Report

On

Emotion Detection On Text

By

Bhagyashree

MIT2018002

Under the supervision of:

Prof. U. S. Tiwari

Professor

IIIT-ALLAHABAD

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, ALLAHABAD

(A UNIVERSITY ESTABLISHED UNDER SEC.3 OF UGC ACT, 1956 VIDE NOTIFICATION NO. F.9-4/99-U.3 DATED 04.08.2000 OF THE GOVT. OF INDIA)

A CENTRE OF EXCELLENCE IN INFORMATION TECHNOLOGY ESTABLISHED BY GOVT. OF INDIA

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the M.Tech. Mini-Project Report entitled "Emotion analysis on text", being submitted as a part of Mid Semester Project Evaluation to the Department of Information Technology of Indian Institute Of Information Technology, Allahabad, is an authenticated record of my original work under the guidance and supervision of Prof. U.S. Tiwari from February 2019 to June 2019. I have adequately cited and referenced the original sources and have adhered to all principles of academic honesty and integrity.

Date: May 2019

Place: IIIT Allahabad

Signatures:

Bhagyashree

(MIT2018002)

CERTIFICATE FROM SUPERVISOR

This is to certify that the statement made by the candidates is correct to the best of my knowledge and belief. The project entitled “Emotion analysis On text” is a record of candidates’ work carried out by her under my guidance and supervision. I do hereby recommend that it should be accepted in the fulfilment of the requirements of the mini project at IIIT Allahabad.

Prof. U.S. Tiwari,

IIIT-Allahabad

May, 2019

ACKNOWLEDGEMENT

I would like to acknowledge and extend our heartfelt gratitude to Prof. U.S. Tiwari, Indian Institute of Information Technology Allahabad, who guided me through this project. His keen vital encouragement, superb guidance, and constant support are the motive force behind this project work. I would like to show my warm thanks to all my friends for their continuous motivation and encouragement during this project. I am very thankful to all the technical and non-technical staffs of the college for their assistance and cooperation.

Bhagyashree

MIT2018002

Table of Content

VISION

In this era of artificial intelligence, intelligent machines are being designed which are expected to perform tasks which can be performed by humans. We are making machines that are intelligent and smart just like human beings are. However, one thing machines can't do is, not able to understand by itself and can't identify various emotions just like humans can do. Despite so much progress in the field of machine learning and natural language processing, machines have still not become capable of detecting emotions clearly.

There are some basic form of expressing the emotions: Through facial expressions, speech or text. A lot work is being done in recognizing emotions through facial expressions and speech, yet we haven't been able to develop machines that can detect emotions in a text successfully.

In the era of internet world, most of the information exists in the form of text. Be it digital encyclopedias, websites, search engines; text constitutes around 70% of the information available to us. Hence, our aim is to make machines detect emotions hidden in every piece of textual information available.

OBJECTIVE

Through my project I aim to make machines detect emotions from any form of text, which constitutes about 70% of information available to us. Given any form of text, the machine will identify the specific emotions of happiness, sadness, anger, surprise, fear or love that the text expresses. Understanding emotions exposes us to many possibilities: personalized information generation, like advertisements, search results and so on, development of powerful human computer interaction machines and evolution of more intuitive and emotionally characterized text to speech systems.

MOTIVATION

Emotion Analysis is the broadly studied application of NLP (natural language processing) and Machine learning. The goal of Emotion Analysis is to extract important information regarding public opinion which helps to make the business decisions easily, political campaigns and to increase the product consumption.

A lot of work has been done in the field of sentiment analysis. However, it is a difficult task to study the emotion of a human mind. Thus, I have taken this project to study about the different emotions of a man kind. My goal is to find the emotions of a particular word, and sentence. What the overall impact of sentence is?

I aim to design such a machine which can analyse the emotions in text/sentence or even in paragraph and can tell the overall impact of the paragraph or a chapter of a book. It will help in understanding the human computer interaction.

Also in text classification the purpose of using text classification is to determine the subjective value of a text-document, i.e. how positive or negative is the content of a text document.

INTRODUCTION

Definitions about emotions and its categories have been an important research since long. However, till now no one has been able to answer the question "How many different emotions we have?" because human emotions are complex. Deriving one emotion from other is a quite complex task because it depends on various factors. It is not easy to read one's state of mind. It can be quite hard to determine that where one emotion starts and where it ends. However, one can have multiple emotion at once, so it becomes even more complex to determine.

According to Paul Ekman model, six basic emotions are concluded which are anger, disgust, fear, happiness, sadness, and surprise. Each emotion has its own neural network in the brain, and corresponding behavioural response.

These basic emotions which compose/combine to form what advanced emotions. For example: the emotion of excitement can be a combination of joy and surprise whereas the emotion of disgust can be a combination of anger and sadness.

There are many challenges while detecting emotion in text. One is context-dependence of emotions within text. A phrase can have element of *anger* without using the word "anger" or any of its synonyms. For example, the phrase "Shut up!"

Another challenge in emotion detection is the lack of a labelled emotion database. Another challenge faced is ambiguity in the text.

PROBLEM DEFINITION

I aim to make machines understand and detect emotions from any textual input. A text based emotion recognition system which give an overall sentiment of the sentence/paragraph in terms of positivity/negativity and neutrality and determine the overall emotion.

Given any textual input such as a word/phrase/sentence the machine should be able to give a percentage of various emotions that constitutes the input.

LITERATURE REVIEW

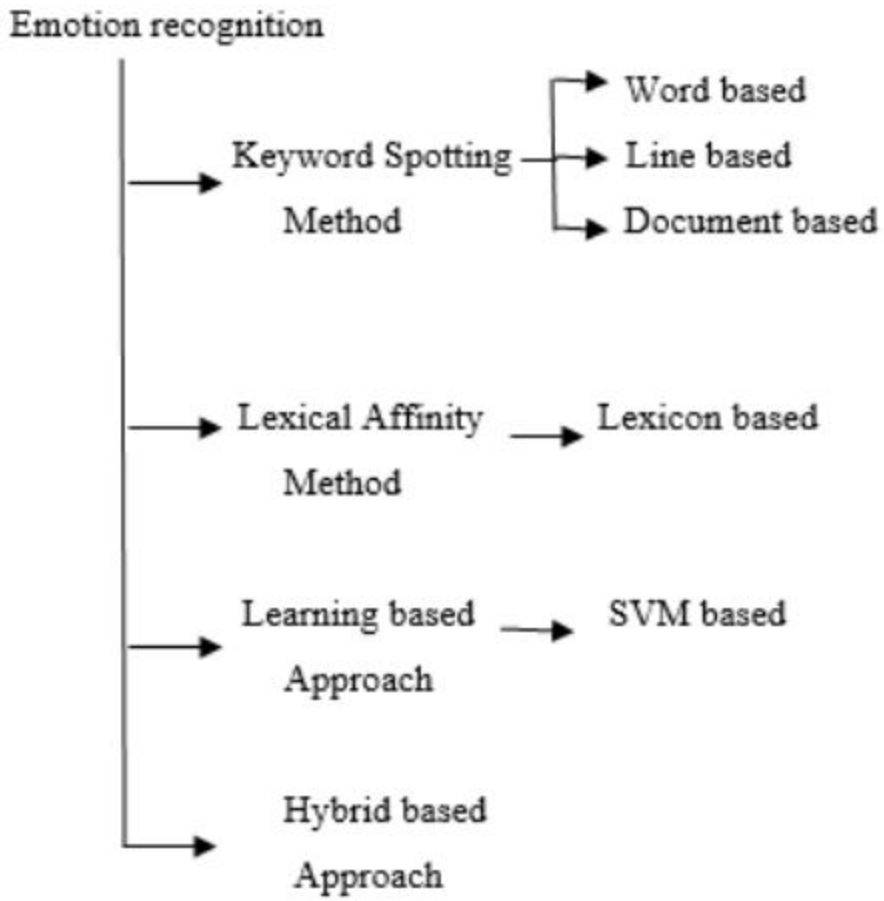
The field of emotion recognition has attracted many researchers from computer science, psychology, cognitive sciences, linguistics, and so on. Emotions can be expressed by a person through facial expressions, speech, gestures, and textual conversations. Speech and facial recognition have been the most preferred form for researchers all around the world for detecting emotions as they convey emotion more clearly without any disambiguates. Comparatively, very less work has been done in the field of emotion recognition through text.

Within text, considerable amount of work has been done in sentiment analysis of movie reviews, twitter feeds and so on. However, nothing significant has been achieved in detecting raw emotions of happiness, sadness, anger, fear, etc in text. The main reason for this is that there are no standard classifications of all human emotions due to complex nature of human minds and any emotion classifications can be seen as 'labels' annotated afterwards for different purposes. Since there is not any standard emotion-word hierarchy, focus is on related research about emotion in cognitive psychology domain.

Researchers have used techniques such as keyword spotting, assigning probabilistic affinities to various emotions, use of learning based methods and knowledge based artificial neural networks, rule based system to extract semantics related to specific emotions , detecting emotions based on the cause triggering them and development of situational personalized emotion model.

However, all the present researches do not take into account explicit emotions and analyze only a specific part of the psychological reason that leads to emotion in texts. They have not analyzed lack of keywords, disambiguate, lack of linguistic information and do not take the context into account. The corpus used in the research has been specifically developed such that they suit the main theme of the paper. None of the systems can produce satisfactory results for texts with disambiguates or lack of linguistic information.

METHODOLOGY



There are four different text based emotion recognition techniques: Keyword based detection method, Lexical Affinity Method, learning based method and hybrid methods. These methods are divided into subcategories as shown in above figure.

1. **Keyword based detection:** This method includes finding occurrences of keywords from a given set as substrings in a given string. It finds the presence of keywords and may involve pre-processing with a parser and emotion dictionary. This can be applied in real time chat systems. It is domain specific, relies on the presence of keywords for accurate results and requires pre-processing for improved accuracy results.

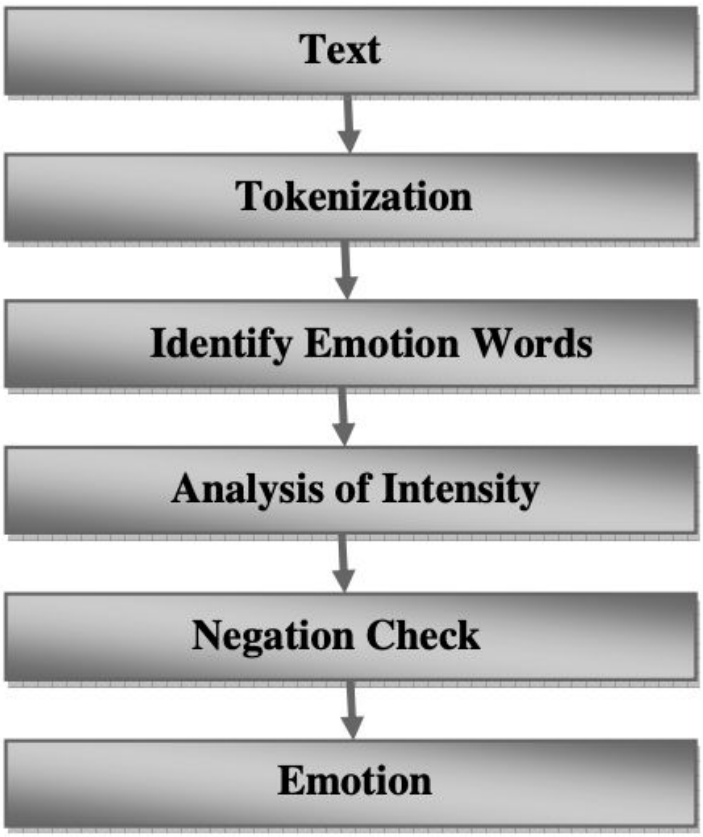


Fig: Keyword spotting technique

2. **Lexical Affinity Method:** Lexical Affinity method is an extension of keyword spotting method. This assigns a probabilistic affinity for a particular emotion to arbitrary words rather than detecting predefined emotional keywords from text.

3. **Learning based approach:** This approach uses a trained classifier to categorise input text into emotion classes by using keywords as features, which mapped with various machine learning classifiers such as support vector machines, k-means classifier etc.

4. **Hybrid Based Approach:** This approach consists of a combination of the keyword based implementation and learning based implementation. The main advantage of this approach is that it can yield higher accuracy results from training a combination of classifiers and adding knowledge-rich linguistic information from dictionaries. It will balance the high cost involved for information retrieval tasks and minimize difficulties.

DATASET

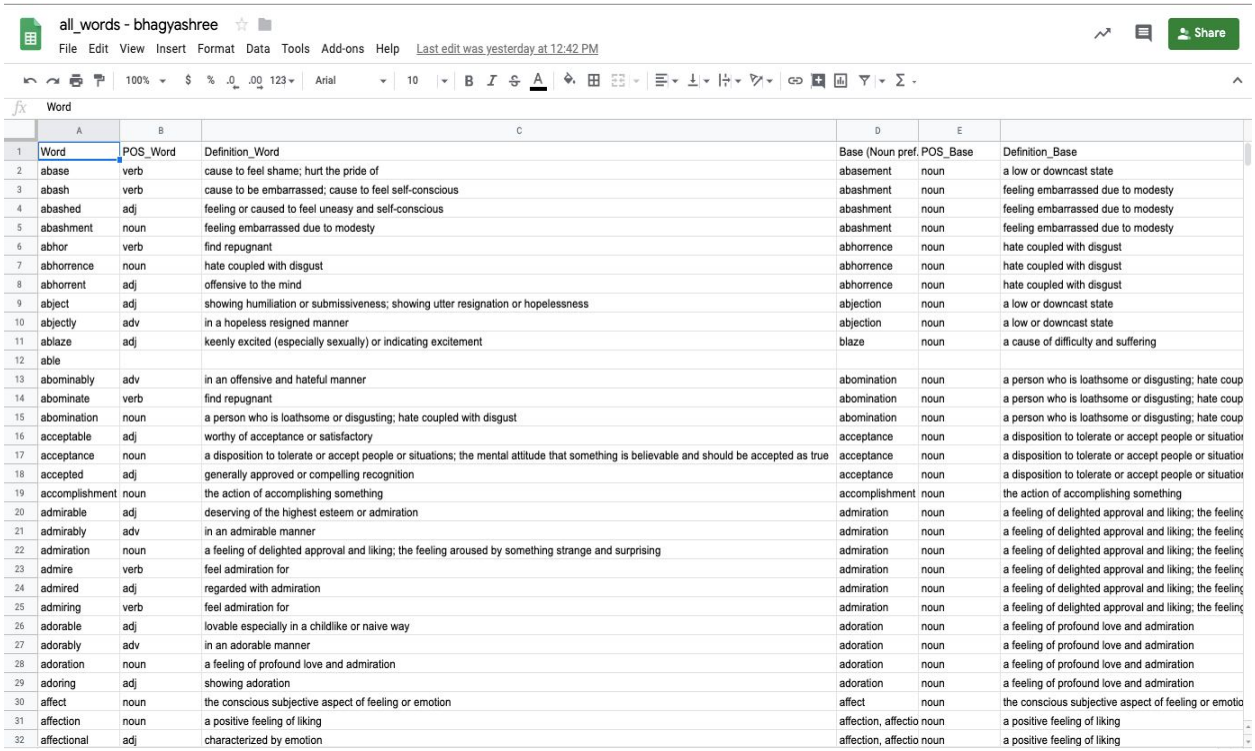
First and foremost work to proceed any task further is to have a proper dataset. Paragraphs of a book present a rich resource for publicly available text. They hold different styles of expression, and are in almost all existing languages. The empirical research I present in this paper harnesses the huge potential in chapters of book for text-based emotion detection.

Former experiences have shown that it is quite complicated when it comes to emotion extraction from text as compared to emotion detection using face, voice and gestures. Among the difficulties that hinder text- based emotion detection, there are: the complexity of natural language, its continuous evolution (new expressions everyday), and the ambiguous context of the author.

I will be classifying emotions according to the six basic emotions of Ekman. Each emotion category is represented by a list of expressive words. And to determine the emotion expressed in a piece of text, I first classify its component words. I start by extracting Adjectives, Nouns, Verbs and Adverbs (NAVA words) from text. The other words (pronouns, interjections, prepositions...) are not considered, because they are all the time neutral.

Later the data is clustered and Unsupervised or supervised learning(SVM) will be applied to detect the emotion of the overall given sentence.

Affective words are manually been pos tagged and defined.



all_words - bhagyashree

File Edit View Insert Format Data Tools Add-ons Help Last edit was yesterday at 12:42 PM

Share

Word

	A	B	C	D	E	
1	Word	POS_Word	Definition_Word	Base (Noun pref.	POS_Base	Definition_Base
2	abase	verb	cause to feel shame; hurt the pride of	abasement	noun	a low or downcast state
3	abash	verb	cause to be embarrassed; cause to feel self-conscious	abashment	noun	feeling embarrassed due to modesty
4	abashed	adj	feeling or caused to feel uneasy and self-conscious	abashment	noun	feeling embarrassed due to modesty
5	abashment	noun	feeling embarrassed due to modesty	abashment	noun	feeling embarrassed due to modesty
6	abhor	verb	find repugnant	abhorrence	noun	hate coupled with disgust
7	abhorrence	noun	hate coupled with disgust	abhorrence	noun	hate coupled with disgust
8	abhorrent	adj	offensive to the mind	abhorrence	noun	hate coupled with disgust
9	abject	adj	showing humiliation or submissiveness; showing utter resignation or hopelessness	abjection	noun	a low or downcast state
10	abjectly	adv	in a hopeless resigned manner	abjection	noun	a low or downcast state
11	ablaze	adj	keenly excited (especially sexually) or indicating excitement	blaze	noun	a cause of difficulty and suffering
12	able					
13	abominably	adv	in an offensive and hateful manner	abomination	noun	a person who is loathsome or disgusting; hate coup
14	abominate	verb	find repugnant	abomination	noun	a person who is loathsome or disgusting; hate coup
15	abomination	noun	a person who is loathsome or disgusting; hate coupled with disgust	abomination	noun	a person who is loathsome or disgusting; hate coup
16	acceptable	adj	worthy of acceptance or satisfactory	acceptance	noun	a disposition to tolerate or accept people or situatio
17	acceptance	noun	a disposition to tolerate or accept people or situations; the mental attitude that something is believable and should be accepted as true	acceptance	noun	a disposition to tolerate or accept people or situatio
18	accepted	adj	generally approved or compelling recognition	acceptance	noun	a disposition to tolerate or accept people or situatio
19	accomplishment	noun	the action of accomplishing something	accomplishment	noun	the action of accomplishing something
20	admirable	adj	deserving of the highest esteem or admiration	admiration	noun	a feeling of delighted approval and liking; the feelin
21	admirably	adv	in an admirable manner	admiration	noun	a feeling of delighted approval and liking; the feelin
22	admiration	noun	a feeling of delighted approval and liking; the feeling aroused by something strange and surprising	admiration	noun	a feeling of delighted approval and liking; the feelin
23	admire	verb	feel admiration for	admiration	noun	a feeling of delighted approval and liking; the feelin
24	admired	adj	regarded with admiration	admiration	noun	a feeling of delighted approval and liking; the feelin
25	admiring	verb	feel admiration for	admiration	noun	a feeling of delighted approval and liking; the feelin
26	adorable	adj	lovable especially in a childlike or naive way	adoration	noun	a feeling of profound love and admiration
27	adorably	adv	in an adorable manner	adoration	noun	a feeling of profound love and admiration
28	adoration	noun	a feeling of profound love and admiration	adoration	noun	a feeling of profound love and admiration
29	adoring	adj	showing adoration	adoration	noun	a feeling of profound love and admiration
30	affect	noun	the conscious subjective aspect of feeling or emotion	affect	noun	the conscious subjective aspect of feeling or emotio
31	affection	noun	a positive feeling of liking	affection, affectio	noun	a positive feeling of liking
32	affectional	adj	characterized by emotion	affection, affectio	noun	a positive feeling of liking

IMPLEMENTATION

Given any sentence, the first task is to consider individual words to analyze the emotional content. This is achieved by a process called tokenizing. i consider the tokens in my case to be separated by whitespace characters, such as a space or line break, or by punctuation characters.
For the same, I have used nltk(natural language toolkit) package for the removal of stop words and for the pos tagging.

Removal of Stop words: Stop words are basically a set of commonly used words in any language. These are few unnecessary words in the sentence, which needs to be removed. Removing the stop words from the sentence reduces memory overhead and also it can potentially improve power of prediction.

Pos Tagging: In a sentence**,** not all words contribute to the level emotion in a sentence. Certain parts of speech specifically contribute to the emotion level in a sentence. It tokenizes as well as indicate various parts of speech.

Removal of Stop words:

In [24]:
```
#Removing Stop words
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize

s1 = "I am trying to remove stop words from this sentence using stopword library of nltk."
stop_words = set(stopwords.words('english'))
word_tokens = word_tokenize(s1)

#filtered_sentence = [w for w in word_tokens if not w in stop_words]
filtered_sentence = []

for w in word_tokens:
    if w not in stop_words:
      filtered_sentence.append(w)

print(word_tokens)
print(filtered_sentence)
```

```
['I', 'am', 'trying', 'to', 'remove', 'stop', 'words', 'from', 'this', 'sentence', 'using', 'stopword', 'library', 'o
f', 'nltk', '.']
['I', 'trying', 'remove', 'stop', 'words', 'sentence', 'using', 'stopword', 'library', 'nltk', '.']
```

Pos Tagging:

```
In [28]: # Pos Tagging removing stop words
         from nltk.corpus import stopwords
         from nltk.tokenize import word_tokenize, sent_tokenize

         s1 = "I am trying to remove stop words from this sentence using stopword library of nltk."
         stop_words = set(stopwords.words('english'))
         word_tokens = sent_tokenize(s1)

         #filtered_sentence = [w for w in word_tokens if not w in stop_words]
         filtered_sentence = []

         for w in word_tokens:
           wordsList = nltk.word_tokenize(w)
           #for i in wordsList:
            # if i not in stop_words:
           wordsList = [w for w in wordsList if not w in stop_words]
           tagged = nltk.pos_tag(wordsList)

         print(tagged)

[('I', 'PRP'), ('trying', 'VBG'), ('remove', 'VB'), ('stop', 'JJ'), ('words', 'NNS'), ('sentence', 'NN'), ('using',
'VBG'), ('stopword', 'NN'), ('library', 'NN'), ('nltk', 'NN'), ('.', '.')]
```

Dataset which is used for the implementation is Twitter dataset which contains 40,000 tweets. Supervised machine learning (SVM and Naive Bayes) is used to recognize the emotion from the given text.

#Dataset

```
In [3]: 1 data = pd.read_csv('text_emotion.csv')
        2 data = data.iloc[:100,:]
        3
        4 data
```

Out[3]:

	tweet_id	sentiment	author	content
0	1956967341	empty	xoshayzers	@tiffanylue i know i was listenin to bad habi...
1	1956967666	sadness	wannamama	Layin n bed with a headache ughhhh...waitin o...
2	1956967696	sadness	coolfunky	Funeral ceremony...gloomy friday...
3	1956967789	enthusiasm	czareaquino	wants to hang out with friends SOON!
4	1956968416	neutral	xkilljoyx	@dannycastillo We want to trade with someone w...
5	1956968477	worry	xxxPEACHESxxx	Re-pinging @ghostridah14: why didn't you go to...
6	1956968487	sadness	ShansBee	I should be sleep, but im not! thinking about ...
7	1956968636	worry	mcsleazy	Hmmm. http://www.djhero.com/ is down
8	1956969035	sadness	nic0lepaula	@charviray Charlene my love. I miss you
9	1956969172	sadness	Ingenue_Em	@kelcouch I'm sorry at least it's Friday?
10	1956969456	neutral	feinyheiny	cant fall asleep
11	1956969531	worry	dudeitsmanda	Choked on her retainers
12	1956970047	sadness	Danied32	Ugh! I have to beat this stupid song to get to...
13	1956970424	sadness	Samm_xo	@BrodyJenner if u watch the hills in london u ...
14	1956970860	surprise	okiepeanut93	Got the news
15	1956971077	sadness	Sim_34	The storm is here and the electricity is gone
16	1956971170	love	poppygallico	@annarosekerr agreed
17	1956971206	sadness	brokenangel1982	So sleepy again and it's not even that late. I...

#Data Preprocessing:

Pre - processing includes:

- Removal of regular expressions, symbols using the 're' library
- Removal of lemmas (Lexicon Normalization) using WordNetLemmatizer from NLTK
- Removal of multi-letter ambiguities, e.g 'noooo' gets converted to 'no'
- Removal of stop-words - caused decrease in f1-score as well as overall accuracy

In [15]:

```
#comprehensive cleaning
def cleaning(text):
    txt = str(text)
    txt = re.sub(r"http\S+", "", txt)
    if len(txt) == 0:
        return 'no text'
    else:
        txt = txt.split()
        index = 0
        for j in range(len(txt)):
            if txt[j][0] == '@':
                index = j
        txt = np.delete(txt, index)
        if len(txt) == 0:
            return 'no text'
        else:
            words = txt[0]
            for k in range(len(txt)-1):
                words+= " " + txt[k+1]
            txt = words
            txt = re.sub(r'[^\w]', ' ', txt)
            if len(txt) == 0:
                return 'no text'
            else:
                txt = ''.join(''.join(s)[:2] for _, s in itertools.groupby(txt))
                txt = txt.replace("'", "")
                txt = nltk.tokenize.word_tokenize(txt)
                #data.content[i] = [w for w in data.content[i] if not w in stopset]
                for j in range(len(txt)):
                    txt[j] = lem.lemmatize(txt[j], "v")
                if len(txt) == 0:
                    return 'no text'
                else:
                    return txt
```

#Algorithm (SVM)

In [20]:

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics import classification_report
from sklearn import svm
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(data.content, data.sentiment, test_size=0.25, random_state=0)

x_train = x_train.reset_index(drop = True)
x_test = x_test.reset_index(drop = True)

y_train = y_train.reset_index(drop = True)
y_test = y_test.reset_index(drop = True)

vectorizer = TfidfVectorizer(min_df=3, max_df=0.9)

train_vectors = vectorizer.fit_transform(x_train)
test_vectors = vectorizer.transform(x_test)

model = svm.SVC(kernel='linear')
model.fit(train_vectors, y_train)
predicted_sentiment = model.predict(test_vectors)

print(classification_report(y_test, predicted_sentiment))
predicted_sentiments = []
for s in range(len(predicted_sentiment)):
    predicted_sentiments.append(predicted_sentiment[s])

prediction_df = pd.DataFrame({'Content':x_test, 'Emotion_predicted':predicted_sentiment, 'Emotion_actual': y_test})
prediction_df.to_csv('emotion_recognizer_svm.csv', index = False)

elapsed_time = time.time() - start_time
print ("processing time:", elapsed_time, "seconds")
```

#Result(SVM)

```
              precision    recall  f1-score   support

       anger       0.00      0.00      0.00         9
     boredom       0.00      0.00      0.00         7
       empty       0.00      0.00      0.00        53
  enthusiasm       0.00      0.00      0.00        29
         fun       0.00      0.00      0.00        52
   happiness       0.10      0.01      0.02       110
        hate       0.55      0.12      0.20       149
        love       0.30      0.07      0.11        88
     neutral       0.33      0.29      0.31       485
      relief       0.00      0.00      0.00        60
     sadness       0.29      0.31      0.30       536
    surprise       0.00      0.00      0.00       129
       worry       0.36      0.66      0.46       793

   micro avg       0.34      0.34      0.34      2500
   macro avg       0.15      0.11      0.11      2500
weighted avg       0.29      0.34      0.29      2500

processing time: 1785.6436619758606 seconds
```

#Accuracy(When implemented on 10000 tweets):

```
[ ]  from sklearn.metrics import accuracy_score, confusion_matrix

     cm = confusion_matrix(predicted_sentiment, y_test)
     print(cm)

     print(accuracy_score(y_test, predicted_sentiment))
```

```
[[  0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   1   1   0   1   0   1   3   1   2]
 [  0   0   0   0   0   0  18   0   2   0   8   1   4]
 [  0   1   0   0   1   1   0   6   2   1   5   1   2]
 [  3   2  13   6  14  23  19  13 139  12  57  25  96]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   2   8   6   9  24  42  33  70  12 165  22 168]
 [  0   0   0   0   0   0   0   0   0   0   0   0   1]
 [  6   2  32  17  27  61  70  35 272  34 298  79 520]]
0.3396
```

#TestData

Emotion is predicted in a given sentence using SVM.

```
In [28]: 1 prediction_df
```

Out[28]:

	Content	Emotion_predicted	Emotion_actual
0	people be so rude to you isaac they should get...	worry	sadness
1	totally screw up my ability to talk to a parti...	neutral	hate
2	n o t e x t	neutral	sadness
3	to take days off of work or have the money to ...	worry	worry
4	text	neutral	neutral
5	Got ta go to bed soon	neutral	neutral
6	n o t e x t	neutral	love
7	n o t e x t	neutral	neutral
8	that suck B be you ok	worry	neutral
9	my ipod i cant fall asleep	worry	sadness
10	i ve just be wrap up in day to day stuff so i ...	neutral	neutral
11	to summer strawberry be not availble in the Ch...	sadness	worry
12	ae get marry and it wasn t to alex	worry	worry
13	the hill in london u will realise what tourtur...	sadness	sadness
14	text	neutral	worry
15	when she start type on her computer in the mid...	worry	hate
16	talk at the Balisage Markup Conference 2009 Pr...	sadness	neutral
17	think i m definitely go to get an ear infectio...	worry	sadness
18	want to go out	sadness	neutral
19	I miss you	worry	sadness

#Algorithm(NaiveBayes)

In [35]:
```
data['content'] = data['content'].map(lambda x: cleaning(x))

data = data.reset_index(drop=True)
for i in range(len(data)):
    words = data.content[i][0]
    for j in range(len(data.content[i])-1):
        words+= ' ' + data.content[i][j+1]
    data.content[i] = words
```

```
/Users/bhagyashree/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
```

In [36]:
```
x = int(np.round(len(data)*0.75))
train = data.iloc[:x,:].reset_index(drop = True)
test = data.iloc[x:,:].reset_index(drop = True)
```

In [37]:
```
from textblob.classifiers import NaiveBayesClassifier as NBC
```

In [38]:
```
training_corpus = []

for k in range(len(train)):
    training_corpus.append((train.content[k], train.sentiment[k]))
test_corpus = []

for l in range(len(test)):
    test_corpus.append((test.content[l], test.sentiment[l]))

model = NBC(training_corpus)

```

#Accuracy(When implemented on 10000 tweets)

In [38]:
```
training_corpus = []

for k in range(len(train)):
    training_corpus.append((train.content[k], train.sentiment[k]))
test_corpus = []

for l in range(len(test)):
    test_corpus.append((test.content[l], test.sentiment[l]))

model = NBC(training_corpus)

print(model.accuracy(test_corpus))
```

```
0.314
```

Future scope of the project

Emotion detection is a growing research field. A lot of work has already been done on sentiment analysis, facial recognition and speech recognition. For now, I have implemented a supervised model to detect the emotion in a given sentence(tweet) which is classified word to word. As compared, SVM gives slightly better efficiency than Naive Bayes.

Further My project aims to recognize emotions in a sentence or paragraph. Determining overall impact of sentence can help in various fields especially in marketing. Consumers state of mind can be read by analysing what they write. More features(emotions) can be added to detect separately like recognizing sarcasm. Further it aims to show the relationship between Valence, arousal and Dominance.

References

1. Emotion detection from text by V V Ramalingam.
   https://iopscience.iop.org/article/10.1088/1742-6596/1000/1/012027/pdf

2. Computational Approaches for Emotion Detection in Text by Haji, Chen Wu
   http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.863.8593&rep=rep1&type=pdf

3. Text Based Emotion Recognition: A Survey by Chetan R. Chopade
   https://pdfs.semanticscholar.org/50cc/848978797112a27d124af57991321f545069.pdf

4. Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text
   https://ieeexplore.ieee.org/document/6797872

5. Recognition of Emotions in text with SVM by Zhi Teng , Fuji Ren, and Shingo Kuroiwa
   https://link.springer.com/chapter/10.1007/978-3-540-37275-2_87

6. Emotion detection from text by Shiv Naresh Shivhare and Prof. Saritha Khethawat
   https://airccj.org/CSCP/vol2/csit2237.pdf