# Abstract

Contents

[Introduction 3](#_Toc44766165)

[Theoretical elements 3](#_Toc44766166)

[Python programming language 3](#_Toc44766167)

[Object Oriented Programming 3](#_Toc44766168)

[SQL 4](#_Toc44766169)

[Technologies 4](#_Toc44766170)

[PyCharm 4](#_Toc44766171)

[PyTorch 4](#_Toc44766172)

[SQL Server 4](#_Toc44766173)

[State of the art 5](#_Toc44766174)

[Machine learning 5](#_Toc44766175)

[Keyword-based detection 5](#_Toc44766176)

[Hybrid detection 5](#_Toc44766177)

[Description of the Application 9](#_Toc44766178)

[References 10](#_Toc44766179)

# Introduction

# Theoretical elements

## Python programming language

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

## Object Oriented Programming

Object-oriented Programming, or OOP for short, is a programming paradigm which provides a means of structuring programs so that properties and behaviors are bundled into individual objects.

Focusing first on the data, each thing or object is an instance of some class. The primitive data structures available in Python, like numbers, strings, and lists are designed to represent simple things like the cost of something, the name of a poem, and your favorite colors, respectively.

Classes are used to create new user-defined data structures that contain arbitrary information about something. In the case of an animal, we could create an Animal() class to track properties about the Animal like the name and age.

It’s important to note that a class just provides structure—it’s a blueprint for how something should be defined, but it doesn’t actually provide any real content itself. The Animal() class may specify that the name and age are necessary for defining an animal, but it will not actually state what a specific animal’s name or age is.

While the class is the blueprint, an instance is a copy of the class with actual values, literally an object belonging to a specific class. It’s not an idea anymore; it’s an actual animal, like a dog named Roger who’s eight years old.

Inheritance is the process by which one class takes on the attributes and methods of another. Newly formed classes are called child classes, and the classes that child classes are derived from are called parent classes.

It’s important to note that child classes override or extend the functionality (e.g., attributes and behaviors) of parent classes. In other words, child classes inherit all of the parent’s attributes and behaviors but can also specify different behavior to follow.

## SQL

SQL ( Structured Query Language) is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS). It is particularly useful in handling structured data, i.e. data incorporating relations among entities and variables.

# Technologies

## PyCharm

PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Python language. It is developed by the Czech company JetBrains. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django as well as Data Science with Anaconda.

PyCharm is cross-platform, with Windows, macOS and Linux versions. Features include:

* Coding assistance and analysis, with code completion, syntax and error highlighting, linter integration, and quick fixes
* Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages
* Python refactoring: includes rename, extract method, introduce variable, introduce constant, pull up, push down and others
* Integrated Python debugger

## PyTorch

PyTorch is an open source machine learning library based on the Torch library, used for applications such as computer vision and natural language processing, primarily developed by Facebook's AI Research lab (FAIR). It is free and open-source software released under the Modified BSD license. Although the Python interface is more polished and the primary focus of development, PyTorch also has a C++ interface.

PyTorch provides two high-level features:

- Tensor computing (like NumPy) with strong acceleration via graphics processing units (GPU)

- Deep neural networks built on a tape-based automatic differentiation system

## SQL Server

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run either on the same computer or on another computer across a network (including the Internet).

Microsoft SQL Server also allows user-defined composite types (UDTs) to be defined and used. It also makes server statistics available as virtual tables and views (called Dynamic Management Views or DMVs). In addition to tables, a database can also contain other objects including views, stored procedures, indexes and constraints, along with a transaction log. The data in the database are stored in primary data files with an extension .mdf. Secondary data files, identified with a .ndf extension, are used to allow the data of a single database to be spread across more than one file, and optionally across more than one file system.

# State of the art

In order to begin talking about the current techniques employed to achieve emotion recognition, we have to understand the main idea behind the process. There are multiple studies which classify emotions in different ways, whether we talk about Ekman’s 6 categories(anger, disgust, fear, happiness, sadness, surprise), Izard’s 9 emotions(anger, disgust, fear, guilt, joy, sadness, shame, surprise) or even custom ways of categorizing emotions. This emotions can be manually extracted from text by using our natural intuition, but the problem is about how to do it with a machine where that intuition doesn’t exist.

The methods used in order to classify emotions are: Machine Learning, Keyword-based detection and Hybrid approaches.

## Machine learning

Using this method there are a wide variety of ways in which you can achieve your goal. Multiple methods from statistics are employed, such as support vector machine, feature extraction methods(“bag-of-words” or POS tagging) and also deep learning. Later into the process, a classifier is used to sort out and produce a result, the most used classifier being Naïve Bayes(or some variations of it)

## Keyword-based detection

Here, classifying emotions is done by searching for the emotional keywords in the input sentence. This requires a very big dataset which is well classified prior to using it and then matching the words found in your sentence to the ones in the dataset, thus resulting in the emotions that we are looking for.

However, classification methods based on only keywords suffer from the ambiguity in the keyword definitions in the sense that a word can have different meanings according to usage and context, the incapability of recognizing emotions within sentences that do not contain emotional keywords, and the lack of linguistic information.

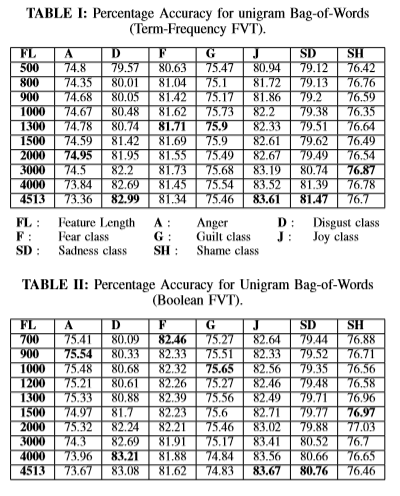
## Hybrid detection

This one is the most common way of doing emotion analysis thanks to its proven efficiency (around 75 – 80 % in most cases) and also to its flexibility. A few hybrid approaches with high efficiency where found in order to exemplify the place where we are right now regarding the technology and the ideas behind them are going to be presented.

In (Bincy Thomas, 2014) there is a method that showed an accuracy of 79% while comparing it to similar other methods. What it proposes is doing a feature extraction using “bag-of-words” with stemming, which is supposed to reduce the amount of feature space(stemming removes various forms and inflections in words). Compared to this, also POS tagging and lexical emotion words methods have been tried but they didn’t prove as effective.

After this step, feature selection has been done through 2 techniques: Mutual Information (measures the amount of information a variable holds about the other) and Term Frequency-Inverse Document Frequency (used to determine the words that are more favorable to use in a query in a corpus of documents. TF-IDF calculates for each word in a document is inverse proportion of the frequency of a word in a particular document to the percentage of documents they word appears)

As a last step, the classification is done on the synthesized feature space using a Mutinomial Naïve Bayes as an emotion classifier, thus resulting in the average 79% accuracy. (Bincy Thomas, 2014)

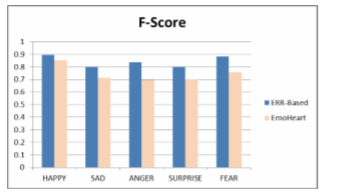


Another method proposed in (Shadi Shaheen, 2014) is by using an annotated dataset focusing on the six Ekman emotions for classifying blog texts. They propose that the sentences should be reduced to their core elements through compositionality principle (which states that in a meaningful sentence, if the lexical parts are taken out of the sentence, what remains are the rules of composition) and semantic and syntactic analysis.

The output of the syntactic and semantic analysis just mentioned is translated into their model to an intermediate representation of the sentence that they call Emotion Recognition Rule (ERR). ERRs are composed of four types of constructs: (1) verb-noun clauses (VNCs), (2) noun clauses (NNs), (3) adjectives (JJs), and adverbs (RBs). Using this constructs and analysis the relationships between them they found a way of describing the emotions.

To go through the annotation phase, they used the Stanford POS tagger and then the Stanford dependency parser in order to construct the dependency tree of the input sentence. On this tree, a set of rules where applied I order to capture the emotional part of the sentence.

For the classification part, a KNN classifier was used to compare the currently generated ERR to all the other ERRs in the Annotated set (and if the input was rejected, a PMI classifier would undertake the same job). Through this method they reached an impressive average of 84% accuracy, almost 10% higher than the state-of-the-art method of EmoHeart.



Other studies such as (Lea Canales, 2015) focused on presenting ways in which the process of Emotion Annotation can be done most efficiently. Their proposal was a semi-automatic methodology which they called EmoLabel that was split in 2 steps:

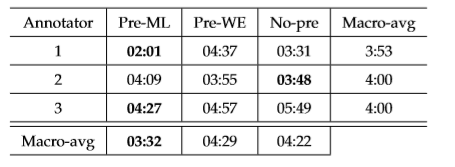
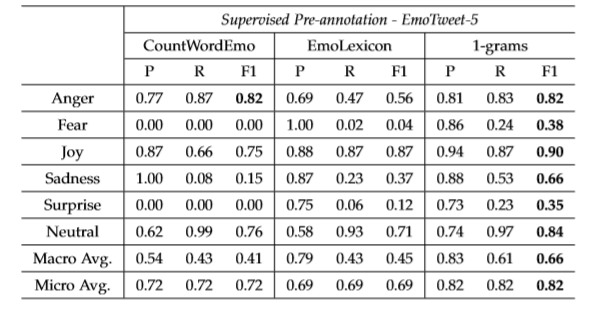
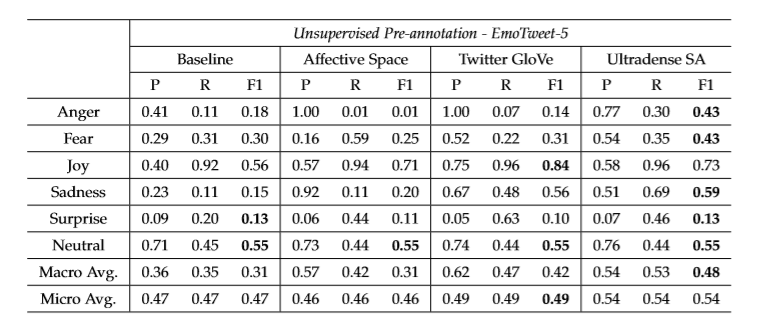
First step was the pre-annotation process where they compare 2 processes: an unsupervised approach based on Distributional Semantic Models (DSM’s) and a supervised method based on Machine Learning (ML). For input they both receive a collection of unlabeled sentences and a set of emotional categories (Ekman’s).

The first one is based on the idea that the meaning of a word can be inferred from its usage so this models build semantic vectors through a statistical analysis of the context in which the words occur, resulting in a word vector. For this approach, firstly the emotions and sentences are encoded into a semantic space with the help of distributional representations. Secondly the association begin between emotions and sentences(since now they are comparable thanks to the step before), then reordering(based on polarity) and selections is done(first 3 emotions based on polarity)

For the second method, 3 experiments have been done: Count-Emotion-Words-per-Emotion which contained a 8-feature array classification, 6 spots for the 6 emotions and the last 2 for polarity(resulted from Sentiment Analysis); Emotion-Lexicon-Words (EmoLexicon), a classification with features derived from the emotion lexicon and Unigrams (1-grams) which was a corpus-based classification. All three experiments employed a Support Vector Machine multi-class classifier.

In the second step, manual annotation was employed in order to evaluate the impact of the first step in matters of time and accuracy. All 3 types of pre-annotation where checked: with ML, with DSM and without any type of pre-annotation.

All this experiments showed in the end the using pre-ML results in the most accuracy of all the other methods presented when it come to pre-annotating.



Also worth mentioning is (Nadia Afroz, 2015) where they propose a framework based on Izar’s 9 emotions using a rule based approach and Data Wareshousing and Decision Support System.

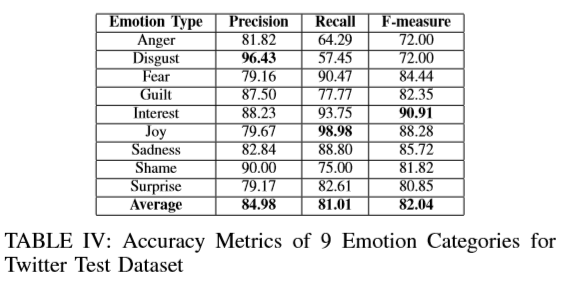
This warehouse contains words of the English language and their associated “intensity vector” which represents a set of 9 numbers ranging between 0 and 1 who describe the “closeness” of a word to 1 of the 9 emotions. Since this study focuses on internet text, more exactly it focuses on tweets, the warehouse also contains emoticons, abbreviations and modifiers.

The framework is composed of 2 modules: the Training Module and the Emotion Extraction Module.

The first one extracts data from twitter through Twitter4J API while focusing only on tweets in English language. Also extraction takes place out of User Query and already existing DW’s. After extraction is completed, the transformation phase begins where the framework cleans the input data(clear duplicates, remove “#” and “@” and so on) and marks it with a possible emotion class(they focus on the emotions that hashtags and emoticons represent). After this, the dataset is ready to train the system which uses the Stanford POS tagger on the text and then a Naïve Bayes classifier to compute the intensity of each emotion.

In the second module there are multiple sub-modules. The first one is the Emoticon and Abbreviation Analyzer whose responsibility is to get the emotions out of the 2 features. If it does not find anything then the second module will start: the Sentence Analyzer. This one first takes the dominant parts of a phrase(principal and subordinate clauses) and send them to a Phrase Level Analyzer which computes the intensity of the phrase(just like they compute for words) and after that Word Level Analysis takes place.

The last module consists of the Calculation. All the intensity vectors found from the Word Level Analysis and Phrase Level Analysis are added in this stage. Then the system determines the resultant emotion by checking the highest intensity.



The accuracy Average is of 82.04% for the Twitter test dataset which is slightly higher in comparison with other established systems.

Due to the fact that there is no strict rule that states how emotion recognition in text should be done, there are also a few viable ways found in Sentiment Mining. In the paper (Francesco Colace, 2014) they propose using a Mixed Graph of Terms which is basically constructed of terms(nodes) and the relationship between them(edges). In order to construct the Graph, an annotated lexicon is used (in this case WordNet) and the Latent Dirichlet Allocation algorithm. With these 2 and the pre-processing (tokenization, stopwords ﬁltering and stemming) the Graph is created and ready to use. Other papers presented general ways of achieving the means (Pankaj Gupta, 2016) through already mentioned techniques like using a bag of words and then a Naïve Bayes classifier and a Support Vector machine, while other focused on different ideas, like topic extraction (Vamshi Krishna, 2018) using LDA.

# 

# Description of the Application

# References

Bincy Thomas, D. K. (2014). Exploration of Robust Features for Multiclass Emotion Classiﬁcation.

Francesco Colace, M. D. (2014). Sentiment Mining through Mixed Graph of Terms.

Giannis Haralabopoulos, C. W. (2018). A Multivalued Emotion Lexicon Created and Evaluated by the Crowd.

Lea Canales, W. D.-B. (2015). EmoLabel: Semi-Automatic Methodology for Emotion Annotation of Social Media Text.

Nadia Afroz, M.-U. A. (2015). An Intelligent Framework for Text-to-Emotion Analyzer.

Pankaj Gupta, R. T. (2016). Sentiment Analysis and Text Summarization of Online Reviews: A Surve.

Shadi Shaheen, W. E.-H. (2014). Emotion Recognition from Text Based on Automatically Generated Rules.

Shivangi Chawla, M. M. (n.d.). An Ensemble-Classifier Based Approach for Multiclass Emotion Classification of Short Text.

Sreeja P. S., G. S. (2019). PERC-An Emotion Recognition Corpus for Cognitive Poems.

Tirath Prasad Sahu, S. A. (n.d.). Sentiment Analysis of Movie Reviews: A study on Feature Selection & Classification Algorithms.

Vamshi Krishna, D. A. (2018). Topic Model Based Opinion Mining and Sentiment Analysis .