Emotion profile analysis on movies

Lorenzo Cimini

Università degli Studi di Milano

1. Introduction

Our emotions, which are subjective experiences, can influence our thoughts, behaviors, and moods. They can be positive or negative, and they can range from simple emotions like happiness and sadness to more complex ones like surprise, love, envy, and humiliation. Emotions, which are fundamental elements in our life, have an impact on our daily decisions, social interactions, and even our physical health. In this context, Emotion recognition, a branch of sentiment analysis, offers itself as a tool that aims to extract fine-grained emotions from various types of data.

It's easy to see why emotion recognition is crucial: it can help marketers better understand how customers feel about their goods and services, which can boost customer satisfaction; it can be used in healthcare to identify mental health conditions and keep track of patients' emotions during therapy; in safety, to identify early warning signs of aggression or violent behavior, which allows early intervention to prevent harm; etc. Emotion recognition is then a useful technique that can improve our comprehension of human behavior and foster wellbeing across a range of industries.

We have the ability to communicate our feelings verbally and nonverbally. While nonverbal signals like facial expressions, tone of voice, gestures, and body language are used as nonverbal indicators of our emotions, words such ”happy,” ”angry,” or ”sad” are examples of linguistic ways of expressing our emotions.
Since there are many different ways through which feelings can be expressed, there are many technologies, each specialized to work with a different type of data and with its own strategy, which are specifically designed to recognize emotions by using different type of information, such as images and video (to capture nonverbal signs) or text (to collect verbal indicators). The task of emotion recognition applied to texts will be the focus of my efforts for this project.

When attempting to achieve the intended goal, several difficulties must be overcome. The model's ability to accurately identify emotions is one of the main problems because emotions can be complex and ambiguous. The wide range of emotions across different languages and cultures shows another difficulty. In addition, the model can have trouble understanding emotions in complex or ambiguous scenarios such as sarcasm or humor. Finally, the datasets that are now accessible are mainly small, covering just a small subset of the emotion taxonomy, and using imprecise classification [1].

2. Research question and methodology

The goal of this project is to first build a model capable of detecting emotions (multiclass classification task) contained in texts and, secondly, exploit this model to study an emotional profile of the main characters in one of the movies included in the Cornell Movie–Dialogs Corpus.

2.1. Emotion Taxonomy

To avoid making the classification process excessively complex and to remove emotional overlap, the list of emotions considered has been simplified. To select the right emotions to use, the six basic emotion categories (fear, joy, anger, disgust, sadness, and surprise) proposed by Ekman [2] in 1992 have been taken into consideration when applying this concept to the data with 'disgust' deleted as the most overlapping one.

2.2. Dataset

In order to have a acceptable amount of data, two datasets, GoEmotion (from Google) and Emotion (from huffingface), have been used for our model's training.
GoEmotion is proposed as the largest fully human-annotated dataset of 58k Reddit comments extracted from popular English-language subreddits and labeled with 27 fine-grained emotion categories while Emotion consists of three sets: train, validation, and test set for a total of 20 thousand annotated sentences, and has six kinds of emotion: sadness, joy, love, anger, fear, and surprise.
Regarding the emotion recognition task on movie characters, Cornell Movie-Dialogs Corpus [3] has been used. The corpus includes a large, well annotated collection of fictional speaks that were taken from 617 movie scripts. From this corpus, data from the movie 'Good Will Hunting', has been extracted.

Text preprocessing
To determine what kind of data is available and then determine which text processing tasks need to be completed, it is necessary to review and tackle both the GoEmotion and the Emotion datasets. By analyzing the datasets is easy to understand which actions have to be applied to the data:

- GoEmotion contains special tokens like [RELIGION] and [NAME] used to mask proper names referring to people and religion terms;
- GoEmotion contains URL addresses, emojis, numbers and dates;
- GoEmotion contains reddit comments for which the emotion is ambiguos;
- Both datasets have to be manipulated in order to take into account only the emotions descripted before;
- Each emotion must be represented by a distinct identifier integer.

Dataset balancing
The dataset has been resided to guarantee that each class is balanced in order to avoid biases in the model. With the goal to do that, undersampling is performed on each class with the aim to produce a dataset containing 3357 sentences for each emotion.

text	label
i didnt feel humiliated	0
im grabbing a minute to post i feel greedy wrong	3
i am feeling grouchy	3

Example annotations from the dataset

2.3. Emotion Recognition Models

Two alternative approaches have been employed in the project to examine and evaluate various techniques. The first one is based on the traditional models used for binary and multiclass classification, such as logistic regression whereas, the second one is based on BERT (Bidirectional Encoder Representations from Transformers). The goal is to develop the best model (for our task and dataset) for each strategy, compare them, and utilize the most effective one to analyze the emotional profile of characters in the movie dialogues.

2.4. Models evaluation

To evaluate the models used within the project and make the comparison, several metrics have been used.

- Precision
  The precision can be interpreted as the measure that describes how big is the portion of our result that is correct:
  $$P = \frac{TP}{TP + FP}$$
  The measure can assume values between 0 and 1.
  The downside of this measure is that if we decide to retrieve few documents (even just one) that are correct, we would get the maximum precision we can get.
- Recall
  The recall can be interpreted as the measure that describes how big is the portion of the correct results that our model has retrieved:
  $$R = \frac{TP}{TP + FN}$$
  The measure can assume values between 0 and 1.
  The downside of this measure is that if we decide to retrieve the whole set of documents we would get the maximum recall we can get.
- F1
  One measure, that even in this case assumes values between 0 and 1, that combines both precision and recall is F1-score. This score is the harmonic mean of the precision and recall:
  $$2 \cdot \frac{precision \cdot recall}{precision + recall}$$
- Accuracy
  In order to understand how big is our model's correct decision fraction, we can use the accuracy:
  $$\frac{TP + TN}{TP + TN + FP + FN}$$

3. Experimental results

This section provides experimental results from the two employed techniques.

3.1. Classical approaches

Five models—RandomForestClassifier, LinearSVC, MultinomialNB, LogisticRegression, and SGDClassifier—have been evaluated and compared in order to determine which is the best classifier for the given task.
After having vectorized documents by using the Tfidf approach, a 5-fold cross-validation has been done for each model.

Model	Accuracy
LinearSVC	0.7439
LogisticRegression	0.7276
MultinomialNB	0.7069
RandomForestClassifier	0.5862
SGDClassifier	0.7574

Models' average accuracy on 5-fold cross validation.

Data from the model's evaluation using cross-validation indicates that the SGDClassifier performs better than the other models with an average accuracy of 0.7574.

Once the optimal model for the task has been identified, hyperparameter tuning has been carried out using RandomizedSearchCV from Scikit-Learn, which, rather than attempting all possible combinations, searches the best configuration at random through some configurations that are given.

Model explanation

What happens under the good when the model tries to predict the emotion in a text is not exactly clear. In order to understand what the model has learned during the training and how it predicts, ELI5, that stands for 'explain like i'm 5', has been used. In particular, ELI5 is a Python package which helps to debug machine learning classifiers and explain their predictions providing support for a wide variety of machine learning frameworks and packages.

In our case, the gol is to understand how our classifier makes decision and for this purpose we can use *ELI5.show_weights*:

y=1 top features		y=4 top features		y=3 top features		y=0 top features		y=5 top features	
Weight?	Feature	Weight?	Feature	Weight?	Feature	Weight?	Feature	Weight?	Feature
+5.569	sad	+5.372	glad	+4.408	rushed	+4.731	afraid	+5.501	surprised
+3.987	sorry	+4.712	happy	+4.364	violent	+4.640	scared	+4.445	shocked
+3.961	unfortunate	+4.167	enjoy	+4.175	distracted	+4.213	vulnerable	+3.947	impressed
+3.691	discouraged	+3.512	fun	+4.017	irritated	+4.147	frantic	+3.751	curious
+3.566	devastated	+3.486	sincere	+3.684	impatient	+4.104	hesitant	+3.368	wondering
+3.476	sadly	+3.250	clever	+3.654	insulted	+3.991	terrified	+3.298	amazed
+3.350	defeated	+3.165	enjoyed	+3.624	stubborn	+3.681	shaken	+3.210	dazed
+3.287	foolish	+3.081	passionate	+3.618	rebellious	+3.635	terrifying	+2.694	believe
+3.251	humiliated	+3.001	joy	+3.578	greedy	+3.589	apprehensive	+2.686	surprise
+3.218	exhausted	+2.991	respected	+3.542	rude	+3.576	scary	+2.674	omg
+3.184	punished	+2.977	trusting	+3.521	hated	+3.529	reluctant	+2.671	wonder
+3.162	painful	+2.967	lucky	+3.435	annoyed	+3.328	frightened	+2.512	oh
+3.101	inadequate	+2.871	proud	+3.416	selfish	+3.323	restless	+2.497	wow
+3.049	pathetic	+2.856	ecstatic	+3.361	hate	+3.148	shy	+2.260	fat
+3.039	heartbroken	+2.847	reassured	+3.314	resentful	+3.045	skeptical	+2.259	wondered
+3.038	melancholy	+2.797	superior	+3.292	fuck	+3.032	horrible	+2.141	surprisingly
+3.015	blank	+2.786	intelligent	+3.282	hell	+2.950	uncomfortable	*... 1237 more positive ...*	
+2.953	groggy	+2.786	accepted	+3.263	envious	+2.946	fearful	*... 3318 more negative ...*	
+2.937	hopeless	+2.761	festive	+3.243	offended	+2.932	fear	-2.140	happy
+2.933	lost	+2.732	awesome	+3.228	bitchy	+2.893	nervous	-2.216	terrifying
... 2265 more positive ...		*... 2530 more positive ...*		*... 2273 more positive ...*		*... 1368 more positive ...*		-2.467	sad
... 3939 more negative ...		*... 3928 more negative ...*		*... 4027 more negative ...*		*... 3952 more negative ...*		-3.156	feel

Fig. 1. Model's weights showed by ELI5

The values indicate the importance of a specific feature for the prediction while the sign indicates the direction. For instance, if the word "surprised" appears in the text, it will push significantly the classification process toward the emotion 'surprise'.

Furthermore, by using this data, we can create different WordClouds (one for each emotion) to visualize the most important features that the model considers when deciding wheter to classify a text as belonging to one class or another.

Fig. 2. WordClouds of 'joy', 'fear', 'sadness'

Model evaluation
By testing the model with unseen data (test dataset) it achieves the following scores: It is evident

accuracy 0.6982549684924867

	precision	recall	f1-score	support
0	0.70	0.71	0.71	961
1	0.77	0.80	0.78	1173
3	0.67	0.71	0.69	1114
4	0.67	0.54	0.59	472
5	0.60	0.52	0.56	406

Fig. 3. SGDClassifier on unseen data

when looking at the model's results for each class, that the model is not really able to differentiate between emotions that are semantically close like 'fear' and 'sadness'.
In conclusion, taking into account the model's complexity and the scores retrieved from the testing phase, we can say that the model's performances are quite good.

3.2. BERT

The second strategy used within the project for recognizing emotions is BERT [4]. This model, release by Google, is a pre-trained model that has been trained on the whole English Wikipedia and Brown Corpus based on the trasformer architecture.
The big advantage of this model is that it can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks such as sentyment analysis and language inference without making substantial architecture modifications. During various experiments done with BERT, it has advanced the state of the art for eleven NLP tasks.
In our case the model, trained on unlabeled data over different pre-training tasks, is first initialized with the pre-trained parameters and then fine-tuned using labeled data from our dataset.

Text preprocessing
Texts must be processed and prepared in accordance with BERT's requirements in order to feed BERT with our data. We can use a specific class, that is *AutoTokenizer*, in order to perform all necessary steps, such as padding and special tokens (like [CLS]) insertion, that are needed to retrieve the model's input.

Fine-tuning the model
A 15-epochs fine-tuning process has been performed in order to optimize model's parameters with our labaled dataset and for our specific task.

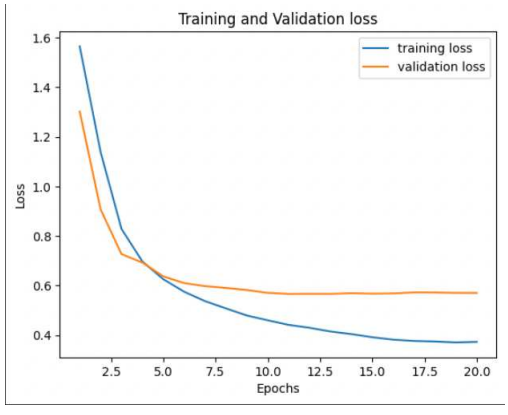

Fig. 4. Training and validation loss curve

According to the good fit shown in the picture, is clear how the model has been able to learn during the training. With the validation loss having a ”small” gap with the training loss, it is particularly clear how both the training loss and the validation loss steadily reduce to a point of stability.

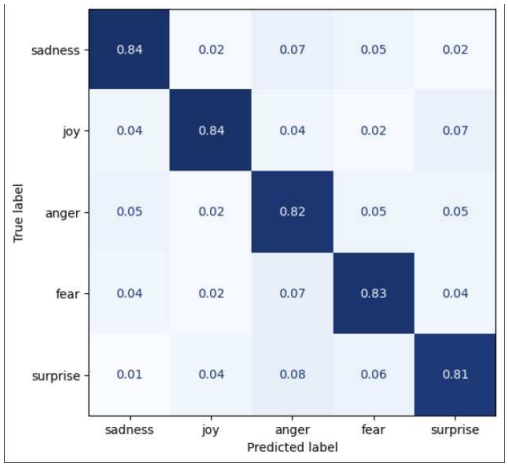

Fig. 5. Confusion matrix on test dataset

When the model is used on the train dataset, it proves how well it is able to classify compared to the SGDClassifier, reaching an accuracy of 0.83, which is +0.13 greater than the one gained using the classic method. Even in this situation, it is shown by plotting the confusion matrix that the model frequently predicts incorrectly when the sentence has a correct emotion that is semantically similar to the correct one, such as ”sadness” and ”fear”.

3.3. Emotional profile analysis

Once decided that the fine-tuned BERT is the best model to use for emotion recognition in our project, it is been used to analyze Will’s emotion profile in ’Good Will Hunting’ movie. In order to gather some insight from the data, multiple analyses have been performed on data that belongs to the movie characters Will and Sean.

Emotion detection pipeline

It’s very difficult to apply emotion detection to dialogues between two people in movies, just like it is in every other context. In fact, it’s possible that humor, overlapping emotions, and many other issues could occur when performing this kind of activity. Additionally to the problems mentioned above, there is also the issue of ’neutral’ utterances, utterances in which neither of the emotions are expressed, that has to be solved.

The method for dealing with ”neutral” utterances is based on the idea that one may tell whether an utterance is neutral or not by evaluating the model’s confidence in his prediction and comparing it to a threshold. In this way, we can classify even ”neutral” utterances by creating a threshold below which all sentences are considered to be neutral. Any neutral utterance given after this process won’t be considered in our analysis.

The so-called logits, representing the probability distribution without normalization, can be retrieved from the model’s output and used to calculate the classifier’s confidence in his prediction. The idea is to use the softmax function to normalize these probabilities so that the resulting series of numbers add to 1, will represent the model’s confidence for each possible class. In fact, a vector of numbers—which might be positive or negative—will be the output of the model and, by using the mentioned function, these numbers can be transformed into a vector of real numbers that sum to 1 using the softmax function so that they can be read as probabilities.

The equation is the following:

$$\sigma(\overline{z}_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

In conclusion, the model's confidence for each of the classes will be calculated and the greatest of these probabilities will be compared with the threshold (which, after numerous experiments, was chosen to be 0.7) after each prediction in order to assign the emotion predicted if the confidence is higher than the threshold, 'neutral' else.

Will and Sean's relationship

The friendship between Sean and Will is a key element in the film. In particular, it changes throughout the film from beginning as a very hostile relationship to ending as a very good friends relationship. Because of this, the model has been applied on data which includes discussions between Will and Sean to determine whether it can be used to recover those changes.

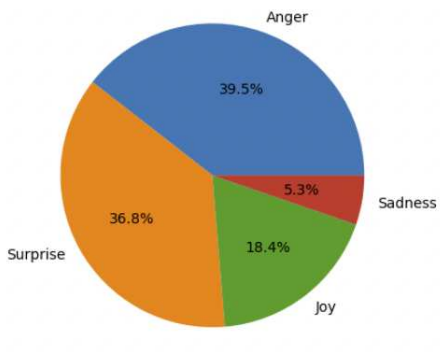

Fig. 6. Will's emotion in conversations with Sean

It is clear from the chart that "anger" played a significant role in the dialogue between Will and Sean. A different view on the same data might be produced in order to dig further into the analysis:

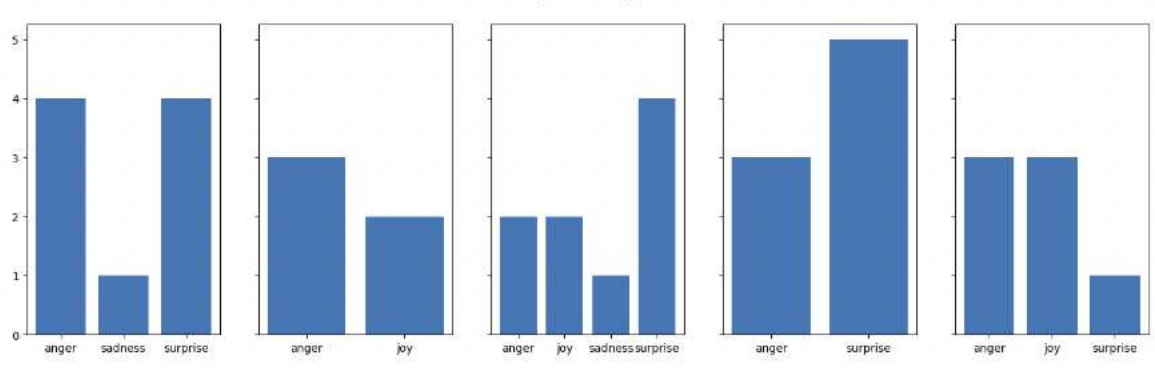

Fig. 7. Will's emotion with Sean grouped

This graph divides Will's feelings into five distinct groups and displays them in chronological order. It is clear from looking at the various graphs that the 'joy' emotion is missing from the conversations in the first part but starts to become more evident in the conversations between the two characters about half of the conversations.

In contrast to Will's feelings, we can examine Sean's emotions this time as they were observed

during interactions with Will in order to produce the same chart:

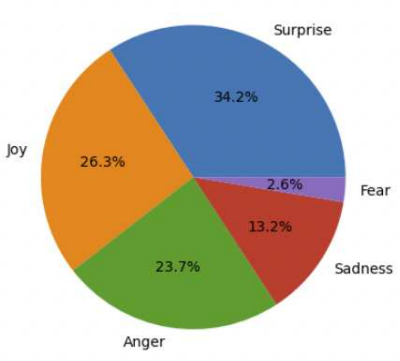

Fig. 8. Sean's emotion in conversations with Will

We can observe from the chart that the emotions are there, but in different ratios than those that were obtained from Will. For instance, while joy and sadness are both more frequent than anger, the latter is only displayed half as often. Additionally, Sean has experienced 'fear' that Will has never expressed.

Will's emotion

Comparing Will's emotions during his conversations with Sean with Will's emotions during the whole movie can give us more insight between their relationship.

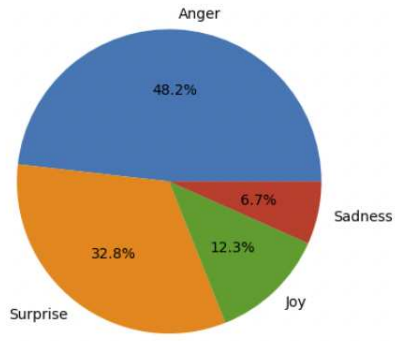

Fig. 9. Will's emotions in the movie

Even in this situation, we may claim that "anger" is the will's main feeling. This feeling dominates about half of his conversations, while joy is only present in 10% of them.

4. Concluding remarks

Emotion identification is important and challenging to execute correctly, as was mentioned in the beginning. As evidenced by the 'ambigous' label given to sentences in the GoEmotion dataset for which the emotion is unclear, this could generally be a highly challenging task, sometimes even difficult for humans. For these reasons, more complex models and a bigger volume of data are required to complete this work correctly.

Even though the range of emotions has been decreased, the project's proposed model has achieved a very high level of accuracy. However, more could be achieved by using other techniques that can accurately classify a wider range of emotions or capture other writing-specific characteristics like humor.

References

[1] D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Nemade, and S. Ravi, *GoEmotions: A Dataset of Fine-Grained Emotions*. 05 2020.

[2] P. Ekman, "Are there basic emotions?," *Psychological Review*, vol. 99, no. 3, pp. 550–553, 1992.

[3] C. Danescu-Niculescu-Mizil and L. Lee, "Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs.," in *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*, 2011.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2018.