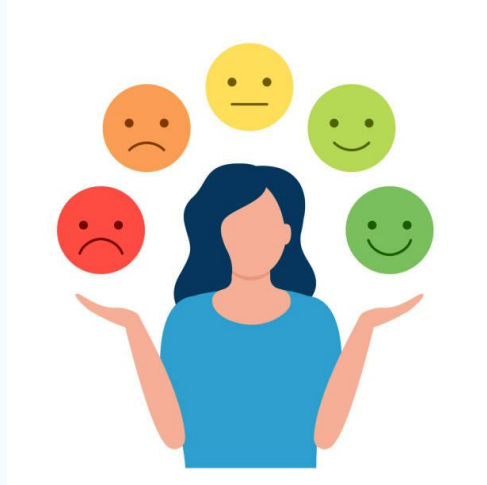
Autores:

- Marcos Raúl Flores Duarte
- Elías Sebastián Gill Quintana

Contenido

1. Introducción
2. Preliminares
3. Arquitectura del Sistema
4. Metodología
 - 4.1. Lector de Datasets
 - 4.2. Lexicón de Sentimientos
 - 4.3. Fusificación
 - 4.4. Sistema de inferencia
 - 4.5. Defusificador
 - 4.6. Benchmarks
5. Resultados
6. Conclusiones

Introducción

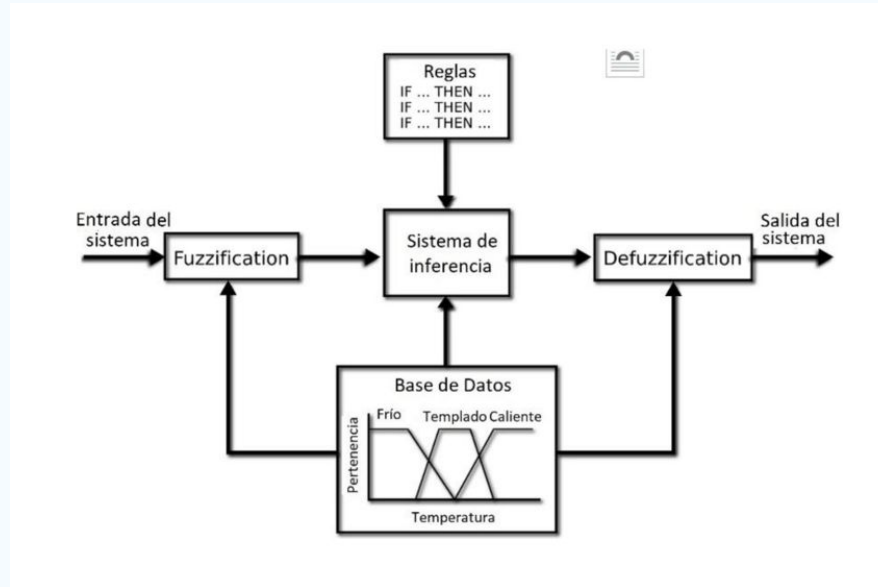


Sentimientos de un
Tweet



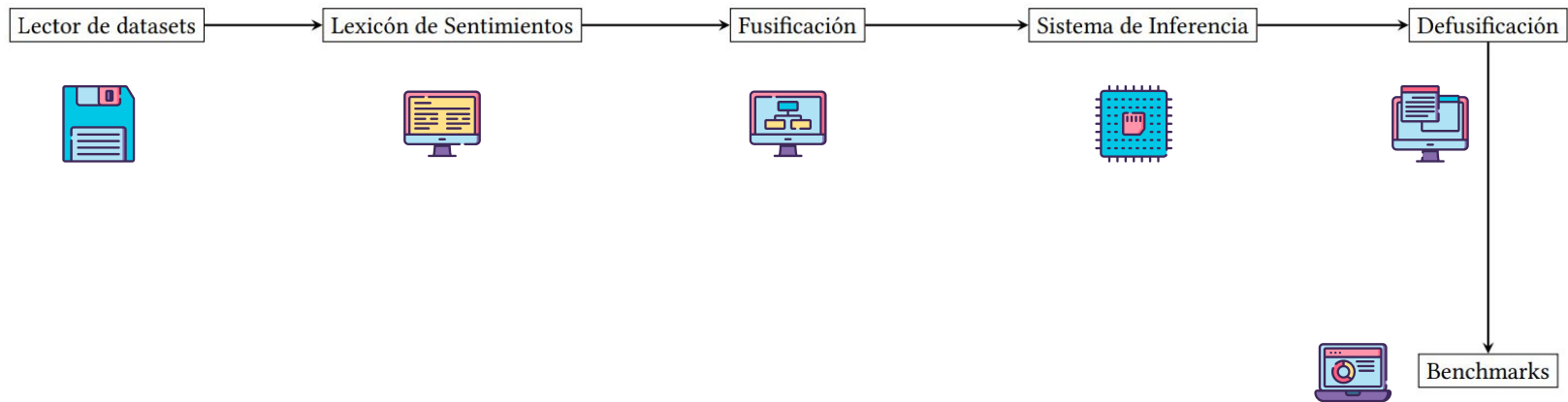
Red Social Twitter/X

Preliminares de lógica difusa



Ejemplo de sistema de control Difuso

Arquitectura del sistema



Módulos del sistema

Metodología

Lector de Datasets

SemEval2017	SemEval2013	Sentiment140
11,977 tweets	3,545 tweets	359 tweets

Metodología

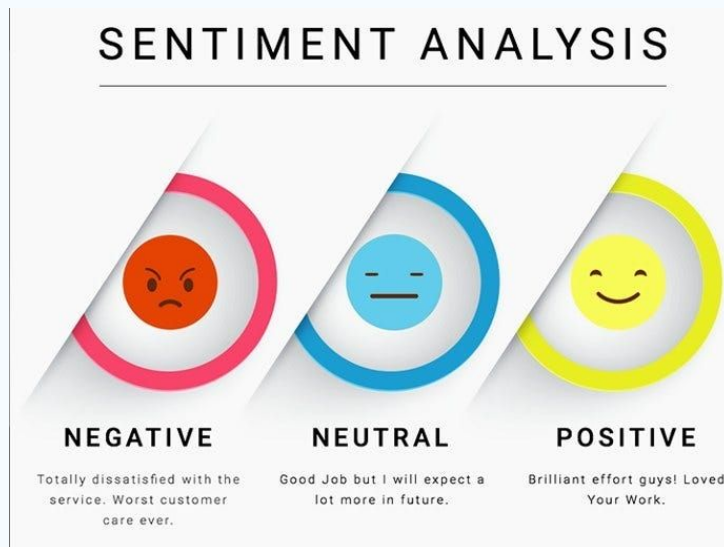
Lector de Datasets - Procesamiento del texto

```
# Preprocesamiento del texto y limpieza de los datos

def generar_puntuaciones(tweet) -> tuple[float, float, float]:
    # limpieza del tweet
    tweet = re.sub(r"won't", "will not", tweet)
    tweet = re.sub(r"can't", "can not", tweet)
    tweet = re.sub(r"@", "", tweet) # removal of @
    tweet = re.sub(r"http\S+", "", tweet) # removal of URLs
    tweet = re.sub(r"#", "", tweet) # hashtag processing
    tweet = re.sub(r"n't", " not", tweet)
    tweet = re.sub(r"\ 're", " are", tweet)
    tweet = re.sub(r"\ 's", " is", tweet)
    tweet = re.sub(r"\ 'd", " would", tweet)
    tweet = re.sub(r"\ 'll", " will", tweet)
    tweet = re.sub(r"\ 't", " not", tweet)
    tweet = re.sub(r"\ 've", " have", tweet)
    tweet = re.sub(r"\ 'm", " am", tweet)
```

Metodología

Lexicón de Sentimientos

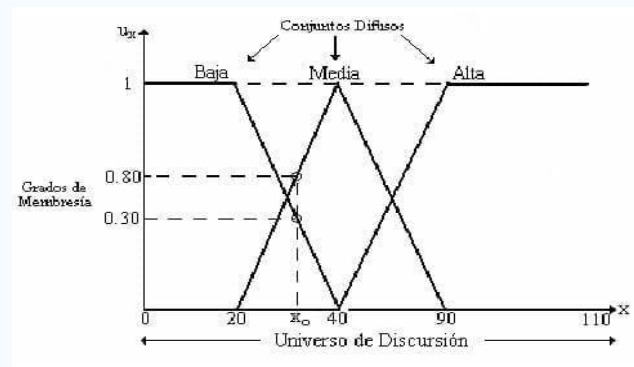


Metodología

Fusificación

Transformar los datos "crisp" a valores de pertenencia. Se aplica funciones triangulares a las métricas de sentimiento proporcionadas por VADER.

$$\mu(x) = \begin{cases} 0, & \text{si } x \leq a \text{ o } x \geq c, \\ \frac{x-a}{b-a}, & \text{si } a \leq x < b, \\ \frac{c-x}{c-b}, & \text{si } b \leq x < c. \end{cases}$$



Metodología

Sistema de Inferencia Difusa

Aplicacion de la reglas de Mamdani y agregación de los resultados.

Regla	Nivel de Positividad	Nivel de Negatividad	Resultado
1	Bajo	Bajo	Neutral
2	Medio	Bajo	Positivo
3	Alto	Bajo	Positivo
4	Bajo	Medio	Negativo
5	Medio	Medio	Neutral
6	Alto	Medio	Positivo
7	Bajo	Alto	Negativo
8	Medio	Alto	Negativo
9	Alto	Alto	Neutral

Table 1: Reglas difusas definidas en el sistema.

Metodología

Defusificador

Aplicación del método del centroide para el cálculo del sentimiento final.

```
if 0 < (res_defuzz) < 3.33: # R
    sent_calculado = "Negativa"

elif 3.34 < (res_defuzz) < 6.66:
    sent_calculado = "Neutra"

elif 6.67 < (res_defuzz) < 10:
    sent_calculado = "Positiva"

sentimientos_calculados.append(sent_calculado)
```

Metodología

Benchmarks

- Tiempo de ejecución total
- Tiempo promedio de procesado de un tweet
- Precision del modelo

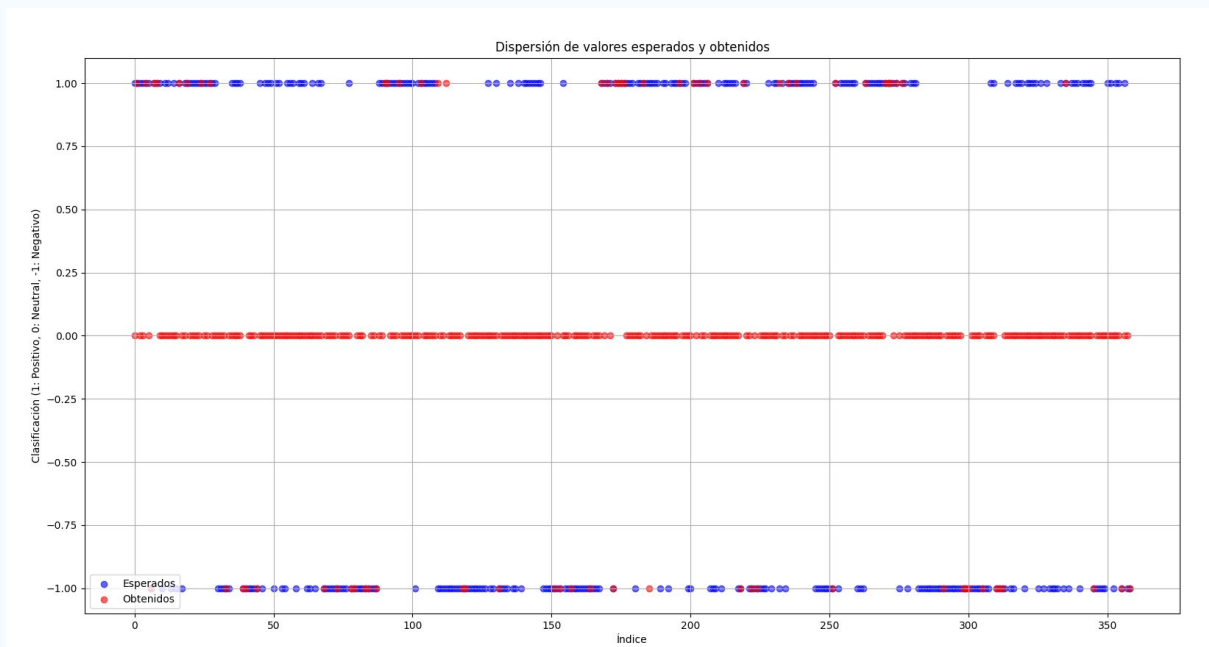
Resultados

Comparación de rendimiento

	Sentiment140	SemEval2017	SemEval2013
Tweets	359	11.977	3.545
Tiempo total	0.221s	7.2s	2.302s
Promedio por tweet	0.000617	0.000649	0.000601
Precision	19%	45%	52%

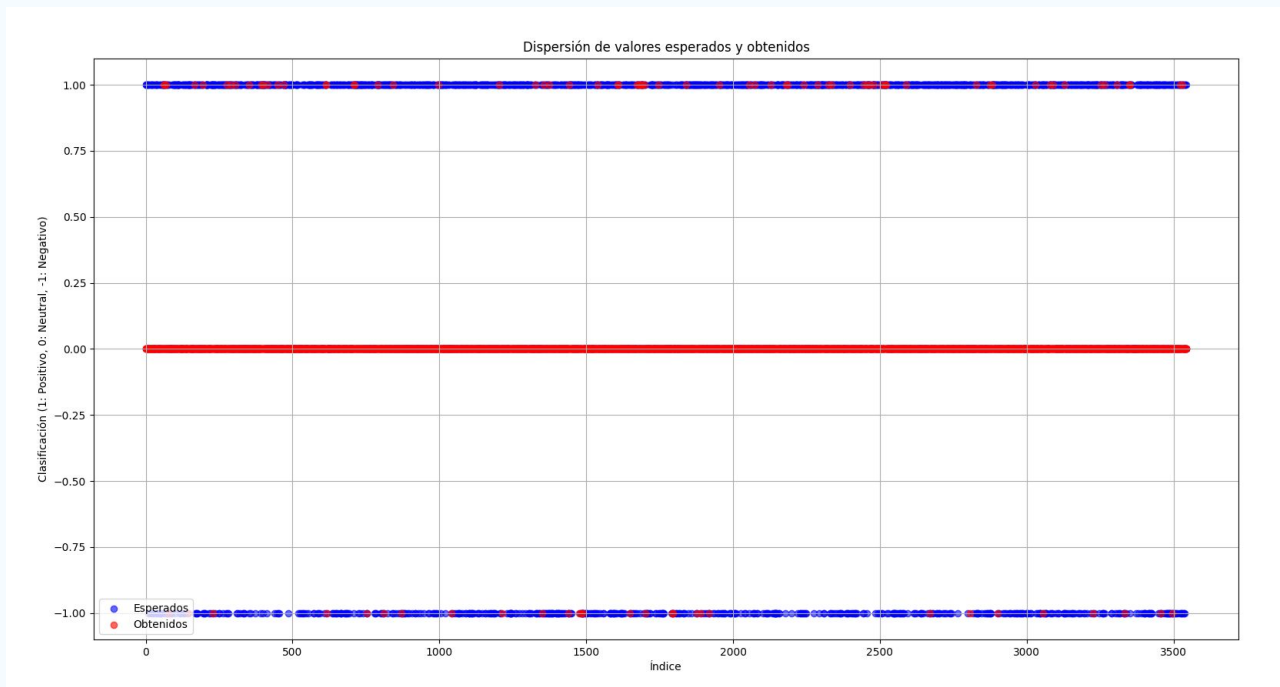
Resultados

Sentiment140 Test-Data



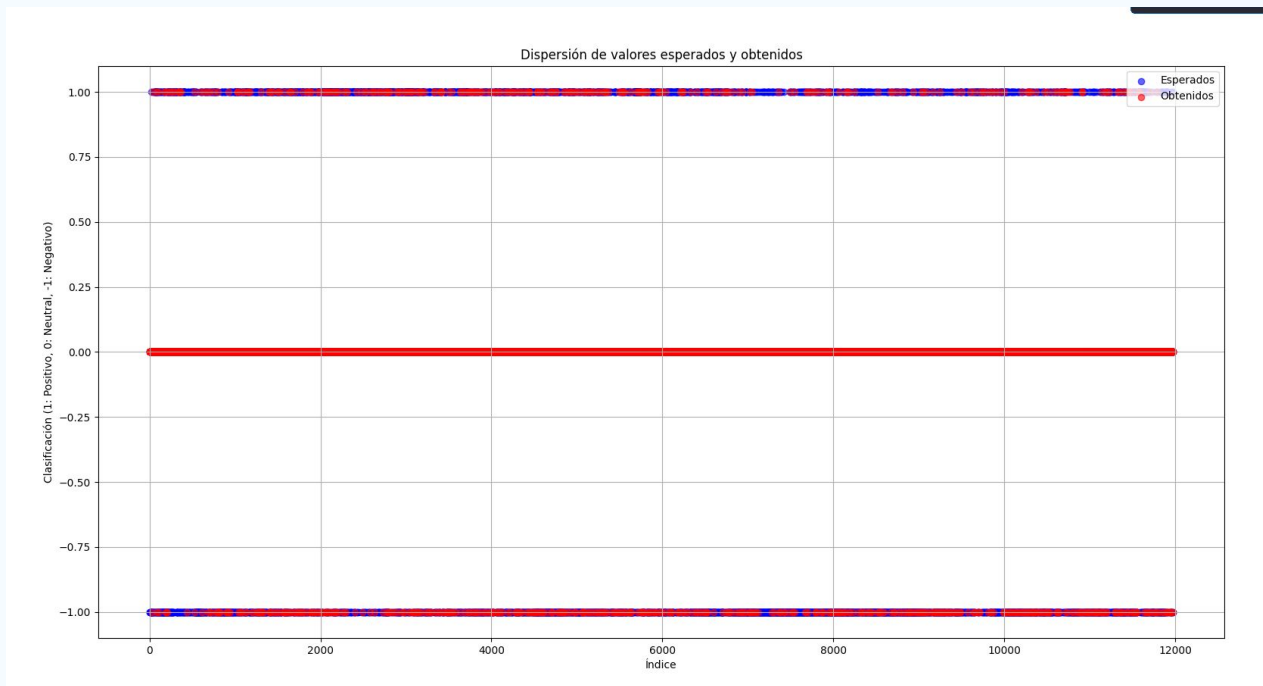
Resultados

SemEval 2013 Test-Data



Resultados

SemEval 2017 Test-Data

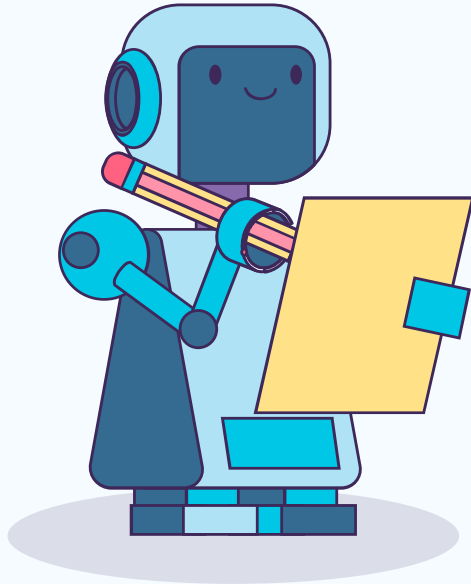


Conclusiones

- El sistema de análisis de sentimientos basado en lógica difusa demostró ser eficaz para interpretar emociones en redes sociales, manejando incertidumbre y ambigüedad del lenguaje.
- Se alcanzó una precisión del 52% en la detección del tono emocional.
- Ventajas destacadas: rapidez en el análisis y capacidad para procesar grandes volúmenes de datos.
- Oportunidades de mejora: optimizar el sistema de inferencia para aumentar la precisión en futuros desarrollos.



¡Gracias!



Este trabajo forma parte del proyecto de examen final para la materia de Matemática Aplicada, impartida por el profesor Marcos Villagra en la Facultad Politécnica de la Universidad Nacional de Asunción.