Detecting Sentiment Discrepancies in Spanish/English translations using NLP and LLMs

Duran Rondon, Maria[1], Peralta-Martinez, Ixchel[1], and Whitener, Brittany[1]

[1]M.S Data Science, University of North Carolina at Wilmington

April 30, 2024

Abstract

In this project, we explore the application of Natural Language Processing (NLP) and Large Language Models (LLMs) to identify discrepancies between original texts and their translations from Spanish to English, and vice versa. We employ two translation models—GPT-4 and Deep Translator[2]—and two types of sentiment analysis models, pysentimiento [19] (Sentiment Classification) and DistilBERT [20] (Multi-faceted Analysis of Subjective Texts, MAST), focusing on detecting misinterpretations, particularly prevalent in idiomatic expressions. As bilingual speakers, we have observed frequent inaccuracies in translation that significantly impact the conveyance of the intended message. By constructing unique datasets from a variety of sources, including idioms and podcast transcripts, we assess the performance differences between sentiment analysis models and translation tools. Our goal is to make users aware of potential sentiment mismatches, thus empowering them to critically verify and interpret translation outcomes. This project underscores the importance of using NLP and LLMs in conjunction with human feedback in a language translation review context.

Keywords
pysentimiento, Sentiment Classification, DistilBERT, Multifaceted Analysis of Subjective Texts, Chatgpt-4, Deeptranslator, NLP, LLM

1 Introduction

Sentiment analysis, which leverages the capabilities of Natural Language Processing (NLP) and machine learning, is crucial in various sectors for tasks such as monitoring brand reputation, improving customer support, conducting market research, and detecting biases in AI technologies [14]. This technique analyzes textual data to determine the emotional tone behind a message, which is vital for understanding consumer responses and improving service delivery[22].

Accurate translation is essential for maintaining the sentiment and intent of the original message. Translation errors can lead to humorous or even drastic changes in the intended meaning [9], which can have significant consequences for brands or user comprehension. For example, translation mistakes such as a care label that says "Hand is washing" in Spanish instead of "Hand wash only," or a mistranslated to Spanish "turkey" referring to the country rather than the bird [21], can cause confusion, amusement, or offense. In advertising, such errors may reduce the impact of a campaign by altering the perceived message.

To address these challenges, our application of sentiment analysis focuses on identifying discrepancies between Spanish and English translations. By using sentiment analysis scores, our tool flags potential mismatches, alerting users to possible misunderstandings and enabling them to critically evaluate the sentiment conveyed in translations. This proactive strategy helps avoid translation errors and continuously improves models tailored for specific applications in both corporate and individual settings.

The second part of our solution is a GPT-4 powered assistant [18] for translating and interpreting slang between Latin American Spanish and American English. This capability allows users to effectively understand and compare slang expressions across languages, enhancing cross-cultural communication.

2 Methodology

2.1 Data Collection

2.1.1 Idioms Dataset

The idioms dataset was created using resources from different web pages[10][11][3][15][4][16][5], and the data was collected through web scraping. This technique was employed to extract English and Spanish idioms with their meanings from various websites using Python libraries such as Requests, BeautifulSoup, and Pandas.

Web scraping is a technique for extracting data from websites through programmatic access to web pages. The Requests Library is used to make HTTP requests to a specified URL, manage the request to the server, receive the response, and access the response data, which is crucial for fetching the HTML content of the web page for data extraction. BeautifulSoup was used to parse the HTML content retrieved via Requests. This library facilitates easy data extraction by allowing searches through HTML elements using tags, attributes, and navigational strings. Lastly, Pandas is used for data manipulation and analysis, helping to organize the scraped data into a structured DataFrame format, which can then be exported to a CSV file for further cleaning.

In the process of cleaning and organizing CSV files into structured data frames for analysis, several steps were taken using Python and the Pandas library. Initially, CSV files are read and loaded into data frames, ensuring any missing headers are correctly defined. Rows containing only white spaces or null values are removed to maintain data integrity. For translation and reformatting purposes, specific columns in the data frames are renamed for clarity, such as translating or relabeling columns to accurately reflect their contents. Additionally, text data is stripped of leading and trailing spaces, and any duplicate entries are eliminated to ensure uniqueness. To better organize the data, columns are inspected and reformatted, and data frames are checked for any remaining null or NaN values.

2.1.2 Transcripts Datasets

After randomly selecting 10 Spanish [13] [17] and 10 English podcasts [1] [12], we chunked their transcripts into separate lines of a .csv file, split by each incident of speaking. To accomplish this, we leveraged the formatting of the transcripts to write a function that identified the appropriate speaking lines.

Due to the regularity of lines in each transcript, we were able to split the text along new lines before eliminating the lines that did not contain a speaking part. In the example below, the transcript has a period of 4, with speaking incidents occurring on the line with index 2 within each cycle (starting from index 0).

Example_transcript = """ Mariana 00:04 My name is Mariana. Jeffrey 00:06 I'm Jeffrey. Mariana 00:10 A few months ago, a documentary crew started following me and a few other trans friends around, navigating everything from family, relationships, work, and more." """ After determining the period ('line_freq') and the relevant row index ('line_hit') for each podcast, the following function was applied to each using a for loop with the corresponding information for each transcript.

def eng_transcript_to_csv(transcript, transcript_name, line_freq, line_hit)

This function output 20 .csv files, each corresponding to one of the 20 podcasts, containing only the speaking portions from each podcast, with each speaking incident as a string.

2.2 Data Creation: Translations

2.2.1 Deep Translator

To translate text directly in Python, we employed the deep-translator Python library, which aggregates various free and unlimited translation services. This enables translations between multiple languages through diverse translators such as MyMemory Translator, DeepL Translator, PONS Translator, Microsoft Translator, and Google Translator. This toolkit supports batch translation, which enhances its utility. For our use case, Google Translator was selected due to its extensive language support and massive daily translation volume, which exceeds 100 billion words. Google Neural Machine Translation (GNMT) [6] leverages deep learning models, specifically a type of artificial neural network, to translate whole sentences at once rather than segment by segment. This approach allows the system to grasp the broader context of a sentence, thereby enhancing the accuracy and naturalness of the translations.

We imported GoogleTranslator [7] from deep-translator to translate from English to Spanish and vice versa, setting the parameters source='en' for English and target='es' for Spanish. Then, a for loop was used to iterate over all the transcript CSV files and produce a dataframe with two columns: 'original text' and 'translated text'. Along the way, we learned that the Google Translator package was limited to 5000-word strings. Fortunately, there was only one speaking line that exceeded this limit, so it was dropped. Afterward, the results

were compiled into two dataframes: one for all the Spanish-to-English text and another for all the English-to-Spanish text.

2.2.2 ChatGPT4

To integrate GPT-4's [8] multimodal capabilities, which include understanding and generating both text and images, into a practical application, we developed a plugin leveraging these features. The core training dataset for GPT-4 comprised a varied assortment of publicly available text, licensed data, and contributions from human trainers. The model underwent a training regimen using a technique known as reinforcement learning from human feedback (RLHF). This approach fine-tunes the model's responses through iterative enhancements driven by human evaluative feedback, aligning the outputs with human values and preferences.

For our specific use case, we created a plugin leveraging GPT-4's understanding capabilities and fine-tuning parameters. The development involved fine-tuning GPT-4 using customized prompts and few-shot learning techniques, where the idioms dataset served as input with idioms as prompts and their meanings as expected outputs. This approach enhanced the plugin's translation accuracy and adaptability, enabling effective handling of nuanced language in real-world scenarios. However, the plugin faced a limitation: it could only process ten rows of data at a time. This constraint necessitated batching the input data to ensure efficient processing. Additional features were incorporated to promote more natural dialogue generation and extend the plugin's functionalities.

The plugin was prompted with the phrases "Translate each pair of strings to English" or "Translate each pair of strings to Spanish" to translate individual rows of the podcast CSV files. The translated outputs were then compiled back into the podcast CSV files, ensuring the preservation of the original content alongside the translated text.

2.3 Sentiment Analysis

Sentiment analysis, a subfield of Natural Language Processing (NLP), is concerned with identifying and categorizing opinions expressed in text in order to determine the sentiments toward specific subjects. This process helps in understanding the emotional tone behind words, enabling businesses and researchers to gather insights from vast amounts of data efficiently. The following are core types of sentiment analysis:

- **Sentiment Classification (SC):** This basic form of sentiment analysis involves determining the overall sentiment polarity of a text, categorizing it as positive, negative, or neutral.
- **Aspect-Based Sentiment Analysis (ABSA):** Rather than giving an overall sentiment score, ABSA focuses on analyzing sentiments related to specific aspects of a subject within the text.
- **Multifaceted Analysis of Subjective Texts (MAST):** This advanced type involves exploring complex sentiment phenomena, including irony, hate speech, and mixed emotions, which provide deeper insights into the contextual use of language.

2.3.1 MAST: DistilBert

Model Overview DistilBert, a distilled version of the BERT model, retains 97% of its language understanding capabilities while being smaller and faster. This model is fine-tuned for detecting various emotions in texts, such as sadness, joy, love, anger, and surprise.

Training and Performance DistilBert was fine-tuned on an emotion dataset derived from Twitter, achieving an impressive accuracy of 93.8%. It processes approximately 399 text samples per second, showcasing its efficiency in real-time applications.

Practical Usage The model is employed using the Hugging Face pipeline for text classification, where it excels in predicting emotions from text. It is particularly effective in returning comprehensive scores for detected emotions, allowing for nuanced understanding and response generation.

2.3.2 SC: pysentimiento

Model Overview pysentimiento is a Python toolkit that leverages transformer models for various NLP tasks, including sentiment analysis. It provides a multi-class classifier that returns a single variable indicating positive, negative, or neutral sentiments.

Model Performance and Usage For English, pysentimiento uses BERTweet, which has shown a F1 score of 72.0 ± 0.4, while for Spanish, it utilizes RoBERTuito with a F1 score of 70.2 ± 0.2. This demonstrates its robust performance across different languages.

eng_highest_sent	spa_highest_sent
anger	joy
joy	joy
joy	joy
anger	joy
joy	anger

Figure 1: Above: highest sentiments for the first 5 text samples

eng_highest_sent	spa_highest_sent
joy	joy
joy	joy
joy	joy
joy	joy
joy	joy

Figure 2: Left: English to Spanish, Right: Spanish to English

Ease of Integration The toolkit is easily installed via pip and is designed to be user-friendly for both scripts and interactive notebooks. This accessibility makes pysentimiento an ideal choice for developers and researchers looking to integrate sentiment analysis into their applications efficiently.

3 Results

3.1 Comparing the Results from (Sentiment Analysis with DistilBERT)

After the emotion analysis with DistilBERT, the following methods were used to assess the retention of emotion during the translation: comparison of highest-scoring emotion, analysis of absolute differences, visual comparison of box plots, and a deep look at the worst-scoring emotions.

Similar methods were applied to both methods of translation (DeepTranslator and ChatGPT). The results below focus on the DeepTranslator method since the results from ChatGPT were very similar.

3.1.1 Highest-Scoring Emotions by sample

For each text sample, the highest scoring sentiment was determined using the .idmax() function in pandas (sample results below).

Across all 1094 rows, about 50% of the highest emotions matched for English to Spanish translations with DeepTranslator, and about 45% of the highest emotions matched for Spanish to English translations.

3.1.2 Analysis of Absolute Differences

Since our goal was to check if the before/after translation text samples were different, we then computed the difference and absolute difference for each corresponding pair of emotions. Then, the measures in the following table were computed.

	Mean Absolute Difference	Standard Deviation	Minimum Absolute Difference	Maximum Absolute Difference
Sadness	0.101425	0.239387	0.0	0.991329
Joy	0.379767	0.248711	0.0	0.982791
Love	0.035780	0.139531	0.0	0.986873
Anger	0.281196	0.211342	0.0	0.924713
Fear	0.115572	0.216431	0.0	0.989109
Surprise	0.028316	0.137093	0.0	0.985843

Figure 3: English to Spanish

The Mean Absolute Difference (MAD) was highest for Joy and Anger for both collections of text samples (originally Spanish and originally English). So, compared to the other emotions, Joy and Anger tended to have a greater disparity of scores before/after translation, which suggests that the translator did the worst at retaining Joy and Anger sentiments.

Standard deviations for absolute differences were high across all emotions (considering that the absolute difference is always between 0 and 1 for this dataset). This suggests that the spread of the absolute differences is high, indicating that the retention of sentiment is inconsistent: sometimes the difference is close to 0 and other times it is very far from 0.

For each emotion, max/min absolute differences are similar, showing that some text samples had very different scores (almost up to 1.0) within each emotion.

	Mean Absolute Difference	Standard Deviation	Minimum Absolute Difference	Maximum Absolute Difference
Sadness	0.146682	0.288993	0.0	0.985667
Joy	0.395303	0.251140	0.0	0.992608
Love	0.036426	0.140094	0.0	0.989984
Anger	0.306588	0.220373	0.0	0.974665
Fear	0.133996	0.256854	0.0	0.996454
Surprise	0.017766	0.106263	0.0	0.988760

Figure 4: Spanish to English

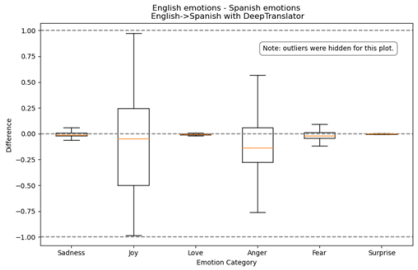

Figure 5: Original English Podcast transcripts - Spanish translations with DeepTranslator

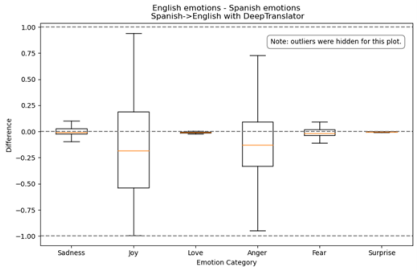

Figure 6: Original Spanish Podcast transcripts - English translations with DeepTranslator

3.1.3 Box Plots of Differences

To further assess the spread of differences, box plots were produced to detect skew within each emotion.

Regardless of whether the original language was Spanish or English, the box plots appeared very similar. Both sets showed that the majority of scores had a difference of less than zero. Since the difference was calculated by subtracting the English score from the Spanish score, a score below zero indicates that the scores were higher in Spanish. A vast spread for Joy and Anger was noted, which aligned with the results for the mean and standard deviation of the absolute differences.

Examination of Anger and Joy Sentiment Discrepancies Given that the sentiments of Anger and Joy exhibited more pronounced differences in comparison to other emotions, a focused investigation was conducted on these scores. This commenced with the analysis of a frequency distribution for the score discrepancies.

3.2 Comparing the Sentiment Analysis Results with pysentimiento from English to Spanish translations

The resulting data is visualized in box plots that highlight the sentiment score differences across three categories: Positive, Neutral, and Negative.

3.2.1 Results from DeepTranslator

The box plots for the DeepTranslator model exhibit a range of score differences across sentiments. For Positive and Negative sentiments, the median score differences are close to zero, suggesting minimal overall change in sentiment detection between the two languages. However, there is a notable variability in the score differences, particularly for the Negative sentiment, where the spread is larger, and the upper quartile extends into positive values.

The most significant observation is in the Neutral category, where the median difference is slightly below zero. This indicates that, on average, translations from English to Spanish using the DeepTranslator model tend to detect less neutrality in content compared to the original English text. The spread for Neutral sentiments is notably narrower compared to the Positive and Negative categories, suggesting a more consistent translation output, albeit with a slight systematic bias towards less neutrality.

3.2.2 Results from GPT-4

The GPT-4 model demonstrates a different pattern in sentiment translation discrepancies. Similar to the DeepTranslator, the median differences for Positive sentiments are close to zero, indicating effective consistency in maintaining the sentiment in translation. However, the Negative sentiment shows a significantly greater spread and a median score difference that suggests a tendency for the Spanish translations to be less negative than the English originals.

Interestingly, the Neutral sentiment analysis reveals a pronounced shift with the median difference positioned at zero but a notable spread towards negative values. This might imply that the GPT-4 translations sometimes interpret neutral English sentiments as slightly more negative in Spanish, contrasting with the other sentiments where the translation effect is less pronounced.

3.2.3 Comparative Analysis

Comparing the two models, both exhibit their strengths and weaknesses in translating podcast sentiments from English to Spanish. DeepTranslator shows a consistent minor negative bias in detecting neutral sentiments, while GPT-4 shows greater variability in translating negative sentiments. The consistency in positive

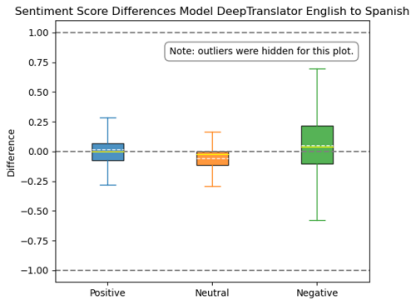

Figure 7: Sentiment Scores with pysentimiento Original English transcripts translated to Spanish using DeepTranslator

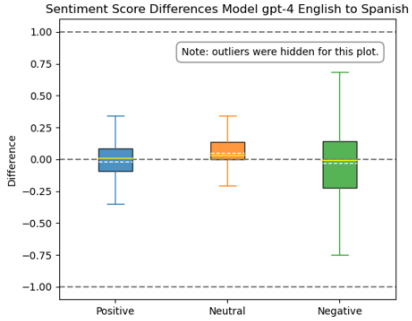

Figure 8: Sentiment Scores with pysentimiento Original English transcripts translated to Spanish using ChatGPT4

sentiment translation across both models suggests that both are well-tuned for recognizing and translating positive emotions effectively between English and Spanish.

3.3 Comparing the Sentiment Analysis Results with pysentimiento from Spanish to English translations

Overall, the comparative analysis reveals that both DeepTranslator and GPT-4 perform similarly across all sentiment categories when translating from Spanish to English. The minor differences in the spread and range might affect the choice of model depending on the application's tolerance for variability and the specific requirements for sentiment accuracy. These findings highlight the capability of both translation models to handle the complexity of sentiment analysis across languages with a reasonable degree of fidelity, though minor deviations suggest room for further optimization.

3.3.1 Negative Sentiment

The Negative sentiment category shows the most notable similarity between the two models, with

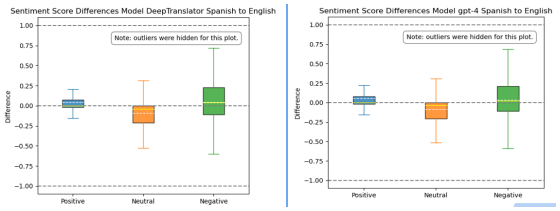

Figure 9: Comparative Sentiments Score Differences between deeptranslator and gpt-4 Spanish to English using pysentimiento

both exhibiting a median difference slightly below zero. This suggests a consistent slight underestimation of negativity in translations from Spanish to English. Both models also display a similar spread and range, indicating comparable performance in handling negative sentiments. The presence of longer tails in both models suggests that there are instances where the negativity is either significantly underestimated or overestimated, but these are less frequent.

3.4 Analysis of Sentiment Score Differences for Idioms Using pysentimiento

The analysis of sentiment scores for idioms translated from Spanish to English highlights the nuances in emotional tone captured by the pysentimiento model. The sentiment score differences, depicted in the box plots, illustrate how idiomatic expressions' sentiments are perceived differently across the two languages when translated.

3.4.1 Negative Sentiments

The analysis of Negative sentiments shows the largest variability among the three categories, with a median score difference also above zero. This implies that idioms with negative connotations in Spanish might be interpreted as less negative in English, or in some cases, even as non-negative. The range and spread are quite large, indicating significant inconsistency in how negative sentiments are translated. This might reflect the complexity of accurately conveying negative emotions across languages due to different cultural contexts or the intensity of expressions used in Spanish versus English.

3.4.2 Overall Insights

The translation of idiomatic expressions poses significant challenges, particularly in maintaining the original sentiment during the translation process. The disparities observed in the sentiment analysis underscore the complexities in-

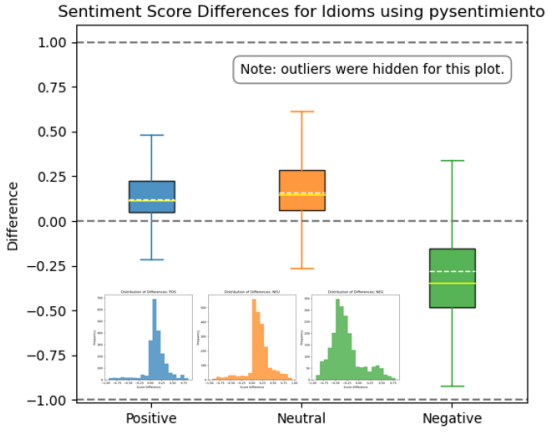

Figure 10: Sentiment Score Differences for Idioms using pysentimiento

Idiom Spanish	Meaning English	Idiom High SC	Meaning High SC
A buen hambre, no hay pan duro.	Beggars can't be choosers.	NEG	NEU
Estar como unasopa	to be soaked to the bone	NEG	NEU
Al mal tiempo, buena cara.	When life gives you lemons, make lemonade.	NEG	POS
Buscarle tres pies al gato	to make something more complicated than necessary	NEG	NEU
Meterse en camisa de once varas	to be out of your depth	NEG	NEU
Un clavo saca a otro clavo.	A new person will make you forget the old one.	NEG	POS
Dios los cría y ellos se juntan	Birds of a feather flock together.	NEG	POS
Echar agua al mar	to do something pointless	NEG	NEU
Echarle leña al fuego	to make matters worse	NEG	NEU

Figure 11: Comparing some mismatches from the sentiment scores perform in the idioms dataset with pysentimiento

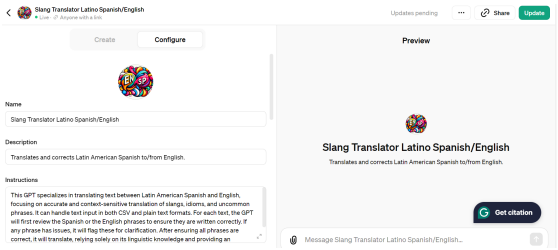

Figure 12: GPT-4 plugin: Slang Translator Latino Spanish/English

volved in translating idioms, which often contain culturally embedded meanings and emotional nuances that are not directly translatable.

The example provided, translating the phrase "Me quemo las pestañas trabajando" to "I'm burning the midnight oil working," aptly illustrates these challenges. While the English translation captures the essence of hard work and long hours implied in the Spanish idiom, subtle nuances in sentiment might still be lost or altered, as evidenced by the sentiment score differences.

This analysis emphasizes the importance of context and cultural sensitivity in the translation process, particularly for applications like content localization, where maintaining the original tone and sentiment is crucial. Such insights are invaluable for further refining translation models and improving cross-cultural communication.

3.4.3 Analysis of Sentiment Discrepancies in Idiom Translations

The table presents a comparative analysis of sentiment mismatches between Spanish idioms and their English translations, as evaluated by the model pysentimiento. This analysis highlights the challenges in maintaining consistent sentiment in translations due to cultural and linguistic differences. For instance, the Spanish idiom "Al mal tiempo, buena cara," typically conveying resilience (translated as "When life gives you lemons, make lemonade"), is noted to carry a negative sentiment in Spanish but is positively perceived in English.

Several other examples, such as "Metere en camisa de once varas" (To be out of your depth), show neutral sentiment in Spanish versus English translations, where it remains neutral. However, "Dios los cría y ellos se juntan" (Birds of a feather flock together) shifts from a neutral tone in Spanish to a positive one in English, reflecting potentially varying cultural perceptions of the phrase's connotation.

Overall, the table indicates a significant sentiment mismatch in approximately 33.46% of the cases, emphasizing the complexity of accurately translating sentiments in idiomatic expressions across languages. This underscores the necessity for advanced translation models that can better understand and adapt to the subtleties of cultural context in sentiment analysis.

3.5 GPT4 plugin

The "Slang Translator Latino Spanish/English" plugin is a sophisticated tool designed to bridge the linguistic gap between Latin American Spanish and English, specifically targeting the translation and correction of slang, idioms, and less common phrases. This plugin leverages GPT-4 technology to ensure context-sensitive and accurate translations, catering to both CSV and plain text inputs. Upon receiving text, the plugin first checks for any linguistic errors and requests clarification if necessary, ensuring that only correctly phrased inputs are processed. This meticulous approach allows the plugin to provide translations that maintain the original nuances and cultural relevance, making it an invaluable resource for users needing authentic and nuanced language translations.

Conclusions

- Using DistilBert model to score text samples on 6 emotions, Joy and Anger were the most difficult emotions to retain, regardless of the translation method used.
- There is a notable disparity in the scores for negative classifications, indicating that while the pysentimiento model identifies the original language as negative, the classification for the translated text varies significantly.
- The median is not zero on average, indicating that the sentiment models capture more differences in the idioms dataset compared to the transcripts. This may be due to idioms having similar meanings but using completely different words.
- Our fine-tuned GPT-4 assistant effectively captures idioms in both languages, which can aid in reinforcement learning for future refinements of the models used.

Acknowledgements

Using ChatGPT for grammar-checking purposes.

References

[1] *Apple shares the most popular podcasts of 2023*, Apple.

[2] *Deep translator: Translation for humans.*

[3] *Dictionary of british slang*, Oxford International English.

[4] *English sayings in spanish*, Speaking Latino.

[5] *English slang*, Language Realm.

[6] *Google neural machine translation*, Wikipedia. Accessed: 2024-04-29.

[7] *Google translate*, Wikipedia. Accessed: 2024-04-29.

[8] *Gpt-4*, OpenAI.

[9] *Hilarious spanish-english translation*, Summa Linguae Technologies.

[10] *Idioms in english*, BYJU'S.

[11] *Idioms with examples*, Leverage Edu.

[12] *Lemonada media*, Lemonada Media.

[13] *Playlist title.*

[14] *Sentiment analysis - ibm*, IBM.

[15] *Spanish idioms*, SpanishDict.

[16] *Spanish idioms*, Language Realm.

[17] *Spotify podcast charts - latam*, Spotify.

[18] OpenAI, *Introducing chatgpt plugins.*

[19] Juan Manuel Pérez, Mariela Rajngewerc, Juan Carlos Giudici, Damián A. Furman, Franco Luque, Laura Alonso Alemany, and María Vanina Martínez, *pysentimiento: A python toolkit for opinion mining and social nlp tasks*, 2023.

[20] Bhadresh Savani, *distilbert-base-uncased-emotion.*

[21] Pablo Valdivia, *17 spanish translation fails that'll make you laugh so hard*, BuzzFeed.

[22] Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing, *Sentiment analysis in the era of large language models: A reality check*, 2023.