Emojis For Sentiment Analysis

Muhammet Ali ŞEN, Muhammed Eren ŞEKKELİ
Bilgisayar Mühendisliği Bölümü
Yıldız Teknik Üniversitesi, 34220 Istanbul, Türkiye
{ali. sen, eren. sekkeli}@std. yildiz. edu. tr

Özetçe —**Bu projenin amacı doğal dil işleme aşamalarından anlamsal analiz olarak bir duygu analizi çalışmasıdır. Twitter platformu üzerinden toplanan ve tek emoji içeren veri seti üzerinden çeşitli makine öğrenmesi yöntemleri karşılaştırılarak emoji tahmin skorları başarım oranları araştırılmıştır.**

Anahtar Kelimeler—Doğal Dil İşleme (NLP) , Doğal Dil Apartı (NLTK) , Makine Öğrenmesi (ML) , Naive Bayes (NB), Destek Vektör Makineleri (SVM) , Terim Sıklığı-ters döküman sıklığı (TF-IDF) , Sayım Vektörleyici (CV), Derin Öğrenme (DL), Emoji Tahmini, Duygu Analizi.

Abstract—**In summary, the project is using tweets containing a single emoji as a dataset to perform sentiment analysis, which is a stage of natural language processing. It will be comparing different methods of machine learning techniques, to evaluate the success rates of emoji prediction. The ultimate goal is to predict the appropriate emoji for a given sentence.**

Keywords—Natural Language Processing (NLP), Natural Language Toolkit (NLTK), Machine Learnig(ML), Emoji Prediction, Sentiment Analysis, Naive Bayes (NB), Support Vector Machines (SVM), term frequency-inverse document frequency (TF-IDF), Counter Vectorizer (CV), Deep Learning (DL)

I. Introduction

Here, we will provide a general overview of the project.

Communication is essential in linking us to the world around us on a regular basis. Effective communication is the key to creating lasting bonds, promoting understanding, and succeeding whether it's exchanging thoughts, expressing emotions, or sharing information. But how much of our communication actually happens through words, do you ever wonder? According to recent studies, non-verbal cues frequently have a bigger impact than what we say when it comes to communicating than what we really say.

Only a small portion of communication, according to research, may be linked to the words we use. The classic Mehrabian communication model, which has been hotly contested, states that body language accounts for 55% of influence in communication, tone of voice for 38%, and speech for 7%. It is evident that non-verbal cues have a considerable impact on human interactions, even though the exact percentages may vary depending on the circumstances and cultural environment [1].

The most effective non-verbal form of communication is body language, which includes facial expressions, gestures, posture, and eye contact. It is remarkably accurate in expressing attitudes, emotions, and intentions. Crossed arms can suggest defensiveness or disagreement, whereas a simple smile can convey warmth and approachability right away. Our body language can dramatically affect the overall message we convey by either supporting or contradicting the things we say.

Another important aspect of communication is voice tone, which is quite influential. The meaning and emotional context of our statements can be influenced by the pitch, loudness, tempo, and intonation of our voices. While a harsh or aggressive tone may arouse fear or wrath, a soothing and reassuring tone can reassure and comfort. When we pay attention to the speaker's tone, which gives the uttered words richness and depth, we can frequently determine the underlying intention behind a message.

Speech should not be undervalued while making up a relatively modest portion of communication's overall influence. The impact of the message is influenced by the words we use, their clarity, and how we construct our sentences. Clear and coherent communication requires accurate and succinct thought and idea expression. The strength of communication resides in its capacity to communicate intricate ideas and stimulate intellectual discussion.

But verbal connection is not the only form of communication. There are many additional ways that communication can occur, each with its own benefits. Written communication has the benefit of thorough consideration and editing, including letters, emails, and even news stories like this one. It enables us to communicate in a more organized and intentional way, leaving a permanent record of our ideas and intentions.

The human inclination for visual processing is tapped into through visual communication, whether it be through pictures, posters, or television. It defies linguistic boundaries, generating feelings and delivering messages instantly. Images have the ability to draw in viewers, tell tales, and forge connections. In the digital age, when platforms like social media mainly rely on visual information to express ideas, visual communication has grown more commonplace [2].

Effective communication skills are more important than ever in this age of cutting-edge technology and worldwide connectedness. In both personal and professional contexts, the capacity to use various communication channels and adapt to various audiences is crucial. Understanding the complex nature of communication allows us to hone our abilities and improve our ability to communicate ourselves and understand others.

In recent years, social media has become an integral part of our daily lives, providing us with a platform to

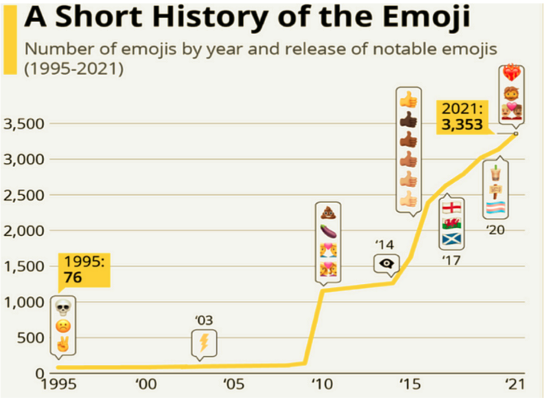


Figure 1 History of Emoji

[3]

connect with others and share our thoughts, feelings, and opinions on various topics. There are numerous social networking platforms such as Twitter, Facebook, Instagram, and WhatsApp that have been created to facilitate communication and connection between individuals. Despite their differences in terms of features and capabilities, these platforms all share one common aspect: the use of emojis. Emojis, which include everything from facial expressions to animals, objects, and places, are widely used to communicate simple ideas and to enhance the emotions and feelings conveyed in our messages. However, the meaning of emojis is not always the same, and their use can vary depending on the context and culture. As a result, processing emojis remains a significant challenge for researchers in the field of natural language processing (NLP). In order to improve the effectiveness of NLP models in interpreting and generating emojis, researchers are constantly working on developing new techniques and algorithms that can better understand the nuances of this visual form of communication. [4].

Furthermore, emojis also play a crucial role in understanding the sentiment and emotion behind text-based communication on social media platforms. Emojis can convey a wide range of emotions, from happiness and excitement to sadness and anger, and can also be used to add sarcasm or irony to a message. Understanding the sentiment and emotion behind text-based communication is crucial in various applications such as sentiment analysis, customer service, and marketing. However, emojis can also introduce ambiguity and subjectivity, making it more difficult to determine the sentiment and emotion behind a message.

Overall, emojis have become an essential part of communication on social media platforms, and their use has brought new challenges for NLP researchers. While understanding and processing emojis can be difficult, research in this area continues to evolve, and new techniques and algorithms are being developed to better understand the nuances of this visual form of communication.

II. Related Work

Here we will talk about related studies.

There are numerous social media-related topics that can be studied and added to the literature because individuals use social media to communicate and express their likes and opinions. Emoji estimate research is one of them, and it will also help with sentiment analysis. Nowadays, emojis are used in practically all sentences. In most cases, it can even take the place of words. Because technology is increasingly more effective at communicating information and feelings to us than words. In fact, due to its widespread use, it has also developed its own universal language.

Social network analysis, which has been the subject of many scientific studies in recent years, is one of the most interesting topics for NLP researchers. Many studies have been conducted in this area. The Semeval data has been used as the data set for many of these studies on sentiment analysis or emoji prediction [5].

According to the article written by the team that collected the data set, SVM has been emphasized to make more accurate inferences compared to other CNN and LSTM methods [6].

The study by Cagri Coltekin and Taraka Rama, working on the same data set, was titled "SVMs perform better than RNNs at Emoji Prediction" and concluded that SVM makes more accurate calculations than RNN methods [7]. Another study on the same data set found that Naive Bayes produced better results than RNN algorithms [8].

In addition to these, there are also studies that emphasize that MNB results produce significantly better results compared to complex deep learning systems [9] and that a series of pre-studies are needed for RNN methods to achieve higher success and that complex layered deep learning architectures are modeled [10].

Our study use the same data was taken from Kaggle. There is a dataset with a nearly total of 70000 tweets. The data set has been labeled with emojis. The data contains 20 label assignments. The majority of these are hearts, along with smiling, sunglass-wearing faces, winking faces, Christmas trees, etc. First, the data set's distribution of tweets based on these emoji labels was looked at. It was noted that there is no regular distribution in this case and that some emoji classes have more classes than others. Four classes were left after combining emoji labels with comparable expressions. These classes' emoji names are: Love, Joy, Vacation, and Sadness (Dark Heart).

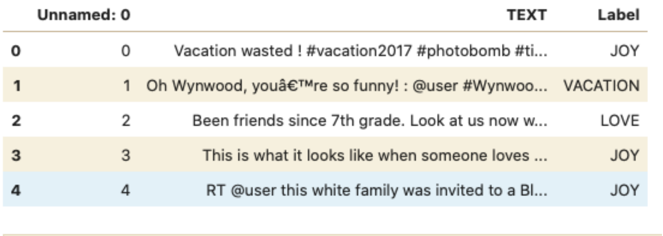
III. Data Preprocessing

Here, we will discuss the data processing steps.

A. Data Set

Our data set is commonly used as a data set in scientific studies, especially in NLP studies.

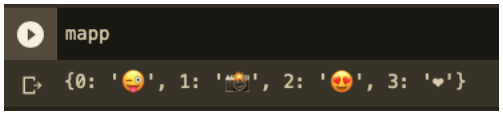
As can be seen, the data is not synthetic and consists of natural, non-artificial data. It has been quite difficult to work with data that is difficult for people who are not familiar with the target audience of the tweets to understand or analyze, and correctly transfer this data to our model. The cleaned version of the data set is shown in Figure 2.



	Unnamed: 0	TEXT	Label
0	0	Vacation wasted ! #vacation2017 #photobomb #ti...	JOY
1	1	Oh Wynwood, youâ€™re so funny! : @user #Wynwoo...	VACATION
2	2	Been friends since 7th grade. Look at us now w...	LOVE
3	3	This is what it looks like when someone loves ...	JOY
4	4	RT @user this white family was invited to a Bl...	JOY

Figure 2 Cleaned Example

When we analyze the data set, the results obtained are not very reliable because the data is actually real-world data. For example, there is the word "baaaack" in the data set, but it is known that the vector values (vectorization) or the numerical results (tokenization) of the words "back" and "baaaack" are different. However, the words are actually used in a similar sense in context. The same can be said for the words "looooooooveeee" and "love". In other examples, there are meaningless words such as "ough" or "ummmm". These words will have a negative impact on the results because they do not have much meaning. After preproccess some sentences have very few words left. This is also a situation that makes it difficult to extract meaning. Therefore, it will be quite difficult to interpret these data using ML methods because the data set consists of natural and real data.



```
mapp
{0: '😜', 1: '📸', 2: '😍', 3: '❤'}
```

Figure 3 Emojis with Indices

Our data set consists of 20 classes. But we filtering 4 common classes as you can see in Figure 3. These classes correspond to our emojis. When we examine the data set, although it consists of 4 classes, the class distribution is not much imbalanced as shown in Figure 4. It is very difficult to model the training correctly on imbalanced data. If our remaining dataset after cleaning was in an unbalanced state, balancing could be achieved with several oversampling methods.

B. Tweet Pre-Proccessing

We need to removed hashtags and "@" symbols with their corresponding words using the tweet preprocessor. These elements may confuse the meaning in the prediction of emojis, so they should not be present in our data set. After that we removed stopwords such as "I, we, can, do. . . " since our data set is in English. In the third step, we removed extra spaces, meaningless single letters, and repeated words or sentences, and our data set was cleaned.

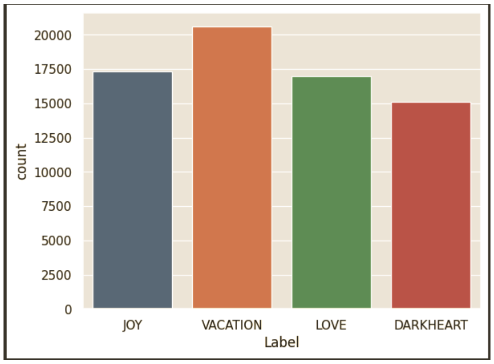
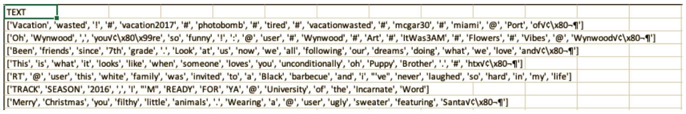


Figure 4 Label Ratios

1) NLTK: NLTK stands for Natural Language Toolkit. It is a popular Python library that provides various tools and resources for working with human language data, particularly in the field of natural language processing (NLP) [11].

The "averaged_perceptron_tagger" and "stopwords" modules from the NLTK library have been downloaded for preprocessing and stopword removal. As mentioned above, stopwords specific to the English language, as well as meaningless characters such as "@", "", have been removed from our dataset using NLTK, and our word corpus has been created using NLTK

There is a significant difference between the data preprocessing before and after, as shown in the Figure 5 and Figure 6 that displays an example of our data before preprocessing and after cleaning.



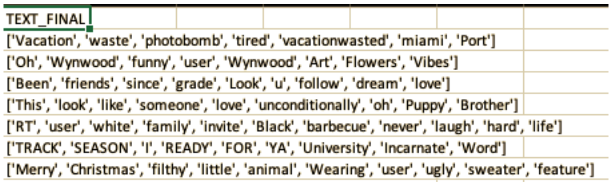
TEXT
['Vacation', 'wasted', '!', '#', 'vacation2017', '#', 'photobomb', '#', 'tired', '#', 'vacationwasted', '#', 'mcgar30', '#', 'miami', '@', 'Port', 'of√¢\x80¬¶']
['Oh', 'Wynwood', ',', 'you√¢\x80\x99re', 'so', 'funny', '!', ':', '@', 'user', '#', 'Wynwood', '#', 'Art', '#', 'ItWas3AM', '#', 'Flowers', '#', 'Vibes', '@', 'Wynwood√¢\x80¬¶']
['Been', 'friends', 'since', '7th', 'grade', '.', 'Look', 'at', 'us', 'now', 'we', 'all', 'following', 'our', 'dreams', 'doing', 'what', 'we', 'love', 'and√¢\x80¬¶']
['This', 'is', 'what', 'it', 'looks', 'like', 'when', 'someone', 'loves', 'you', 'unconditionally', 'oh', 'Puppy', 'Brother', '.', '#', 'htx√¢\x80¬¶']
['RT', '@', 'user', 'this', 'white', 'family', 'was', 'invited', 'to', 'a', 'Black', 'barbecue', 'and', 'i', "'ve", 'never', 'laughed', 'so', 'hard', 'in', 'my', 'life']
['TRACK', 'SEASON', '2016', ',', 'I', "'M", 'READY', 'FOR', 'YA', '@', 'University', 'of', 'the', 'Incarnate', 'Word']
['Merry', 'Christmas', 'you', 'filthy', 'little', 'animals', '.', 'Wearing', 'a', '@', 'user', 'ugly', 'sweater', 'featuring', 'Santa√¢\x80¬¶']

Figure 5 Uncleaned Corpus

After that NLTK "WordNet" is used to tag the words in the sentence as adjective, adverb, noun, or verb. Natural language processing and computational linguistics both benefit from the use of WordNet, a lexical database and semantic network. It’s a sizable, manually curated library of English words arranged into groups called synsets, which are collections of synonyms that each reflect a different concept. WordNet establishes associations between words, such as hyponymy (part of relationship) and hypernymy (is-a relationship), in addition to providing definitions for individual words. This hierarchical structure facilitates semantic navigation and supports operations like sentiment analysis, information retrieval, and word sense disambiguation. The widespread use of WordNet in several applications and research projects has helped in the advancement of semantic comprehension and knowledge representation in computer systems [12].

2) Lemmatization: By eliminating the stop words when developing the TF-IDF model, the preparation work on the data is continued. Word vectors were created using the

Wordnet corpus. Nouns, verbs, adjectives, and adverbs are all assigned to each word. The data set has the appearance seen in the Figure 6 following the lemmatization process. The word vectors before and after the lemmatization procedure are shown in the TEXT_FINAL column for each sentence.



TEXT_FINAL
['Vacation', 'waste', 'photobomb', 'tired', 'vacationwasted', 'miami', 'Port']
['Oh', 'Wynwood', 'funny', 'user', 'Wynwood', 'Art', 'Flowers', 'Vibes']
['Been', 'friends', 'since', 'grade', 'Look', 'u', 'follow', 'dream', 'love']
['This', 'look', 'like', 'someone', 'love', 'unconditionally', 'oh', 'Puppy', 'Brother']
['RT', 'user', 'white', 'family', 'invite', 'Black', 'barbecue', 'never', 'laugh', 'hard', 'life']
['TRACK', 'SEASON', 'I', 'READY', 'FOR', 'YA', 'University', 'Incarnate', 'Word']
['Merry', 'Christmas', 'filthy', 'little', 'animal', 'Wearing', 'user', 'ugly', 'sweater', 'feature']

Figure 6 Cleaned Corpus

3) Test & Train Split: It is crucial to evaluate a machine learning model's performance using new, unexplored data when developing one. We can train our model on part of the data while testing its effectiveness on the remaining unobserved data by dividing the available data into training and test sets. This method enables us to test the model's ability to make accurate predictions by simulating real-world settings where it encounters novel observations.

The data set is divided into training and testing after this procedure. There are 20% of test data and 80% of training data. It was not arbitrary to decide to split the data into an 80% training and 20% test split; rather, it was done in accordance with industry best practices. This division finds a balance between having enough test data for a thorough evaluation and having enough data for the model's training.

We guarantee that the model gains knowledge from a sizeable percentage of the dataset by designating 80% of the data for training. This larger training set helps in the model's ability to recognize underlying relationships, patterns, and nuances in the data, improving performance and generalization.

An objective assessment set is created using the remaining 20% of the data that was set away for testing. It has new information that can be used to evaluate how well the model works with previously unexplored data. This test set provides a benchmark for evaluating how effectively the model generalizes to new data and helps in the detection of any problems like overfitting or underfitting [13].

The train and test data were then converted from text to numeric data using LabelEncoder.

4) Vectorization: Vectorization refers to the process of converting data into numerical vectors that can be used in machine learning algorithms. LabelEncoder helps this vectorization. This process is necessary because most machine learning algorithms cannot operate on data in its raw form, such as text or images. Instead, the data must be transformed into a numerical representation that the algorithms can understand. There are several ways to perform vectorization, including count vectorization (CV) , term frequency-inverse document frequency (TF-IDF) , and word embeddings.

TF-IDF (Term Frequency - Inverse Document Frequency) and Counter Vectorizer are two methods used to represent text data in numerical form, which can then be used as input for machine learning models. As you can see the formulas of TF-IDF with figure 7.

The main difference between the two is that Counter Vectorizer simply counts the number of occurrences of each word in the document, while TF-IDF also takes into account the frequency of each word in the entire corpus of documents. This means that the resulting vectors for each document will have higher values for words that are more unique to that document, and lower values for words that are more common across the corpus.

TF-IDF is generally considered to be a better representation of the importance of each word in a document, as it down-weights words that are very common across the entire corpus. This can be useful when the goal is to identify the main topics or themes in a set of documents, as common words that do not contribute much to the meaning of the document will have lower weights. Counter Vectorizer, on the other hand, is often used when the goal is simply to identify the most common words in a set of documents [14].

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,j}}$$

TF Formula

$$idf(w) = log(\frac{N}{df_t})$$

Figure 7 TF-IDF Formula

The IDF value stays the same until new documents are added, while the TF value will change for every word in every sentence. It is therefore determined specifically for each sentence. We combined the Count Vectorizer and TF-IDF Vectorizer. The architecture of the pipeline provides this unity. Due to this architecture, extremely quick reactions were possible. Naive Bayes training typically takes a short time, whereas SVM training often takes a considerable time. However, with the pipeline architecture, training is finished virtually simultaneously.

IV. Machine Learning Models

Here, we discussed the some machine learning methods used during the training of the model and we will provide information about machine learning.

A. Multinominal Naive Bayes

Multinomial Naive Bayes (MNB) is a classification algorithm that is based on the Naive Bayes theorem, which states that the probability of an event is equal to the probability of each feature occurring independently. MNB is an extension of the standard Naive Bayes algorithm that is used when the features are discrete and the values that they can take are limited. It is commonly used in text classification tasks, where the features are the occurrences of certain words or phrases, and the values are the number of times they occur in a given document. MNB is known for being simple and fast to train, and it is often used as a baseline model in machine learning tasks [15].

Multinomial The term "frequency" in Naive Bayes refers to how frequently a specific term appears in a document. The document length is divided to standardize the term frequency. Term frequency can be used to construct maximum likelihood estimates based on the training data to estimate the conditional probability after the normalization step[16].

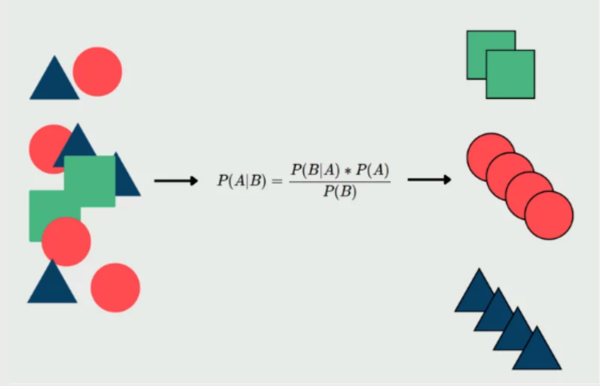
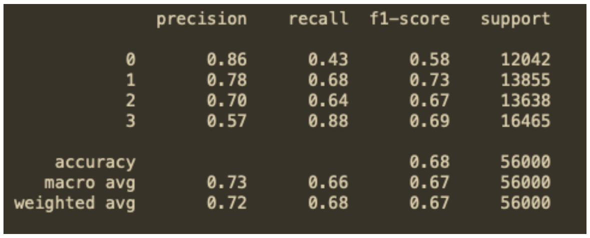


Figure 8 Naive Bayes Formula

Both discrete and continuous data types can be processed using this approach. It is very scalable and simple to implement. As a result, handling huge datasets is simple. Despite these benefits, compared to other probability algorithms, the prediction rate could still be low. It works well for textual categorization but is unsuitable for regression [16].

In our study, we have shared the results of MNB.

Multinominal Naive Bayes Results



```
              precision    recall  f1-score   support

           0       0.86      0.43      0.58     12042
           1       0.78      0.68      0.73     13855
           2       0.70      0.64      0.67     13638
           3       0.57      0.88      0.69     16465

    accuracy                           0.68     56000
   macro avg       0.73      0.66      0.67     56000
weighted avg       0.72      0.68      0.67     56000
```

Figure 9 MNB Scores

As you can see figure 9:

- Precision: It measures the proportion of correctly predicted instances for each class. A higher precision value indicates a lower false positive rate. In this case, class 0 has a precision of 0.86, class 1 has a precision of 0.78, class 2 has a precision of 0.70, and class 3 has a precision of 0.57.
- Recall: It measures the proportion of correctly predicted instances out of the actual instances for each class. A higher recall value indicates a lower false negative rate. In this case, class 0 has a recall of 0.43, class 1 has a recall of 0.68, class 2 has a recall of 0.64, and class 3 has a recall of 0.88.
- F1-score: It is the harmonic mean of precision and recall, providing a single metric that balances both measures. A higher F1-score indicates a better trade-off between precision and recall. In this case, class 0 has an F1-score of 0.58, class 1 has an F1-score of 0.73, class 2 has an F1-score of 0.67, and class 3 has an F1-score of 0.69.
- Support: It represents the number of instances in each class.
- Accuracy: It measures the overall correctness of the model by calculating the ratio of correctly predicted instances to the total number of instances. In this case, the overall accuracy of the model is 0.68, which means it correctly predicts 68% of the instances.
- Macro average: It calculates the average performance across all classes, giving equal weight to each class. In this case, the macro average precision is 0.73, recall is 0.66, and F1-score is 0.67.
- Weighted average: It calculates the average performance across all classes, giving each class a weight proportional to its support. In this case, the weighted average precision is 0.72, recall is 0.68, and F1-score is 0.67.

As shown in Figure 10, a matrix comparing the predicted and actual emojis is provided. When looking at this matrix, it can be observed that emoji number 3 (dark heart), which has numerically fewer occurrences, is surprisingly predicted more accurately compared to the others. This indicates that emoji number 3 is used in more distinctive sentences and with different words compared to the others.

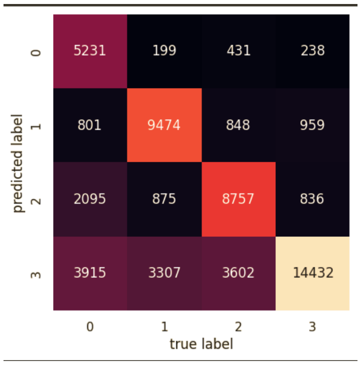


Figure 10 MNB Confusion Matrix

B. *Support Vector Machines*

Support Vector Machines (SVM) is a type of supervised learning algorithm that can be used for classification or regression tasks. It works by finding the hyperplane in a high-dimensional space that maximally separates the data points of different classes. SVM tries to maximize the margin between the two classes, meaning that the distance between the hyperplane and the nearest data points of each class should be as large as possible. In the case of non-linear boundaries, SVM can use kernels to transform the input data into a higher-dimensional space where the hyperplane can be found [17].

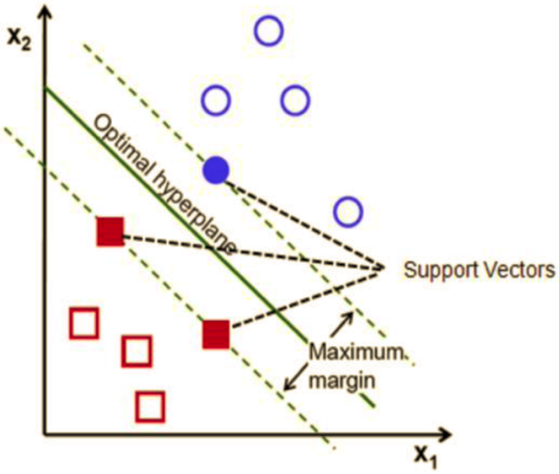


Figure 11 SVM Example

Figure 11 shows two distinct classes, blue and red. Choosing which class the new data will belong to is the primary categorization challenge. The margin is the 1 space along the line that divides the two classes. For classes 2 and up, the greater the margin, the more precise the categorization will be.

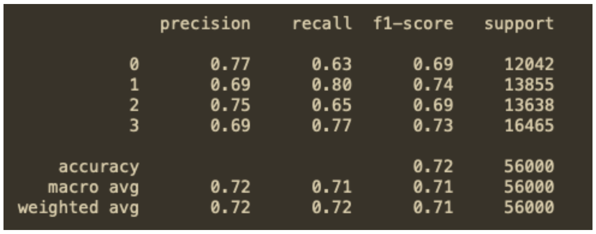
In the context of Support Vector Machines (SVMs) , the kernel is a function that takes in two input data points and returns a single number, indicating how similar or dissimilar the data points are. The kernel function is used to transform the data into a higher-dimensional space, so that it can be separated by a hyperplane. Some common examples of kernel functions include the linear kernel, polynomial kernel, and radial basis function (RBF) kernel. The choice of kernel function can have a significant impact on the performance of an SVM model.

here are several types of kernels that can be used with Support Vector Machines (SVM) . Some of the most common ones are:

- Linear kernel: This kernel is used when the data is linearly separable. It is the simplest kernel and is used as a baseline for comparison.
- Polynomial kernel: This kernel can be used when the data is not linearly separable. It transforms the input data into higher-dimensional space and allows for the data to be separated by a polynomial function.
- Radial basis function (RBF) kernel: This kernel is a popular choice for non-linear classification problems. It uses the distance between points in the input space to transform the data into higher-dimensional space.
- Sigmoid kernel: This kernel is similar to the RBF kernel, but it uses a sigmoid function instead of a radial basis function to transform the data into higher-dimensional space.

In our study, we use SVM linear kernel model.

Support Vector Machines Results



```
              precision    recall  f1-score   support

           0       0.77      0.63      0.69     12042
           1       0.69      0.80      0.74     13855
           2       0.75      0.65      0.69     13638
           3       0.69      0.77      0.73     16465

    accuracy                           0.72     56000
   macro avg       0.72      0.71      0.71     56000
weighted avg       0.72      0.72      0.71     56000
```

Figure 12 SVM Scores

As you can see figure 12:

- Precision: It measures the proportion of correctly predicted instances for each class. A higher precision value indicates a lower false positive rate. In this case, class 0 has a precision of 0.77, class 1 has a precision of 0.69, class 2 has a precision of 0.75, and class 3 has a precision of 0.69.
- Recall: It measures the proportion of correctly predicted instances out of the actual instances for each class. A higher recall value indicates a lower false negative rate. In this case, class 0 has a recall of 0.63, class 1 has a recall of 0.80, class 2 has a recall of 0.65, and class 3 has a recall of 0.77.
- F1-score: It is the harmonic mean of precision and recall, providing a single metric that balances both measures. A higher F1-score indicates a better trade-off between precision and recall. In this case, class 0 has an F1-score of 0.69, class 1 has an F1-score of 0.74, class 2 has an F1-score of 0.69, and class 3 has an F1-score of 0.73.
- Support: It represents the number of instances in each class.
- Accuracy: It measures the overall correctness of the model by calculating the ratio of correctly predicted instances to the total number of instances. In this case, the overall accuracy of the model is 0.72, which means it correctly predicts 72
- Macro average: It calculates the average performance across all classes, giving equal weight to each class. In this case, the macro average precision is 0.72, recall is 0.71, and F1-score is 0.71.
- Weighted average: It calculates the average performance across all classes, giving each class a weight proportional to its support. In this case, the

weighted average precision is 0.72, recall is 0.72, and F1-score is 0.71.

As seen in Figure 13, a matrix comparing the predicted and actual emojis is provided. When looking at this matrix, it can be observed that, like the others, the prediction of emoji number 3 (dark heart) is more accurately detected compared to the others.

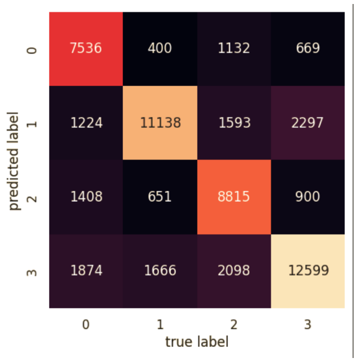


Figure 13 SVM Confusion Matrix

V. Conclusion

Here, we will provide information about the conclusion.

The two predictions seem to succeded. The fact that the full dataset is not made up of English-language tweets is one of the factors influencing this finding. The sentences also contain multilingual Italian, French, and Chinese vocabulary. In addition, the sarcasm problem, which affects sentiment analysis in general, plays a role in this. The success rate may be impacted by statements that we mistakenly categorize as joyful but actually contain irony.

Apart from all these factors, the distribution of emoji in the dataset is the biggest problem. Since the classes are mixed here, it cannot be argued that the correct data set is entirely produced, despite attempts to fix this issue by lowering the class.

https://www.overleaf.com/project/6478b957a77a37c55023adb0

You can see the comparison of the confusion matrix tables of SVM and MNB in Figures 14 and 15. Looking at the tables, although the scores are almost very close to each other, there is a difference in the prediction rates of some emojis between MNB and SVM. First of all, it should be noted that the emoji numbered 3 (dark heart) has a higher prediction rate compared to others because it carries a more negative emotion than other emojis. Therefore, the words in the sentences where the 3rd emoji is used should be semantically very distant from the words where other emojis are used.

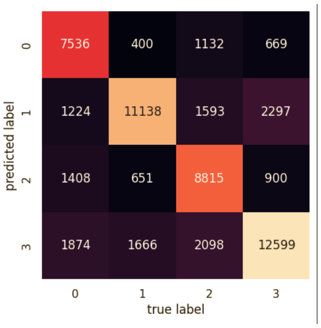


Figure 14 SVM Predictions.

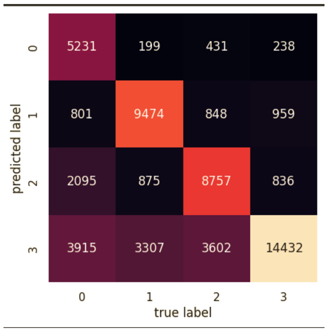


Figure 15 MNB Predictions.

In addition, the emoji numbered 1 (Vacation) was predicted more accurately with the SVM linear kernel. Similarly, the accuracy of the emoji numbered 0 (joy) is more clear in SVM.

It should also be noted that MNB has a higher number of predictions for the 3rd emoji (dark heart). In short, MNB has outputted the 3rd emoji as a prediction in most cases. This has hindered it from predicting the emojis numbered 0, 1, and 2 correctly.

When considering all of them together, it can be said that successful models have been created for emoji prediction using SVM and MNB.

References

[1] P. A. Andersen and L. K. Guerrero, Eds., *Handbook of Communication and Emotion: Research, Theory, Applications, and Contexts*. Routledge, 2020.

[2] Ş. Ş. Demir, "The effects of communication techniques on public relation activities: A sample of hospitality business," *Journal of Human Sciences*, vol. 8, no. 2, pp. 127–150, 2011.

[3] A. Sampietro, "Emojis and the performance of humour in everyday electronically-mediated conversation: A corpus study of whatsapp chats," *Internet Pragmatics*, vol. 4, no. 1, pp. 87–110, 2021.

[4] F. Barbieri, J. Camacho-Collados, F. Ronzano, L. Espinosa-Anke, M. Ballesteros, V. Basile, V. Patti, and H. Saggion, "SemEval-2018 Task 2: Multilingual Emoji Prediction," in *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018)*. New Orleans, LA, United States: Association for Computational Linguistics, 2018.

[5] F. Barbieri, J. Camacho-Collados, F. Ronzano, L. E. Anke, M. Ballesteros, V. Basile, V. Patti, and H. Saggion, "Semeval 2018 task 2: Multilingual emoji prediction," in *Proceedings of the 12th international workshop on semantic evaluation*, 2018, pp. 24–33.

[6] F. Barbieri, J. Camacho-Collados, F. Ronzano, L. Espinosa-Anke, M. Ballesteros, V. Basile, V. Patti, and H. Saggion, "SemEval 2018 task 2: Multilingual emoji prediction," in *Proceedings of the 12th International Workshop on Semantic Evaluation*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 24–33. [Online]. Available: https://aclanthology.org/S18-1003

[7] Ç. Çöltekin and T. Rama, "Tübingen-oslo at semeval-2018 task 2: Svms perform better than rnns in emoji prediction," in *Proceedings of The 12th International Workshop on Semantic Evaluation*, 2018, pp. 34–38.

[8] L. Alexa, A. B. Lorent, D. Gifu, and D. Trandabat, "The dabblers at semeval-2018 task 2: Multilingual emoji prediction," in *Proceedings of The 12th International Workshop on Semantic Evaluation*, 2018, pp. 405–409.

[9] S. Jin and T. Pedersen, "Duluth urop at semeval-2018 task 2: Multilingual emoji prediction with ensemble learning and oversampling," *arXiv preprint arXiv:1805.10267*, 2018.

[10] C. Baziotis, N. Athanasiou, G. Paraskevopoulos, N. Ellinas, A. Kolovou, and A. Potamianos, "Ntua-slp at semeval-2018 task 2: Predicting emojis using rnns with context-aware attention," *arXiv preprint arXiv:1804.06657*, 2018.

[11] S. Bird, "Nltk: the natural language toolkit," in *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, 2006, pp. 69–72.

[12] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, "Introduction to wordnet: An on-line lexical database," *International journal of lexicography*, vol. 3, no. 4, pp. 235–244, 1990.

[13] R. Koch, *The 80/20 Principle: The Secret of Achieving More with Less: Updated 20th anniversary edition of the productivity and business classic*. Hachette UK, 2011.

[14] K. Maity, A. Kumar, and S. Saha, "A multitask multimodal framework for sentiment and emotion-aided cyberbullying detection," *IEEE Internet Computing*, vol. 26, no. 4, pp. 68–78, 2022.

[15] L. Jiang, S. Wang, C. Li, and L. Zhang, "Structure extended multinomial naive bayes," *Information Sciences*, vol. 329, pp. 346–356, 2016.

[16] M. Abbas, K. A. Memon, A. A. Jamali, S. Memon, and A. Ahmed, "Multinomial naive bayes classification model for sentiment analysis," *IJCSNS Int. J. Comput. Sci. Netw. Secur*, vol. 19, no. 3, p. 62, 2019.

[17] D. K. Jain, A. Kumar, and S. R. Sangwan, "Tana: The amalgam neural architecture for sarcasm detection in indian indigenous language combining lstm and svm with word-emoji embeddings," *Pattern Recognition Letters*, vol. 160, pp. 11–18, 2022.