



Teo Manuel Kaucher
tkaucher@udesa.edu.ar

Camila Guerrero
cguerrero@udesa.edu.ar

Bryan Drexler
bdrexler@udesa.edu.ar

Lucio Luque Materazzi
lluquematerazzi@udesa.edu.ar

INTERACCIÓN Y ARQUITECTURA

Con solo dos textos describiendo tu emoción actual y la emoción que querés sentir, la aplicación genera una playlist que acompaña esa transición entre sentimientos.



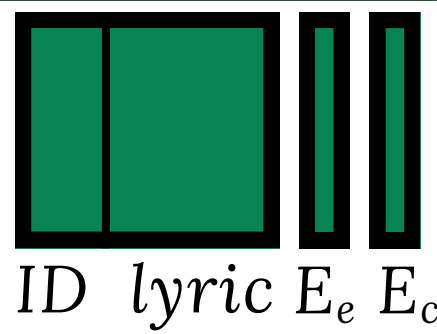
GPT 4o-mini: usado para comunicación fluida, traducción (para uso en cualquier idioma) y reescritura de inputs (desde “me quiero sentir bien” a “me siento bien” para eliminar errores de clasificación en roBERTa y E5).

RoBERTa (preentrenamiento con MLM, fine-tuning sobre GoEmotions): clasifica multi-label de emociones (28 etiquetas), se elimina la etiqueta “neutral”. Obtenido de Huggingface, por SamLowe. Utilizamos sus logits como embeddings emocionales de los inputs y canciones.

Multilingual E5 base (preentrenamiento para emparejar textos, fine-tuning sobre texto clasificado): genera embeddings contextuales de tamaño 768. Obtenido de HuggingFace, por intfloat.

SELECCIÓN DE CANCIÓN

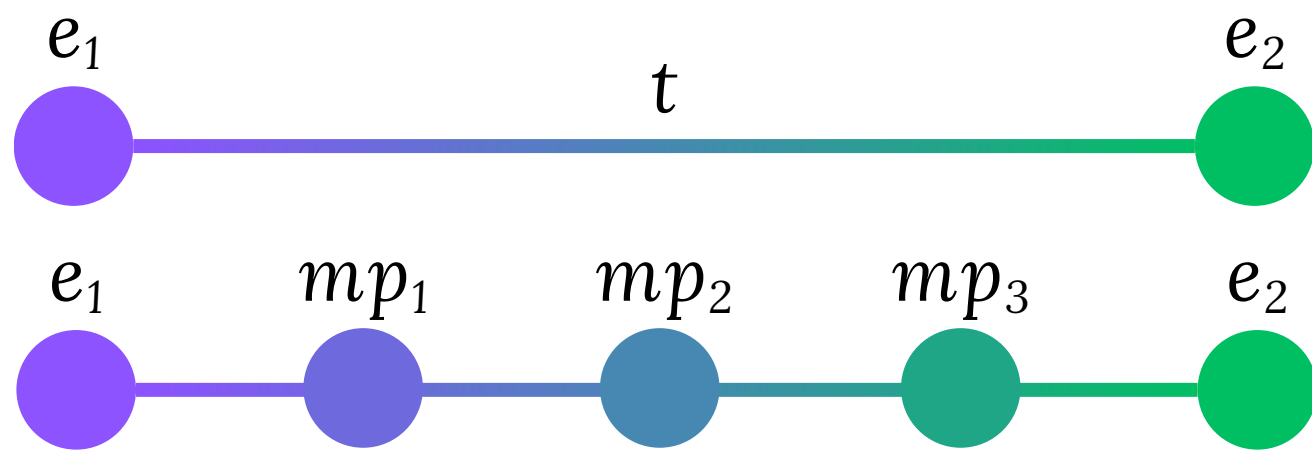
Se parte de un dataset de 11 columnas para 480k canciones, obtenido de Huggingface por brunokreiner. Se utilizan únicamente los campos de ID de spotify y letra.



En una etapa offline se utilizan roBERTa y E5 para calcular un embedding emocional e_{emo} y un embedding contextual e_{cont} para todas las canciones, generando matrices E_{emo} y E_{cont} .

En la etapa online el procedimiento es el siguiente:

- Con roBERTa se calculan embeddings emocionales para los dos inputs del usuario, e_1 y e_2 .
- Se calcula $e_2 - e_1 = t$, definiendo el **vector de transición t**.
- La transición se divide en k **mid-points** (mp) con el siguiente cálculo: $mp_i = e_1 + \frac{t \cdot i}{(k-1)}$
- En cada mp se calcula la similitud coseno contra los embeddings de todas las canciones, calculados en el paso offline: $sim_{emotion}[i] = \frac{mp_i \cdot E_{emo}}{\|mp_i\| \cdot \|E_{emo}\|}$
- Repetir pasos 1-4 para contexto con el modelo E5
- Para cada mp se eligen las n canciones de mayor similitud ponderada entre contexto y emoción con la fórmula $c_i = argmax_c (w_1 \cdot sim_{context}[i] + w_2 \cdot sim_{emotion}[i])$
- Con todas las canciones seleccionadas se arma la playlist de transición emocional.



Los hiperparámetros actuales llevan los siguientes valores: $k=7, n=1, w_1=w_2=0.5$
En la siguiente sección se presentan estudios sobre la relación entre el desempeño de la aplicación y los pesos $w_{1,2}$

EVALUACIÓN

1- Clasificación de sentimiento en prompts

Dataset₁: Generamos 4 prompts etiquetados manualmente por cada una de las 27 emociones, resultando en 108 datos para testear.

Setup: Para cada prompt del dataset₁ predijimos con roBERTa la emoción más fuerte y la comparamos con la emoción etiquetada.

Mejores 5 resultados

	F1-Score	Recall	Precision
Confusion	1.00	1.00	1.00
Disgust	0.86	0.75	1.00
Desire	0.80	1.00	0.67
Optimism	0.75	0.75	0.75
Amusement	0.67	0.75	0.6

Peores 5 resultados

Sadness	0.33	0.50	0.25
Annoyance	0.29	0.25	0.33
Grief	0.00	0.00	0.00
Nervousness	0.00	0.00	0.00
Relief	0.00	0.00	0.00

El modelo no detectó ningún caso de duelo, nerviosismo o alivio. Sin embargo, las predicciones más comunes de esos casos son grief:sadness, nervousness:fear y relief:joy.

2- Clasificación de sentimiento en canciones

Dataset₂: Se seleccionaron 100 canciones del dataset original y se las etiquetó manualmente con las dos emociones más predominantes.

Setup: Para cada canción del dataset₂ predijimos con roBERTa las dos emociones más fuertes y las comparamos con las emociones etiquetadas.

Mejores 5 resultados

	F1-Score	Recall	Precision
Confusion	0.67	0.67	0.67
Love	0.57	0.63	0.53
Disgust	0.50	0.33	1.00
Gratitude	0.50	0.33	1.00
Sadness	0.46	0.42	0.50

Peores 5 resultados

Grief	0.00	0.00	0.00
Fear	0.00	0.00	0.00
Embarrassment	0.00	0.00	0.00
Curiosity	0.00	0.00	0.00
Anger	0.00	0.00	0.00

El modelo no logra detectar ninguna emoción de las peores 5 en el conjunto de test. Sin embargo, las predicciones más comunes de esos casos son grief:sadness, fear:annoyance, embarrassment:remorse, curiosity:realization, anger:annoyance/disapproval

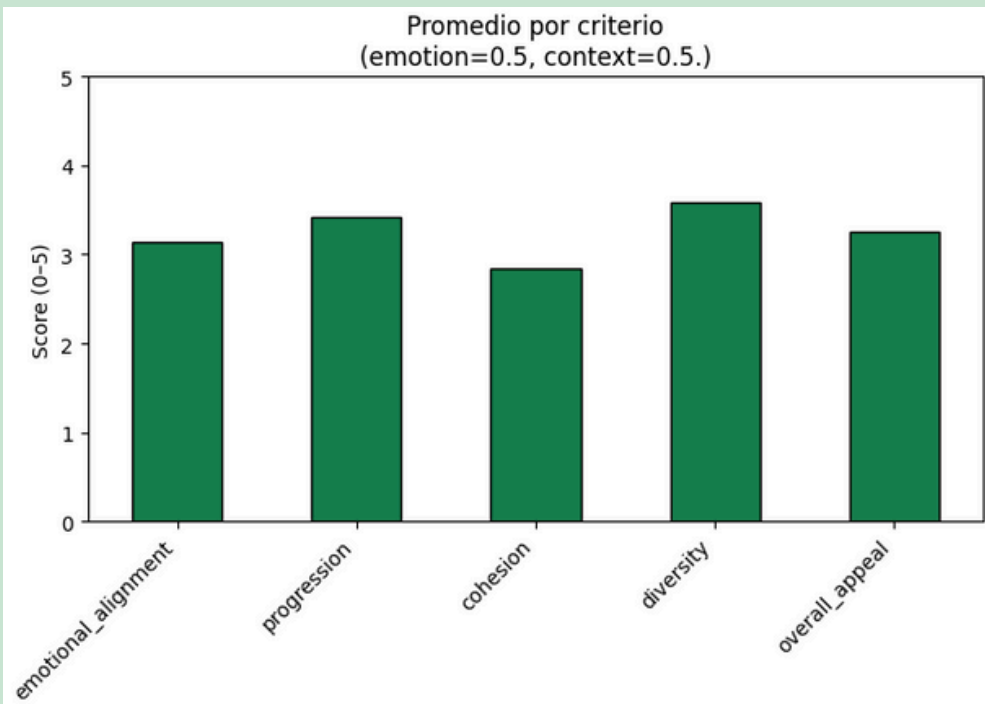
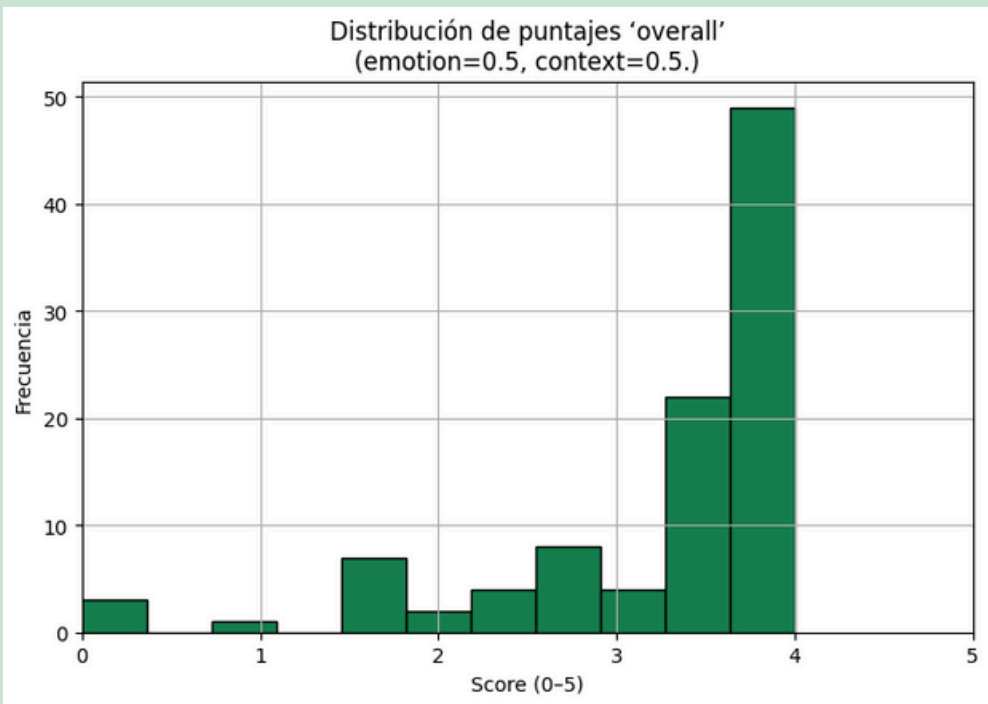
3- Generación de playlist

Dataset₃: Se generaron 100 combinaciones de los inputs de los usuarios con la ayuda de GPT-4 y se generó la playlist para cada combinación.

Setup: Le pedimos a un LLM Judge (GPT 4) que para cada par de inputs evalúe su playlist asociada desde 1 (peor) a 5 (mejor) en campos de alineación emocional, progreso emocional, cohesión, diversidad y un puntaje general.

W1/W2	emotional_alignment	progression	overall_appeal
0.25/0.75	3.35	3.50	3.40
0.5/0.5	3.40	3.60	3.35
0.75/0.25	3.00	3.15	3.05

Al repetir sobre 20 datos para distintas configuraciones de pesos, encontramos que 0.5 maximiza las métricas más relevantes.



4- Formulario completado por usuario al probar la app

	1	2	3	4	5	6	7	8	9	10	11	12	13
Promedio	4.2	3.8	4.4	3.8	3.0	3.0	3.2	4.2	3.4	4.4	3.2	4.2	3.6
Usuario 1	4	4	5	4	3	2	4	5	5	5	4	5	4
Usuario 2	4	3	3	2	4	5	3	4	2	4	3	4	3
Usuario 3	4	3	4	4	2	1	3	4	3	4	2	3	3
Usuario 4	5	5	5	5	3	5	2	3	2	4	3	4	4
Usuario 5	4	4	5	4	3	2	4	5	5	5	4	5	4



LIMITACIONES

La principal limitación fue la falta de datasets con información completa (género, valencia, activación, dominancia, etc), por lo que la recomendación se basó únicamente en la letra, atributo presente en datasets más amplios. Debido a eso, no se pudo personalizar según los géneros más escuchados por el usuario, a pesar de la posibilidad de acceder a esa información.

Se emplea un modelo de detección de emociones, pero no se llegó a realizar fine-tuning para poder generar embeddings emocionales, lo cual podría haber mejorado su desempeño. Por último, la playlist generada se limita actualmente a Spotify sin soporte para otras plataformas.