Youtube Sentiment Analysis on AI in Europe

Di Pasquale Giuseppe
giuseppe.dipasquale.1@studenti.unipd.it

Helbing Marlon Joshua
marlonjoshua.helbing@studenti.unipd.it

Manocchio Andrea
andrea.manocchio@studenti.unipd.it

Passaro Giovanni
giovanni.passaro@studenti.unipd.it

Virzì Daniele
daniele.virzi@studenti.unipd.it

Abstract

This project aimed to perform a sentiment analysis of comments on YouTube videos related to artificial intelligence (AI). We focused on videos from some of the largest countries in Europe, conducting a comparative analysis to identify variations and commonalities in public sentiment toward AI across these nations. Leveraging state-of-the-art pre-trained models, we fine-tuned them for our specific task to ensure accurate results. Our findings offer valuable insights into public opinion on AI in different European countries. We anticipate that this research will be beneficial for both researchers and businesses engaged in the AI sector within Europe.

1. Introduction

Our group proposes an experimental project aimed at conducting sentiment analysis on YouTube videos about artificial intelligence (AI) through the examination of user comments. Understanding public opinion on AI is crucial, as it is a rapidly growing field that is expected to have a significant impact on society.

This sentiment analysis was conducted by scraping comments from some of the largest countries in Europe (France, Germany, Italy, and Spain) to perform a comparative study. Our first goal is to extract data from the web and share our dataset with the community. The second goal is to gain a better insight into public opinion towards AI. In particular, we hypothesize that due to the more rapid advances observed in northern European countries, the sentiment towards AI would be more positive. Yet, we found people in these countries to have a more negative view of it.

2. Dataset

We obtained our data from YouTube comments[7]. Since it was easier to find more comments in the English language, we collected a large dataset of those and smaller ones containing comments for each of the 4 languages we wanted to compare (Table 1). We used the English dataset to build our training, validation, and test set and the other datasets for inference. All the data was collected using the YouTube Data API v3 and then pre-processed to remove any irrelevant information to make it suitable for the model. We identified some queries to search for the videos, then separated all the videos collected by the bias conveyed by the title and content (positive, negative, or neutral).

Language	Comments
English	210,848
French	26,028
German	15,336
Italian	16,222
Spanish	114,109

Table 1. Number of total comments for each language.

3. Methodology

The methodology (Fig. 1) employed in this study is a systematic and comprehensive approach that ensures the integrity and reliability of the research findings. This rigorous methodology ensures a thorough and accurate analysis of the data, leading to reliable and insightful conclusions.

3.1. Models

Building a model from scratch for our sentiment analysis would be time-consuming, lead to high computational costs, and be inferior to the huge amount of state-of-the-art models already available. With the emergence of libraries

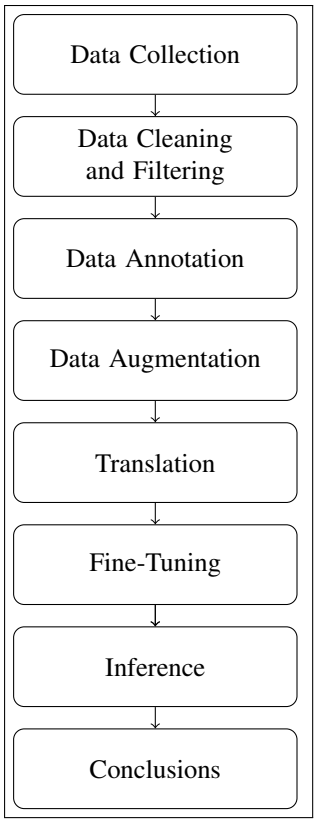

Figure 1. Methodology process.

such as Hugging Face, it is possible to have easy access to models pre-trained on large text corpora, which we will use for our task at hand. In particular, we made use of twelve different models:

Filtering (*English*):

1. *rabindralamsal/BERTsent* [13], a BERT-base [4] trained with SemEval 2017 corpus (39k plus tweets) and is based on (vinai/bertweet-base that was trained on 850M English Tweets (cased) and additional 23M COVID-19 English Tweets (cased).

Labeling (*English*):

1. *cardiffnlp/twitter-roberta-base-sentiment-latest* [10], a RoBERTa-based [9] model trained on 124M tweets.
2. *cardiffnlp/twitter-xlm-roberta-base-sentiment-multilingual* [10], a fine-tuned version of *cardiffnlp/twitter-xlm-roberta-base* on the cardiffnlp/tweet_sentiment_multilingual dataset.
3. *aychang/roberta-base-imdb*, a RoBERTa-based model trained on the benchmark dataset "IMDb" [11].
4. *siebert/sentiment-roberta-large-english* [6], a RoBERTa-large trained and evaluated on 15 datasets from diverse text sources to enhance generalization across different types of texts (reviews, tweets, etc.).
5. *lxyuan/distilbert-base-multilingual-cased-sentiments-student*, a distilBERT multilingual model trained by Zero-shot classifier distillation.

Rephrasing (Data Augmentation):

1. *tuner007/pegasus_paraphrase*, a PEGASUS [16] model fine-tuned for paraphrasing.

Translating:

1. *facebook/mbart-large-50-many-to-many-mmt* [15], a multilingual Sequence-to-Sequence model fine-tuned for multilingual machine translation.

Sentiment Analysis (*French, German, Italian, Spanish*):

1. *cmarkea/distilcamembert-base-sentiment* [3], a distilled version of the CamemBERT [12] model, the state-of-the-art language model for French, trained on a large-scale dataset scraped from "Allociné.fr" user reviews.
2. *ssary/XLM-RoBERTa-German-sentiment* [5], a specifically tailored XLM-T-RoBERTa [1] for the German language, has been fine-tuned on over 200,000 German-language sentiment analysis samples.
3. *osiria/bert-tweet-italian-uncased-sentiment*, a BERT-based model for the Italian language, fine-tuned for sentiment analysis on the SENTIPOLC-16 dataset, using BERT-TWEET-ITALIAN as a pre-trained model.
4. *VerificadoProfesional/SaBERT-Spanish-Sentiment-Analysis*, a finetuned BETO [2] model designed to detect sentiments in Spanish tweets.

3.2. Filtering

Given that our data collection involved real comments from YouTube, we encountered substantial noise, including spam, irrelevant comments, and comments not useful to our analysis. We applied several cleaning techniques for filtering purposes, such as removing stopwords, punctuation, and emojis. Furthermore, we restricted our analysis to comments that were 350 characters or fewer. (Table 2). A deeper analysis of the data led us to remove numbers

and utilize TextBlob to correct spelling mistakes[1]. Following the data cleaning process, we filtered the English comments using a pre-trained model for sentiment classification [8]. In particular, we only kept comments that contained at least one lemmatized word indicative of sentiment, given the model assigned a confidence score greater than 0.95%. Additionally and specific to our task, we kept comments that contained buzzwords such as AI or robot. The rationale behind this method is to retain only the comments that express an opinion and are relevant to AI.

Language	Comments
English	43,151
French	21,524
German	9,855
Italian	11,862
Spanish	72,741

Table 2. Number of total comments for each language after filtering.

3.3. Labeling

Sentiment analysis is a subclass of text classification that measures the emotional tone or polarity of a piece of text. In our case, we wanted to classify sentences as negative, neutral, and positive. Therefore, we had to gather labels for our dataset D, that is

$$D = \{(x_i, y_i) \mid x_i \in X, y_i \in Y \text{ and } i = 1, \ldots, m\}.$$

where X are our comments and Y the sentiments. We used 5 different models to classify English comments, determining the label Y through a majority voting process. This parallel ensemble method leverages the independence between base learners. In the case of voting-based learners, predictions are made, and the one with the highest number of votes is selected. This method can reduce overfitting, improve accuracy, handle noisy data, and increase model robustness. To prove that, just consider that each base binary classifier h_j has an accuracy $P(h_j(x) = f(x))$ and $h_j(x) \in \{-1, 1\}$. So combining T of these classifiers according to

$$H(x) = sign\left(\sum_{j=1}^{T} h_j(x)\right),$$

i.e., H makes an *error* when $> 50\%$ of its base classifiers make errors. According to the *Hoeffding inequality*

$$P(H(x) \neq f(x)) = \sum_{k=0}^{T/2} \binom{T}{k} (1-\epsilon)^k \epsilon^{T-k} \leq e^{-\frac{T}{2}(2\epsilon-1)^2},$$

the *generalization error* decreases exponentially as the ensemble size T increases. Since running more models was computationally costly, we used $T = 5$ which was the best trade-off of costs and benefits of majority voting according to our tests. To improve the accuracy of these labels, we kept only the pairs (x_i, y_i) with y_i labeled with a *confidence score* $\geq 80\%$, removing ambiguous comments. Using this method, we noticed that not a single comment was assigned the label neutral. Therefore, we will omit this class from our analysis. To introduce our own bias, we relabelled comments that had a confidence score between 80% and 90% ourselves.

3.4. Augmentating

Once we obtained our training set D, we observed that the data was balanced across both classes, but the number of samples was insufficient for effectively fine-tuning a language model (Table 3). Thus, we applied Data Augmentation [16]. In particular, we created

- 7 new phrases for each comment with a confidence score between 80% and 90%, resulting in a higher representation of our bias in the data;
- 2 new phrases for comments with a confidence score between 90% and 95%;
- 1 new phrase for comments with a confidence score between 95% and 100%.

We ensured the dataset remained balanced by augmenting an equal number of comments in both positive and negative classes. After these steps, we obtained the training set $\hat{D}$ (Table 4). Note that we augmented on a version of the original comments with just minimal data processing such that we can generate real sentences.

Bias	Comments
Positive	9887
Negative	8452
Total	**18339**

Table 3. Training set D.

Bias	Comments
Positive	22293
Negative	20858
Total	**43151**

Table 4. Augmented Training set $\hat{D}$.

[1]Note that this only worked for the English comments.

3.5. Translating

Subsequently, we translated these English comments into the four objective languages [15], thereby creating four distinct versions of the same dataset. These versions were then utilized to fine-tune the model for each respective language.

3.6. Sentiment Analysis

Finally, we used these models to make inferences on the German, Italian, Spanish, and French datasets and used the results for an in-depth analysis.

3.7. Other methodologies

Before adopting this methodology, we tested different approaches to solve our problem. We first manually labeled 10K English comments (2K for each person) and later planned to fine-tune a multilingual distilBERT [14] to make inferences on European comments. We encountered several problems, such as different opinions on the labeling of some comments and a lack of accuracy of the model on our labeled dataset.

- **Complexity of the task**: It is difficult to understand and assign the sentiment of a YouTube comment that expresses an opinion about AI.
- **Noise**: After manual labeling, we recognized the prevalence of irrelevant comments and how challenging it would have been to get good accuracy on this type of data.

To solve the first problem, we used pre-trained models and a majority vote. To mitigate noise, we filtered the data by selecting only comments labeled with a high confidence score.

4. Experiments

4.1. Data Split

We partitioned our dataset into a train (70%), validation (15%), and test (15%) set, stratifying by labels to keep the data balanced in each split (Table 5). Furthermore, we tokenized the comments of each split using the pre-trained tokenizer of each of the four models respectively to get them ready for processing.

	Training	Validation	Test
Positive	15595	3342	3341
Negative	14598	3128	3129
Total	30193	6470	6470

Table 5. Distribution of comments in training, validation, and test sets

4.2. Fine-Tuning

We fine-tuned each model on the L4 GPU from Google Colab, using the same hyperparameters to make them comparable, since we effectively used the same dataset for each model (Table 6). As we are fine-tuning state-of-the-art models that are already trained on sentiment analysis, we found 2 epochs to be enough to reach a desirable accuracy. Additionally, we had memory issues in some of the models and couldn't train for more than 2 epochs. Note that we used a Learning Rate Scheduler.

Hyperparameter	Value
Epochs	2
Batch Size Train	128
Batch Size Validation	128
Learning Rate (Initial)	1e-4
Weight Decay (L2)	1e-2

Table 6. Hyperparameters used in the model

4.3. Label conversions

Due to the variety of models we chose, their labeling approaches differed. Consequently, we had to adjust these approaches to align with our labeling methodology. The French model employed a labeling system ranging from 1 to 5 stars, with 1 star representing the most negative sentiment. We decided to assign comments with a confidence score $\geq 90\%$ 1 star if negative and 5 stars if positive. Comments with a confidence score between 80 and 90 were assigned 2 stars if negative and 4 stars if positive. We omitted 3 stars as we do not have neutral comments in our dataset. The German model used negative, neutral, and positive labels, so we omitted the neutral class yet again. The Spanish and Italian models had positive and negative labels, so no adjustments were necessary.

4.4. Accuracy Results

Overall, we achieved test accuracies over 0.85 across all models (Table 7).

Model	Accuracy %
French	0.88
German	0.93
Italian	0.91
Spanish	0.89

Table 7. Accuracy on the test set for different fine-tuned models

5. Inference

To analyze public opinion toward AI across Italy, Spain, Germany, and France, we employed our fine-tuned models

to perform inference on the comments we scraped in these four languages. We removed noisy comments by selecting only those with a confidence score $\geq 90\%$ (Table 8).

Language	Comments
French	12265
German	5028
Italian	5419
Spanish	34397

Table 8. Comments for each country

5.1. Different Sentiments in Different Countries

We have found that France and Germany are more negatively oriented towards AI, Italy is mildly positive, while Spain is largely pro AI (Table 9).

Country	Pro AI comments %
France	39.85
Germany	42.70
Italy	54.94
Spain	65.86

Table 9. Inference results

5.2. Common Topics in Comments

To have a better insight into public opinion we looked at the most common words (amongst the relevant ones) that better conveyed the crowds' thoughts on AI across all countries, which were showing similar trends. We achieved so by computing the percentage of each word, merging while normalizing across datasets to avoid the larger ones taking over and translating all to English, taking the top 100, and keeping only the ones expressing some sentiment towards AI (Figure 2).

5.3. Confidence Differences In Labeling

Looking at the labels confidence scores, we see noticeable differences between positive and negative comments. Positive comments tend to be overall classified with a higher confidence score compared to negative ones, which seem to be trickier to identify for a model (Figure 3).

6. Discussion

In this project, we leveraged multiple state-of-the-art natural language processing models to conduct a sentiment analysis of opinions on AI, utilizing real-world data obtained from YouTube, one of the largest social media platforms. The results that we found stand in contrast to our initial hypothesis. Our interpretation is that due to the rapid growth of AI in northern European countries, people in these areas are more exposed to AI and its effects. Consequently, the fear is amplified due to its novelty, advanced complexity, and difficulty in comprehending it. Furthermore, looking at our data we found some common topics of attention like "fear", "work" or "problem", representing the masses' most shared concerns about the future of AI. On the other hand, we also have an important presence of positive sentiment words such as "good", "better" or "interesting". However, we note that many of these positively connotated words may have been used in an ironic sense, thus actually representing a negative sentiment.

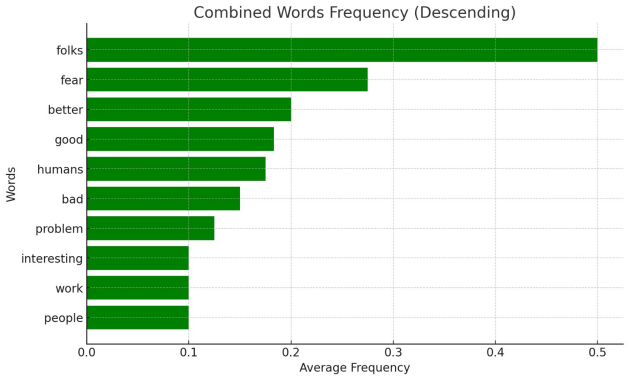

Figure 2. Representative word frequency

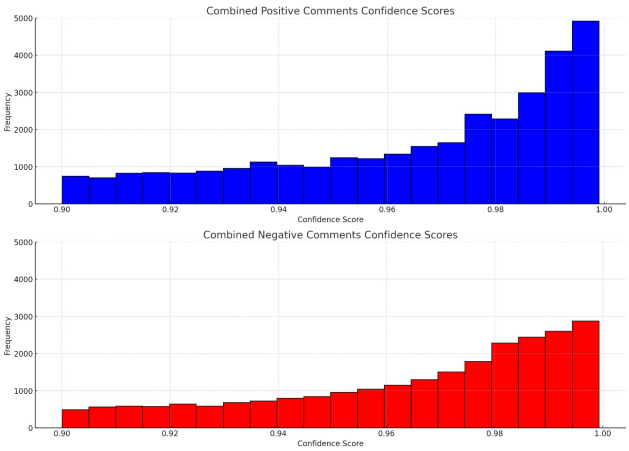

Figure 3. Representative word frequencies

7. Further Work

Due to the lack of computational power and time, this project presents just the prototype of a model that can be enhanced in many ways. First, we propose employing more sophisticated methods to obtain a cleaner set of com-

ments for fine-tuning the models. In particular, we thought about the idea of training a denoising model. A supervised approach could be to have an encoder-decoder structure, which removes noisy parts from comments. The issue here is obtaining the golden labels, as it needs to be done manually, which is time-consuming and likely results in a biased model. Another approach would be a classification denoising model which is trained on a set of noisy comments. Finding the correct training set here is again challenging. Secondly, we assume that using human-based labeling compared to a majority vote over multiple models can result in more accurate results for the task at hand. Even though we picked state-of-the-art models, the concept of positive and negative sentiment is highly abstract and differs from task to task. In our case, a positive sentiment means in favor of AI, while a negative sentiment means against AI. As described earlier, we tried to label 10K comments ourselves, but ran into multiple problems doing so. In the end, we relabelled just a small set of comments and augmented them to introduce our bias more strongly. Thus, we propose a majority vote human labeling approach on denoised comments. Thirdly, we propose scaling up the model to a larger scale. The inclusion of more comments and additional languages can enhance the model's generalizability, leading to less biased results and a better overview of the opinions about AI in Europe. We ran into memory issues when trying to run the fine-tuning process for more than 2 epochs using just a set of 40K comments. Nevertheless, due to the high accuracy we achieved on the test set, our results are impactful. Lastly, we have seen that for each language, comments were predicted to have a positive sentiment with higher confidence compared to comments with a negative sentiment. We therefore propose to lay more focus on these negative, potentially tricky comments in the fine-tuning process to ensure a more robust model for both classes.

References

[1] Francesco Barbieri, Luis Espinosa Anke, and Jose Camacho-Collados. Xlm-t: Multilingual language models in twitter for sentiment analysis and beyond, 2022.

[2] José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. Spanish pre-trained bert model and evaluation data. In *PML4DC at ICLR 2020*, 2020.

[3] Cyrile Delestre and Abibatou Amar. DistilCamemBERT : une distillation du modèle français CamemBERT. In *CAp (Conférence sur l'Apprentissage automatique)*, Vannes, France, July 2022.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[5] Oliver Guhr, Anne-Kathrin Schumann, Frank Bahrmann, and Hans Joachim Böhme. Training a broad-coverage German sentiment classification model for dialog systems. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1627–1632, Marseille, France, May 2020. European Language Resources Association.

[6] Jochen Hartmann, Mark Heitmann, Christian Siebert, and Christina Schamp. More than a feeling: Accuracy and application of sentiment analysis. *International Journal of Research in Marketing*, 40(1):75–87, 2023.

[7] Marlon Helbing, Daniele Virzì, Giuseppe Di Pasquale, Andrea Manocchio, and Giovanni Passaro. Youtube Sentiment Analysis, June 2024.

[8] Rabindra Lamsal, Aaron Harwood, and Maria Rodriguez Read. Twitter conversations predict the daily confirmed covid-19 cases. *Applied Soft Computing*, 129:109603, 2022.

[9] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

[10] Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-Collados. Timelms: Diachronic language models from twitter, 2022.

[11] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[12] Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. Camembert: a tasty french language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020.

[13] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. BERTweet: A pre-trained language model for English Tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 9–14, 2020.

[14] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020.

[15] Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. Multilingual translation with extensible multilingual pretraining and finetuning. 2020.

[16] Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization, 2019.

8. Contributions

- **Scraping of videos** *(Helbing, Di Pasquale, Manocchio, Passaro, Virzì)* : We scraped all the videos together using YouTube's API. (50 hours)
- **Data Cleaning & Filtering** *(Helbing, Passaro, Virzì)* : Helbing, Passaro and Virzì were responsible for brainstorming the ideas while Helbing and Virzì implemented them. (7 hours)
- **Picking out the models** *(Virzi, Manocchio, Passaro)* : The three picked out all the models from HuggingFace. (2 hours)
- **Data Augmentation** *(Manocchio, Helbing)* : Both worked out the logic behind the Data Augmentation and Helbing implemented the code. (25 hours)
- **Data Translation** *(Passaro, Virzi, Manocchio, Helbing)*: Virzi had the idea of translating comments to fine-tune multiple models and Passaro implemented it. Together we translated in different languages. (50 hours)
- **Data Fine-Tuning** *(Helbing)*: Helbing wrote the models for the fine-tuning process and fine-tuned them. (35 hours)
- **Data Inference** *(Helbing, Manocchio, Di Pasquale, Virzi, Passaro)*: Together we implemented the Code for Inference in the end and then ran our fine-tuned models together. (8 hours)
- **Data Visualization** *(Virzi, Passaro, Manocchio)*: Virzi delivered the prototype and ideas for our data visualization and Passaro and Manocchio made it accessible with our inferenced data. (8 hours)
- **Presentation** *(Virzi)*: Virzi prepared the whole Presentation. (12 hours)
- **Report** *(Virzi, Helbing, Passaro)*: The three worked together on preparing the report. (25 hours)