



Emotion Analysis on Text Using Multiple Kernel Gaussian...

S. Angel Deborah¹ · T. T. Mirnalinee¹ · S. Milton Rajendram¹

Accepted: 25 January 2021 / Published online: 20 February 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC part of Springer Nature 2021

Abstract

The ability to discern human emotions is critical for making chatbox behave like humans. Gaussian Process (GP) is a non-parametric Bayesian modeling and can be used to predict the presence of either a single emotion (single-task GP) or multiple emotions (multi-task GP) in natural language text. Employing multiple kernels in GP can enhance the performance of the emotion analysis tasks. The particular choice of kernel functions determines the properties such as smoothness, length scales, sharpness, and amplitude, drawn from the GP prior. Using a specific kernel may be a source of bias and can be avoided by using different kernels together. The default kernel used with GP is a Radial Basis Function (RBF). It is infinitely differentiable; GP with this function has mean square derivatives of all orders and is thus very smooth. The sharpness which occurs in the midst of the smoothness can be detected using the exponential kernel. The multi-layer perceptron kernel has greater generalization for each training example and is good for extrapolation. Our experiments show that, for learning the presence of a single emotion in a natural language sentence (single-task), multiple kernel GP with the sum of RBF and multi-layer perceptron kernels performs better than single kernel GP. Likewise, for learning the presence of several different emotions in a sentence (multi-task), multiple kernel GP with the sum of RBF, exponential and multi-layer perceptron kernels performs better than single kernel GP. Multiple Kernel Gaussian Process also outperforms Convolutional Neural Network (CNN).

Keywords Natural Language Processing (NLP) · Gaussian Process (GP) · Emotion analysis · Single-task GP · Multi-task GP · Multiple kernel learning · RBF kernel · Exponential kernel · Multi-Layer perceptron kernel

1 Introduction

Emotion analysis is a computational study of how opinions, attitudes, emotions and perspectives are expressed in natural language, and provides techniques for extracting and summarizing useful information about them from natural language text. Automatic detection of emotions in texts finds applications in opinion mining, market analysis, affective comput-

✉ S. Angel Deborah
angeldeborahs@ssn.edu.in

¹ Department of CSE, Sri Sivasubramaniya Nadar College of Engineering, Chennai, India

ing, natural language interfaces and e-learning environments. Consequently, it has attracted increasingly greater attention in Natural Language Processing (NLP) research.

The different methods used by people provide window into their emotional realm. People use their facial expression, vocal intonation, body language, physiological response and written text to convey their emotions. In online communication, most often emotional information is encrypted in text. In the absence of non-verbal reminders, writers become accustomed to the medium by infusing messages with emotion reminders either explicitly or implicitly to allow for more normal communication. With the increase in occurrence of emotional contents on the Web, particularly on social media and microblogs, automatic emotion detection in text is gaining significant consideration from researchers and business people involved in exploring how emotions affect decision making, behaviors, and quality of life [1].

Automatic emotion detection in natural language processing requires techniques to find emotions expressed in written discourse. Designing computers that has ability to find the emotion expressed in text is an application in computational linguistics. Research in sentiment analysis provides a hopeful direction for fine-grained sentiment analysis of subjective content. In most research works, sentiment analysis research functions at a coarser level. Sentiment analysis is mostly aimed at recognizing the subjectivity or semantic position of a unit of text rather than a specific emotion [2]. Frequently, finding closely how a person reacts emotionally towards a specific provocation does matter. For illustration, while fear and sadness are both negative emotions, distinguishing between them can be crucial. In the occurrence of a disaster, fear may be used to detect an onset of the disaster whereas sadness may be linked with later stages.

In real business applications, automatic emotion detectors can offer good insights into how a particular audience feels about a product, person, event or topic. Business people are trying to obtain innovative methods for evaluating user-generated content to study about consumer's emotional responses toward their products, events and services. For example, automatic emotion detection systems used online product reviews to aid businesses for identifying and tracking emotional responses toward their products and services. Such kind of automatic anger detection systems in customer service emails can be used by customer service representatives to identify angry customers quicker, so that necessary immediate actions can be taken to increase customer retention rate. In market consumer analytics, automatic emotion detection systems offer businesses with non-invasive tactics to sell and advertise their offerings better to their customers.

Recognizing emotions is a major challenge for both humans and machines. On the one hand, people often express their own emotions vaguely [3]. On the other hand, machines need to have accurate ground truth for emotion modeling, and developing these emotion models require advanced machine learning algorithms.

We demonstrated the effectiveness of Multiple Kernel Gaussian Process (MKGP) in emotions present in sentences. We have used SemEval 2007 affective text dataset and SemEval 2017 fine-grained sentiment analysis dataset to study and analyse the performance of Single-task Single Kernel Gaussian Process (SSKGP), Multi-task Single Kernel Gaussian Process (MSKGP), Multi-task Multiple Kernel Gaussian Process (MMKGP) and Single-task Multiple Kernel Gaussian Process (SMKGP). Our team SSN_MLRG1 participated in SemEval 2017 task 5: fine-grained sentiment analysis on financial microblogs and developed a Multiple Kernel Gaussian Process (MKGP) model for a single-task problem. This paper extends MKGP model to a multi-task problem. The MMKGP requires not only requires multiple kernel, it also requires multitask GP. By using multi-task GP only single model can be used to predict six different emotions as given in SemEval 2007 affective text dataset.

The rest of this paper is organized as follows. Section 2 presents a brief review of work related to emotion analysis. Gaussian Process (GP), GP Regression and multi-task GP are defined in Sect. 3. Section 3.1 describes multiple kernel learning. Section 3.2 gives an overview of the system we developed. The dataset, kernels used, and evaluation results are discussed in Sects. 4 and 5 respectively. Section 7 is a conclusion and pointer to future directions.

2 Related Work

Existing automatic methods can be classified into five main categories; It includes lexicon-based methods, learning-based methods, manually constructed rules, knowledge-based methods and hybrid methods. Lexicon-based methods, considered to be easier to implement than other methods, use a lexicon to detect emotions in text. This method is based on the assumption that individual words bear emotional coloring [3], and that emotions articulated in text can be sufficiently represented at the word level. This method is the earliest approaches used for automatic emotion detection in text. It is also known as keyword spotting.

Learning-based approaches can be divided into two groups: supervised and unsupervised. Supervised learning approach uses marked up training data with pre-defined labels. Unsupervised learning approach uses similarity between data points to find if they can be characterized as belonging to a cluster. The facility for learning-based methods to take into account contextual information and to capture emotional cues in segments longer than a word makes it an appealing method to handle text with more nuanced emotional coloring.

Supervised machine learning approaches are more common than unsupervised approaches for automatic emotion detection in text. A human-annotated corpus is needed to first train and evaluate a machine learning model. An emotion corpus contains text segments that are manually annotated with a pre-defined set of emotion categories. With the help of corpus, the machine learning algorithm learns patterns associated with different emotion categories. The features like, bag of-words (BoW) and word n-grams are found to be popular features for emotion detection in text. BoW has been proven to be a successful feature set in sentiment analysis [4–6]. Rezaeinia [7] introduced Improved WordVectors (IWV) that increased the accuracy of pre-trained word embeddings in sentiment analysis. In binary classification, a text segment is classified as either being a positive or negative example of an emotion category. Determining if a text segment is emotional or non-emotional is an example of binary classification [8]. Joint emotion analysis on Semeval 2007 affective text, using multi-task GP based on coregionalisation, is introduced in [9]. Supervised statistical text classification approach leveraging a variety of semantic and sentiment features is used to perform sentiment analysis on short informal text by [10]. Chatterjee [11] proposed a novel deep learning based approach to detect emotions such as “happy”, “sad” and “angry” in textual dialogues. For sentences that contain more than one emotion, researchers have either included them in a separate category labeled as “mixed emotions” [12] or allowed multiple labels to be assigned to each sentence (i.e., multi-label classification problem) [8]. Sadr et al. and Zhang et al. [13–15] proposes multi-view deep network that uses intermediate features extracted from convolutional and recursive neural networks to perform classification. Based on the results of the experiments, the proposed multi-view deep network not only outperforms single-view deep neural networks but also has superior efficiency and generalization performance. In machine learning algorithms, Support Vector Machines (SVMs) are popular for this problem

space as they can scale to a large number of features and can do better than other classifiers for text classification [16]. Chaffar and Inkpen [17] showed that SVMs performed and generalized well on unseen data in emotion classification. They investigated the performance of classifiers using Naïve Bayes, decision trees and SVM on a diverse corpus annotated with six basic emotions and reported that SVM yielded the greatest accuracy improvement compared to the baseline.

Unsupervised learning methods have been used recently for emotion detection. Most of these methods are proposed to detect emotions that are expressed implicitly in text. One such famous unsupervised learning method in this problem domain is Latent Semantic Analysis (LSA). Strapparava and Mihalcea [18] assessed the semantic similarity among the terms in a given text and emotion concepts using a variation Latent Semantic Analysis (LSA), an unsupervised learning method. LSA and SVM were used for automatic detection of emotions in text by [19] which used a mix of unsupervised and supervised techniques to learn word vectors capturing semantic term document information as well as rich sentiment content. LSA allows vectors containing emotion words, its synonyms or synsets and document vectors containing generic terms to be mapped into a concept. Of the five approaches done by Strapparava and Mihalcea [18], the LSA approaches obtained in relatively higher recall and F-score than lexicon-based and supervised learning-based approaches but obtained the worst precision. Zhang [20] also used LSA to perform emotion processing of an intelligent agent in a role-playing virtual drama application. LSA has also been employed to detect emotion in Amazon customer reviews by Ahmad and Laroche [21]. Specific words depicting four emotions of interest, their associated synonyms and the consumer review were represented as vectors in concept space. The manually built rule-based method uses rules to decide if a text segment contains emotion or not. Initially, rules are created manually from an initial data set. Researchers have to scrutinize sample text to look for grammatical patterns connected with each emotion category or derive patterns based on a theoretical framework. These patterns are manually converted into a list of rules, which acts as the basis for a rule engine or inference engine. Rules need not be limited to lexical cues (e.g., keywords) in text, but can also deal with the more complex syntactic and semantic structures of a sentence. Syntactic and semantic information is obtained by examining texts through a parser. Along with the lexicon-based approach, Automatic emotion detection in text using manually constructed rules is also one of the early approaches. Many manually constructed rule-based approaches develop complex rules based on emotion lexicons to deal with the complexity of language. Zhe and Boucouvalas [22] constructed syntactic rules to include only emotion words expressed in first person form, took into account present continuous and perfect continuous tense as an indicator of emotion intensity, and excluded conditional sentences in an Internet chat environment. Donath et al. [23] (1999) set up rules to detect phrases in all capital letters, excessive punctuations, and profanities to find the angry present in a conversation. In processing news titles, Chaumartin [24] used syntactic rules to find the subject of the news title, as well as to find differences and accentuations between good news and bad news. Liu et al. [25] framed four rules to represent affective commonsense sentences from the Open Mind Commonsense Corpus. Most often, only a limited number of rules are defined to capture the obvious and non-ambiguous patterns. The generalizability of rules is also a cause for concern.

The ontology-based method is centered on the generation of a machine-readable formal representation of human emotions. Ontology is an “explicit specification of conceptualization” for a particular domain [26]. This structural representation includes a domain vocabulary, descriptions of concepts and attributes, as well as the relations between concepts. Unlike lexicons, ontologies do not operate on a word-level. Rather, they are defined

in terms of high-level concepts. Concepts are connected through taxonomic relations, and semantic relations. Motivation for researchers to implement this method mainly stemmed from the lack of agreement in how emotion is defined in the research community. Proponents of the ontology-based approach aim to define a standard set of descriptors that can help reduce the ambiguity in the interpretation of emotion expressed in text. Ontology-based methods are concerned with the creation, modification, and testing of emotion ontologies. The adoption of ontology-based methods for emotion detection in text is still fairly new, and has started to appear in the literature a few years ago. One of the earliest attempts to build an emotion ontology came from Grassi [27]. Grassi [27] defined only high-level emotion concepts and properties in the Human Emotions Ontology (HEO). The concepts, properties, and relations were derived from multiple emotion theories well-known in psychology. Although ontologies provide some form of consistency on the knowledge of emotion, extensive efforts are needed to build a consistent, if not a comprehensive one.

Hybrid method works by combining at least two of the four main methods used for emotion detection in text: lexicon-based, learning-based, manually constructed rules, and ontology-based. A hybrid approach aims to strategically control the strengths of diverse selected methods in an integrative framework. A combination of keyword spotting for emotion estimation of words and a set of rules for emotion estimation of sentences was used to build a textual emotion prediction system by Ma et al. [28] to chat with the animated agent. The usage of hybrid approaches is apparent in more recent research. In 2011, i2b2/VA/Cincinnati conducted Medical Natural Language Processing Challenge to assign emotions to suicide notes. Many proposed systems were designed using hybrid methods [29]. In 2012, Yang et al. designed a voting-based system to pick emotions for each sentence based on outputs from a mixture of keyword spotting, Conditional Random Field (CRF), and supervised machine learning methods. Nikfarjam et al. [30] first used rules to filter out sentences with obvious emotional cues and passed the uncertain cases to a supervised machine learning model for a final decision to solve the same problem. It was concluded by Sohn et al.

A hybrid sentiment analysis method for analysing Turkish dataset combines the lexicon-based and machine learning based approaches such as naive Bayes, support vector machines, and J48 [31]. [32] presents a hybrid neural network model called Convolutional Neural Network–Long Short-Term Memory (CNN-LSTM) with multivariate Gaussian model, to perform sentiment analysis on a microblog big-data platform and obtains significant improvement that enhances the generalization ability. Hybrid methods provide a good solution by combining the strengths of one approach to overcome the weaknesses of another approach. Thus these methods are creating more optimal and efficient automatic emotion detectors. Finding out which combination of approaches work optimally together remains a challenge for the research community.

In [33], sentiment analysis in non-structured free-text was carried out to predict the overall rating of product reviews, based on user opinions about the different product features that were evaluated in the reviews. [34] dealt with the problem of social affective text mining, aimed at discovering connections between social emotions and affective terms, based on user-generated emotion labels. The proposed model was a joint emotion-topic model, augmenting LDA with an additional layer for emotion modeling. Experiments about emotion analysis of news headlines were described in [18] which implemented five different systems for emotion analysis, using knowledge-based and corpus-based approaches. [35] and [35,36] performed opinion mining and sentiment analysis on movie review dataset [35], examining the effectiveness of machine learning techniques to sentiment classification problem, and the factors that make it challenging. [37] explored text-based emotion prediction problem empirically,

using supervised machine learning with the SNoW learning architecture. In this paper, we have used GP, a supervised Bayesian learning approach to predict the presence of single emotion or multiple emotions.

3 Gaussian Process

Gaussian Process is a non-parametric Bayesian modeling in supervised setting. Gaussian process is defined as *a collection of random variables, any finite number of which have a joint Gaussian distribution*. Using a Gaussian process, we can define a distribution over functions $f(x)$,

$$f(\mathbf{x}) \sim \text{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

where $m(\mathbf{x})$ is the mean function, usually defined to be zero, and $k(\mathbf{x}, \mathbf{x}')$ is the covariance function (or kernel function) that defines the prior properties of the functions considered for inference [38].

As highlighted in [9,39], Gaussian Process (GP) has the following main advantages: the kernel hyper-parameters can be learned via evidence maximization. GP provides full probabilistic prediction and an estimate of uncertainty in the prediction. Compared to SVMs which need unbiased datasets for good performance, GP does not usually suffer from biased datasets. GP can be easily extended and incorporated into a hierarchical Bayesian model. GP works really well when combined with kernel models. GP is effective even while learning from small datasets.

3.1 Gaussian Process Regression

Gaussian Process model, as they are applied in machine learning, is an attractive way of doing non-parametric Bayesian modeling for supervised learning problems. GP-based modeling has the ability to learn hyper-parameters directly from data by maximizing the marginal likelihood. Like other kernel methods, Gaussian Process can be optimized exactly, given the values of their hyper-parameters, and this often allows a fine and precise trade-off between fitting the data and smoothing.

The Gaussian Process Regression (GPR) framework assumes that, given an input x , output y is a noisy version of a latent function evaluation. As stated in [40], in a regression setting, we usually consider a Gaussian likelihood which allows us to obtain a closed form solution for the test posterior. In Algorithm 1, K is a covariance matrix computed using training inputs, I is identity matrix and α is $(K + \sigma_n^2 I)^{-1} y$. \bar{f}_* is Gaussian process posterior mean, \mathbf{k}_* is a vector, short form of $K(X, x_*)$, when there is only a single test case and $V[f_*]$ is the predictive variance. Line numbers 2 to 5 in Algorithm 1 address the matrix inversion required by Eqs. 1 and 2 using Cholesky factorization. A practical implementation of Gaussian Process Regression as discussed in [38] is outlined in Algorithm 1:

Algorithm 1: Prediction and log-marginal likelihood for GP regression

Algorithm: GP Regression

Input : X (training inputs), \mathbf{y} (training targets), k (covariance function), σ_n^2 (noise level), \mathbf{x}_* (test input)

Output: Predictive mean, variance and log-marginal likelihood

- 1 $L := \text{Cholesky}(K + \sigma_n^2 I)$
 - 2 $\alpha := L^T \backslash (L \backslash \mathbf{y})$
 - 3 $\bar{f}_* := \mathbf{k}_*^T \alpha$
 - 4 $\mathbf{v} := L \backslash \mathbf{k}_*$
 - 5 $V[f_*] := k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{v}^T \mathbf{v}$
 - 6 $\log p(\mathbf{y}|X) := -\frac{1}{2} \mathbf{y}^T \alpha - \sum_i \log L_{ii} - \frac{n}{2} \log 2\pi$
 - 7 **return** f_* (mean), $V[f_*]$ (variance), $\log p(\mathbf{y}|X)$ (log-marginal likelihood)
-

In Algorithm 1, K is a covariance matrix computed using training inputs, I is identity matrix and α is $(K + \sigma_n^2 I)^{-1} \mathbf{y}$. \bar{f}_* is Gaussian process posterior mean, \mathbf{k}_* is a vector, short form of $K(X, x_*)$, when there is only a single test case and $V[f_*]$ is the predictive variance. Line numbers 2 to 5 in Algorithm 1 address the matrix inversion required by Eqs. 1 and 2 using Cholesky factorization.

$$\bar{f}_* := \mathbf{k}_*^T (K + \sigma_n^2 I)^{-1} \mathbf{y} \tag{1}$$

$$V[f_*] := k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T (K + \sigma_n^2 I)^{-1} \mathbf{k}_* \tag{2}$$

For multiple test cases lines 3–5 are repeated. The log determinant required in Eq. 3 is computed from the Cholesky factor.

$$\log p(\mathbf{y}|X) := -\frac{1}{2} \mathbf{y}^T (K + \sigma_n^2 I)^{-1} \mathbf{y} - \frac{1}{2} \log |K + \sigma_n^2 I| - \frac{n}{2} \log 2\pi \tag{3}$$

The algorithm uses Cholesky decomposition, instead of directly inverting the matrix, since it is faster and numerically more stable. The algorithm returns the predictive mean and variance for noise-free test data. To compute the predictive distribution for noisy test data \mathbf{y}_* , we have to add the noise variance σ_n^2 to the predictive variance of f_* . GP can also be used for classification problem [41].

3.2 Multi-task Gaussian Process

Multi-Task Learning (MTL) is motivated by human learning activities where persons often apply the knowledge learned from previous tasks to help learn a new task. MTL is a learning approach in machine learning that aims to leverage useful information contained in multiple related tasks to help improve the generalization performance of all the tasks [42]. Given m learning tasks $\{T_i\}_{i=1}^m$ where all the tasks or a subset of them are related, multi-task learning aims at improving the learning of a model for T_i by using the knowledge contained in all or some of the m tasks. [39] presented multi-task learning models by representing intra-task transfer simply and explicitly as a part of a parameterised kernel function [39]. According to [43], GP is an extremely flexible probabilistic framework and has been successfully adapted for multi-task learning, by modeling multiple correlated output variables. It develops the early work from geostatistics on learning latent continuous spatio-temporal models from sparse point measurements, a problem setting that has clear parallels to transfer learning (including domain adaptation).

Multi-task learning can be done using a separable vector-valued kernel known as Intrinsic Coregionalisation Model (ICM). The ICM is a low rank approach matrix combined with a vector-valued GP. There are three important reasons to employ this model: first, datasets for the task are scarce and small so it is hypothesized that a multi-task approach will result in better models by allowing a task to borrow statistical strength from other tasks. Second, the annotation scheme is subjective and very fine-grained, and is therefore heavily prone to bias and noise, both of which can be modeled easily using GPs. Finally, the goal is to learn a model that shows sound and interpretable correlations between emotions.

Multiple output kernels or vector-valued kernels use the coregionalisation matrix B which is positive definite, and a kernel function K . The coregionalized model shares information across outputs, which the independent models cannot do. In the regions where there is no training data specific to an output, the independent models tend to return to the prior assumptions. In the case of coregionalized model, where both outputs have associated patterns, the fit is better.

Considering a set of D tasks, the corresponding vector-valued kernel is defined as

$$k((\mathbf{x}, d), (\mathbf{x}', d')) = k_{\text{data}}(\mathbf{x}, \mathbf{x}') \times \mathbf{B}_{(d,d')}$$

where k_{data} is a kernel on the input points, d and d' are tasks or metadata information for each input, and $\mathbf{B} \in \mathbb{R}^{D \times D}$ is the coregionalisation matrix which encodes task covariances and is symmetric and positive semi-definite.

The approach in [39] treats the diagonal values of \mathbf{B} as hyperparameters and, as a consequence, is able to leverage the inter-task transfer between each independent task and the global pooled task. It, however, fixed non-diagonal values to 1, which in practice is equivalent to assuming equal correlation across tasks. This can be limiting in that this formulation cannot model anti-correlations between tasks. This restriction is lifted by adopting a different parameterisation of \mathbf{B} that allows the learning of all task correlations. A straightforward way to do this would be to consider every correlation as a hyperparameter, but this can result in a matrix which is not positive semi-definite. To ensure this property, [9] followed the method proposed by [44], which decomposes \mathbf{B} using Probabilistic Principal Component Analysis:

$$\mathbf{B} = \mathbf{U}\mathbf{A}\mathbf{U}^T + \text{diag}(\alpha)$$

where \mathbf{U} is an $D \times R$ matrix containing the R principal eigenvectors and \mathbf{A} is a $R \times R$ diagonal matrix containing the corresponding eigenvalues. The choice of R defines the rank of $\mathbf{U}\mathbf{A}\mathbf{U}^T$, which can be understood as the capacity of the manifold with which we model the D tasks. The vector α allows each task to behave more or less independently with respect to the global task. For numerical stability, [9] used the incomplete-Cholesky decomposition over the matrix $\mathbf{U}\mathbf{A}\mathbf{U}^T$, resulting in the following parameterisation for \mathbf{B} .

$$\mathbf{B} = \tilde{\mathbf{L}}\tilde{\mathbf{L}}^T + \text{diag}(\alpha)$$

where $\tilde{\mathbf{L}}$ is a $D \times R$ matrix. In this setting, all elements of $\tilde{\mathbf{L}}$ are treated as hyperparameters.

4 Multiple Kernel Learning

The heart of every Gaussian process model is a covariance kernel. The kernel \mathbf{k} directly specifies the covariance between every pair of input points in the dataset. The particular choice of covariance function determines the properties such as smoothness, length scales and amplitude, drawn from the GP prior. Therefore, it is an important part of GP modeling

to select an appropriate covariance function for a particular problem. Multi Kernel Learning (MKL) – using multiple kernels instead of a single one—can be useful in two ways:

- Different kernels correspond to different notions of similarity, and instead of trying to find which works best, a learning method does the picking for us, or may use a combination of them. Using a specific kernel may be a source of bias which is avoided by allowing the learner to choose from among a set of kernels.
- Different kernels may use inputs coming from different representations, possibly from different sources or modalities.

It is reported that multiple kernels definitely give a powerful performance [45,46]. Various methodologies to combine kernels is described in detail in [45]. [46] introduced simple closed form kernels that can be used with Gaussian Processes to discover patterns and enable extrapolation. The kernels support a broad class of stationary covariances, but Gaussian Process inference remains simple and analytic.

We studied the possibility of using multiple kernels to explain the relation between the input data and the labels. While there is a body of work on using Multiple Kernel Learning (MKL) on numerical data and images, yet applying MKL on text is still in the early stages of research. We have combined kernels from among Exponential kernel, Multi-Layer Perceptron kernel and Squared Exponential kernel, and found the combinations to perform better. The text data used in sentiment analysis is collected over a period of time. Comments on the same topic may exhibit different emotions, depending on the time it was made, and hence their properties, such as smoothness and periodicity, also vary with time. Since any one kernel learns only certain properties well, multiple kernels are effective in detecting the simultaneous presence of different properties in the data.

The MKL algorithms use different learning approaches for determining the kernel combination function. There are five major approaches: Fixed rules, Heuristic, Optimization, Bayesian, and Boosting. These different learning approaches may combine kernels in a linear or a non-linear way. Linear combination seems more promising, and have two basic categories: unweighted sum (i.e., using sum or mean of the kernels as the combined kernel) and weighted sum. Non-linear combination uses non-linear functions of kernels such as multiplication, power, and exponentiation. We have studied the fixed rules linear combination in this work, which can be represented as

$$\mathbf{k}(x, x') = \mathbf{k}_1(x, x') + \mathbf{k}_2(x, x') + \dots + \mathbf{k}_n(x, x').$$

The various kernels that were used to build the GP models are described below.

Squared Exponential Kernel

The squared exponential (SE) kernel, also known as radial basis function (RBF) or exponentiated quadratic kernel, has become the default kernel in GPs. To model the long term smooth rising trend, we use a squared exponential (SE) covariance term.

$$\mathbf{k}(x, x') = \sigma^2 \exp\left(-\frac{(x - x')^2}{2l^2}\right)$$

Length-scale l shows the extent to which the function is smooth. Small length-scale value means that function values can change quickly; large values characterize functions that change only slowly. Length-scale also determines how far we can reliably extrapolate from the training data. Signal variance σ^2 is a scaling factor which determines variation of function

values from their mean. Small value of σ^2 characterize functions that stay close to their mean value, while larger values allow more variation. If the signal variance is too large, the modelled function will be free to chase outliers.

Exponential Kernel

Use of exponential kernel is common in machine learning, and hence finds use in GPs also. It performs tasks such as statistical classification, regression analysis, and cluster analysis on data in an implicit space. The exponential kernel is closely related to the squared exponential kernel, with only the square of the norm left out. It can be used for identifying the sharpness in the data.

$$\mathbf{k}(x, x') = \sigma^2 \exp\left(-\frac{(x - x')}{2l^2}\right)$$

Multi-layer Perceptron Kernel

The multi-layer perceptron kernel has also been used in GP as it has greater generalization for each training example and is good for extrapolation. It is given by

$$\mathbf{k}(x, x') = \frac{2\sigma^2}{\pi} \sin^{-1} \frac{(\sigma_w^2 x^T x' + \sigma_b^2)}{\sqrt{\sigma_w^2 x^T x + \sigma_b^2 + 1} \sqrt{\sigma_w^2 x'^T x' + \sigma_b^2 + 1}}$$

where σ^2 is the variance, σ_w^2 is the vector of the variances of the prior over input weights and σ_b^2 is the variance of the prior over bias parameters. The kernel can learn more effectively because of the additional parameters σ_w^2 and σ_b^2 .

5 System Overview

We developed a system consisting of the following modules: data extraction, preprocessing, feature vector generation, model selection, hyperparameter optimization and GP model building. Figure 1 shows the architecture of the system. The input dataset is extracted, then XML tagged News headline and Emotion score are separated. The XML tagged News headline is tokenized and lemmatized. Following it, the lemmatized words are used to build a data dictionary. All the words in the data dictionary are mapped to their indices. For each sentence, feature vectors are generated. The feature vectors are BoW representation of each sentence. To build a GP model, key-value pair is generated with feature vectors and emotion scores. The key is the emotion and the value is a matrix where rows are BoW vectors.

To build a Single-task Single Kernel Gaussian Process (SSKGP) model, dataset with BoW feature representation is taken as input and an initial regression model is built using Square Exponential kernel function. Then the hyperparameters of the model such as length-scale, variance and noise are optimized by maximizing the likelihood. The hyperparameters were finally learnt and the model is tested. The hyperparameters are optimized using Limited-Memory Broyden-Fletcher-Goldfarb-Shanno method (L-BFGS). The L-BFGS optimization procedure is a limited memory variation of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) quasi-Newton algorithm. Rather than storing the Hessian, the L-BFGS method stores only the gradient vectors for the last few geometries calculated.

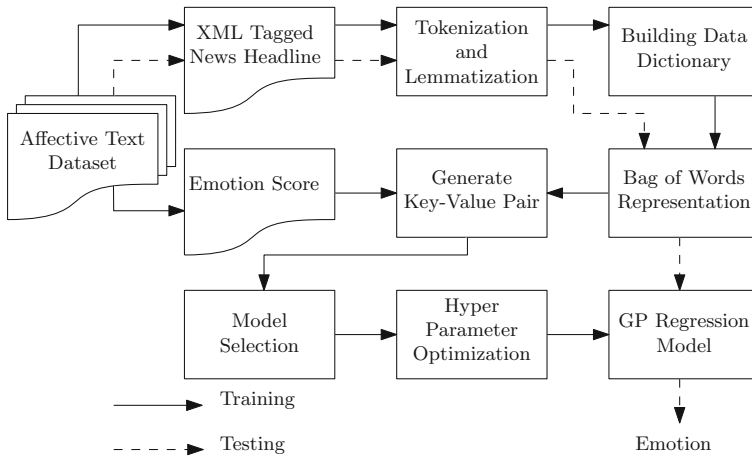


Fig. 1 System architecture

A Single-task Multiple Kernel Gaussian Process (SMKGP) initial regression model is built with BoW feature vectors as input and a linear combination of different kernel functions. Similar to SSKGP, the hyperparameters of the model such as length-scale, variance and noise are optimized by maximizing the likelihood. In single task approach, six different separate models are built to predict six emotions, increasing the time complexity and space complexity. This can be overcome by using multi task learning approach. To build a Multi-task Single Kernel Gaussian Process (MSKGP) model, dataset with BoW feature representation is taken as input and an initial regression model is built using Intrinsic Coregionalisation Matrix and Square Exponential kernel function. Following this, the hyper-parameters such as length-scale, variance, noise and the kappa values for the 6 emotions are optimized by maximizing the likelihood. A Multi-task Multiple Kernel Gaussian Process (MMKGP) initial regression model is built with BoW feature vectors as input, Intrinsic Coregionalisation Matrix and a linear combination of different kernel function.

6 Dataset

SemEval 2007 affective text dataset [47] and SemEval 2017 fine-grained sentiment analysis dataset [48,49] were used to study the performance of SMKGP and MMKGP. The SemEval 2007 dataset includes the text data instances annotated with emotions. The text data instances are new headlines that have been extracted from major newspapers such as New York Times, Google news, CNN. News Headlines generally consist of a few words and are often formulated by creative people to provoke the emotions of the readers, and thereby attract larger group of readers. Such kind of news headlines are most suitable for developing an emotion detection system. And also such kind of short sentences are guaranteed to have affective features that can be very useful in developing a better emotion recognition system. For every instance, the emotion score ranging from 0 to 100 is given for all the six emotions (Anger, Disgust, Fear, Joy, Sadness and Surprise).

SemEval 2017 fine-grained sentiment analysis dataset consists of two separate datasets. One of the datasets comprises comments from different financial microblogs and the other comprises comments from news headlines. Both the datasets include the details of the text

data instances and, for each instance, real number emotion scores ranging from -1 to 1 . The score near -1 is referred to as optimistic and the score near $+1$ as pessimistic. Financial microblog messages consists of StockTwits messages focusing on stock market events and assessments from investors and traders, exchanged via the StockTwits microblogging platform. Typical stocktwits consist of references to company stock symbols, a short supporting text or references to a link or pictures and twitter messages that discuss about some stock market. In order to extend and diversify the data sources, Twitter posts containing company stock symbols were extracted. News Statements and Headlines are sentences taken from news headlines as well as news text. Textual content has been crawled from different sources on the internet, such as Yahoo Finance.

7 Results and Discussion

The SemEval 2007 dataset can be evaluated using either a set of single-task models such as single-task SVM and single-task GP, one multi-task model such as multi-task GP or CNN. The drawback of using single-task SVM and single-task GP is that six different GP models have to be learned to predict the six different emotions, whereas multi-task GP can predict the six different emotions with just one model.

We compared prediction results obtained using multi-task multiple kernel GP with a set of single-task/single-kernel baselines: a Support Vector Machine (SVM) using an RBF kernel, a single-task GP optimised via likelihood maximisation, and a multi-task single-kernel GP. The SVM models were trained using the Scikit-learn toolkit. The results of the various GP models applied to SemEval 2007 dataset are shown in Tables 1 and 2. Table 1 shows the performance

Table 1 A performance comparison based on Pearson Score and Mean Absolute Error (100 instances for training and 900 instances for testing)

Model	SVM	SSKGP	MSKGP	MMKGP (R+E)	MMKGP (R+M)	MMKGP (R+E+M)	CNN
Pearson score							
Anger	0.3084	0.1683	0.2165	0.3437	0.3659	0.3564	0.0975
Disgust	0.2135	0.0035	0.2137	0.3077	0.3218	0.3290	0.1500
Fear	0.3525	0.3462	0.351	0.3592	0.3944	0.3865	0.1078
Joy	0.0905	0.2035	0.2435	0.2261	0.2441	0.2424	-0.1122
Sadness	0.3330	0.3011	0.2933	0.2660	0.3413	0.3264	-0.0678
Surprise	0.1148	0.1599	0.2195	0.2463	0.2571	0.2593	0.0939
All	0.2603	0.3659	0.3815	0.3936	0.4041	0.4055	0.0449
Mean absolute error							
Anger	11.2673	11.489	11.3629	10.6641	10.0342	10.0896	10.8362
Disgust	6.9012	7.094	7.6932	6.9867	7.0444	6.7895	4.9331
Fear	14.4533	14.3436	14.081	13.9399	13.3103	13.2792	16.5453
Joy	17.4323	17.8929	17.7896	17.4143	16.9809	17.4728	17.1354
Sadness	15.9828	16.105	16.156	15.9301	15.6442	15.6139	16.6204
Surprise	10.2902	10.8895	11.4996	10.2274	10.7921	10.2749	10.8680
All	12.5212	12.969	13.0971	12.5271	12.3010	12.1971	12.8231

Bold values are the best values

Table 2 A performance comparison based on Pearson Score and Mean Absolute Error (700 instances for training and 300 instances for testing)

Model	SVM	SSKGP	MSKGP	MMKGP (R+E)	MMKGP (R+M)	MMKGP (R+E+M)	CNN
Pearson score							
Anger	0.4449	-0.0315	0.4816	0.4845	0.4672	0.4329	0.0941
Disgust	0.3691	0.4664	0.5096	0.5119	0.4904	0.4699	0.0584
Joy	0.3002	0.3370	0.4174	0.4136	0.4124	0.4604	-0.0324
Sadness	0.426	0.4521	0.5234	0.5304	0.5270	0.5534	0.0199
Surprise	0.2671	0.2545	0.2485	0.2666	0.2530	0.3479	0.0757
All	0.4971	0.4829	0.5674	0.5709	0.5686	0.5748	0.0540
Mean absolute error							
Anger	5.5282	5.4067	7.4491	6.9467	7.4232	6.9829	8.7598
Disgust	4.2504	5.5907	5.3486	4.9284	5.3208	5.0979	5.7125
Fear	11.4661	13.3080	11.1882	10.9437	11.4337	11.0982	16.5453
Joy	16.6626	16.4699	15.1224	15.1785	15.2778	14.8542	16.9040
Sadness	14.5549	16.1393	14.6757	14.2856	14.7448	13.7383	17.6441
Surprise	9.5506	9.3469	9.7510	9.7379	9.6559	9.5543	9.5188
All	10.3355	11.0436	10.5892	10.3368	10.5627	10.2210	12.4240

Bold values are the best values

comparison taking 100 instances for training and 900 instances for testing, as considered in [9]. Table 2 shows the performance comparison taking 700 instances for training and 300 instances for testing.

We expect multi-task models (MSKGP and MMKGP) to perform better for smaller datasets, when compared to single-task models and CNN model. With small datasets, often there is more uncertainty associated with each task, a problem which can be alleviated using statistics from the other tasks. To measure this behaviour, we performed an additional experiment varying the size of the training sets. Figure 2 shows the Pearson scores obtained. As expected, for smaller datasets, the single-task models are outperformed by multi-task models (MSKGP and MMKGP with ICM), but their performance comes closer as the training set size increases. SVM performance tends to be slightly worse for most sizes. It is observed that MMKGP performs slightly better than MSKGP. The kernel combinations used in Tables 1 and 2 are as follows.

SVM(R):	Support Vector Machine with Radial Basis Function (RBF) kernel,
SSKGP(R):	Single-task Single Kernel Gaussian Process with radial basis function (RBF) kernel,
MSKGP(R):	Multi-task Single Kernel Gaussian Process with radial basis function (RBF) kernel,
MMKGP(R+E):	Multi-task Multiple Kernel Gaussian Process with sum of RBF and exponential kernels,
MMKGP(R+M):	Multi-task Multiple Kernel Gaussian Process with sum of RBF and multi-layer perceptron kernels,
MMKGP(R+E+M):	Multi-task Multiple Kernel Gaussian Process with sum of RBF, exponential, and multi-layer perceptron kernels.
CNN:	Convolutional Neural Network

Table 3 A performance comparison based on Pearson Score (PS) and Mean Absolute Error (MAE) for SemEval 2017 financial microblogs and news headline dataset

Model	PS	MAE
SemEval 2017 financial microblogs dataset		
SSKGP	0.6694	0.2003
SMKGP(R+E)	0.6809	0.1965
SMKGP(R+E+M)	0.6864	0.1931
SMKGP(R+M)	0.6872	0.1930
CNN	0.0765	0.3294
SemEval 2017 news headline dataset		
SSKGP	0.5615	0.2506
SMKGP(R+E)	0.5569	0.2558
SMKGP(R+E+M)	0.5870	0.2445
SMKGP(R+M)	0.5928	0.2426
CNN	0.0235	0.3521

Bold values are the best values

The performance was evaluated based on Pearson Score (PS) r and Mean Absolute Error (MAE), which are calculated using Eqs. 4 and 5.

$$r = \frac{\sum_{i=1}^n (Y_i - \bar{Y})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \tag{4}$$

$$MAE = \frac{\sum_{i=1}^n |Y_i - y_i|}{n} \tag{5}$$

where Y is the actual output, y the predicted output, and n number of records. The greater the Pearson Score (PS) and the smaller the Mean Absolute Error (MAE), the better the performance of the system. It is found from Table 1 that MMKGP(R+E+M) model has better PS for Disgust and Surprise whereas MMKGP(R+M) has better PS for Anger, Fear, Joy, Sadness. However, MMKGP(R+M) has better PS on the whole. Likewise, Table 1 shows that MMKGP(R+E) has better MAE for Surprise, MMKGP(R+M) has better MAE for Anger and Joy, yet MMKGP(R+E+M) has better MAE for Disgust, Fear and Sadness. Considering all six emotions together, MMKGP(R+E+M) has lesser MAE when compared to both MMKGP(R+E) and MMKGP(R+M). Table 2 shows that MMKGP(R+E) has better PS for Disgust and Fear whereas MMKGP(R+E+M) has better PS for Anger, Joy and Surprise. With respect to MAE, Table 2 brings to light that, although SSKGP has lower MAE for Anger and Surprise, the MMKGP(R+E+M) has lower MAE when considering all emotions together. Overall, the results in Tables 1 and 2 reveal that the MMKGP models perform better compared to SSKGP, MSKGP models and CNN. Tables 1 and 2 also shows that SVM and SSKGP has performed much better than deep learning model, CNN.

The results of Single-task Single Kernel GP and Single-task Multiple Kernel GP on SemEval 2017 financial microblogs dataset and news headline dataset are shown in Table 3. Since SemEval 2017 dataset has only one emotion to be predicted, we used it to evaluate Single-task GP. 70% of the dataset was taken for training and 30% for testing. The tables show that SMKGP(R+M), Multiple Kernel Gaussian Process with sum of squared exponential and multi-layer perceptron kernels, performs better. The kernel combinations used in Table 3 are

SSKGP(R): Single-task Single Kernel Gaussian Process with radial basis function (RBF) kernel,

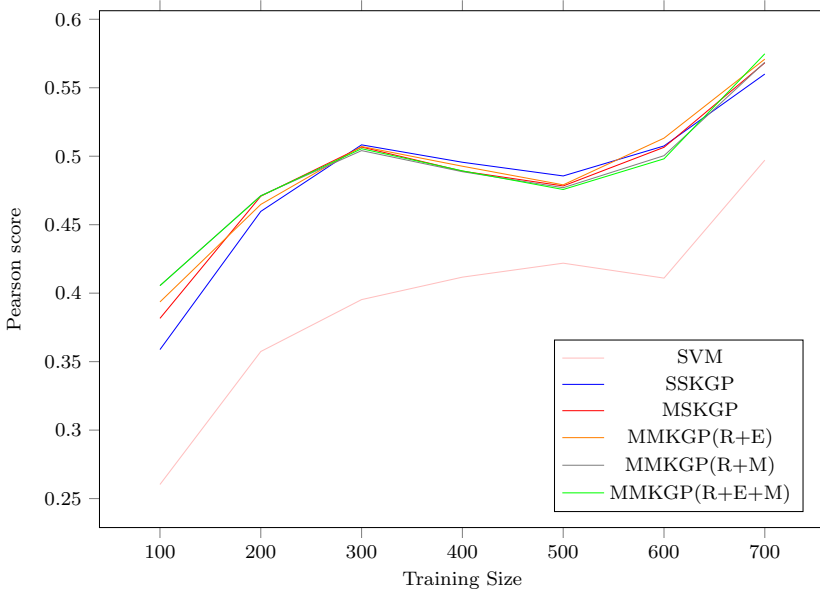


Fig. 2 Pearson score for varying training set size

- SMKGP(R+E): Single-task Multiple Kernel Gaussian Process with sum of RBF and exponential kernels,
 SMKGP(R+M): Single-task Multiple Kernel Gaussian Process with sum of RBF and multi-layer perceptron kernels.
 SMKGP(R+E+M): Single-task Multiple Kernel Gaussian Process with sum of RBF, exponential and multi-layer perceptron kernels,
 CNN: Convolutional Neural Network

The evaluations on SemEval 2007 and SemEval 2017 dataset show that both Single-task GP and Multi-task GP perform better with multiple kernels than with a single kernel. The RBF kernel, exponential kernel and multi-layer perceptron kernel, when used in different linear combinations, are capable of learning different properties like smoothness, periodicity, etc.

8 Conclusion

In this paper, we have presented a Multi Kernel Gaussian Process (MKGP) regression model for emotion analysis on news headlines and fine-grained sentiment analysis of financial microblogs and news. We used Bag of Words input feature vectors as input, ICM model for multi-task learning, and fixed rule multiple kernel learning to build GP model.

The experiments on SemEval 2017 dataset show that Multiple Kernel GP, by learning the different properties present in the text, has improved performance over Single Kernel learners and CNN. Likewise, from the results on SemEval 2007 dataset, we have found Multi-task Multiple Kernel GP performs better than a collection of Single-task learners and CNN (as well as Multi-task Single Kernel GP). It is possible to further enhance the results by using different feature generation approaches and more effective multi kernel learning approaches.

The approach proposed gives better result for the small sized dataset. The methodology can be further studied and experimented by incorporating deep learning into GP. Also the time

required for prediction using GP can be further enhanced using some optimization approaches in future.

Acknowledgements SemEval 2017 dataset - Horizon 2020 ICT Programme Project SSIX: Social Sentiment analysis financial Indexes, ICT-2014–2015. A Big OpenData, Grant Agreement No.: 645425 for Innovation action (2015–2018).

References

1. Zhao Y, Qin B, Liu T, Tang D (2016) Social sentiment sensor: a visualization system for topic detection and topic sentiment analysis on microblog. *Multimedia Tools Appl Springer* 75(15):8843–60
2. Li S, Huang L, Wang R, Zhou G (2015) Sentence-level emotion classification with label and context dependence. *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing*, vol 1, pp 1045–1053
3. Pajupuu H, Kerge K, Altrov R (2012) Lexicon-based detection of emotion in different types of texts: preliminary remarks. *Eesti Rakenduslingvistika Ühingu aastaraamat* 8:171–84
4. Pang B, Lee L, Vaithyanathan S (2002) Thumbs up? Sentiment classification using machine learning techniques. *arXiv preprint arXiv:cs/0205070*. May 28
5. Salvetti F, Reichenbach C, Lewis S (2006) Opinion polarity identification of movie reviews. In: *Computing attitude and affect in text: theory and applications*. Springer, Dordrecht, pp 303–316
6. Lewis DD (1992) An evaluation of phrasal and clustered representations on a text categorization task. In: *Proceedings of the 15th annual international ACM SIGIR conference on research and development in information retrieval*, June 1, pp 37–50
7. Rezaeinia SM, Rahmani R, Ghodsi A, Veisi H (2019) Sentiment analysis based on improved pre-trained word embeddings. *Expert Syst Appl* 117:139–47
8. Alm CO, Roth D, Sprout R (2005) Emotions from text: machine learning for text-based emotion prediction. In: *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, pp 579–586
9. Beck D, Cohn T, Specia L (2014) Joint emotion analysis via multi-task Gaussian processes. *Proceedings of the 2014 conference on empirical methods in natural language processing*. *ACL*, pp 1798–1803
10. Kiritchenko S, Zhu X, Mohammad SM (2014) Sentiment analysis of short informal texts. *J Artif Intell Res* 20(50):723–62
11. Chatterjee A, Gupta U, Chinnakotla MK, Srikanth R, Galley M, Agrawal P (2019) Understanding emotions in text using deep learning and big data. *Comput Hum Behav* 93:309–17
12. Aman S, Szpakowicz S (2008) Using roget's thesaurus for fine-grained emotion recognition. In: *Proceedings of the third international joint conference on natural language processing: volume-I*
13. Sadr H, Pedram MM, Teshnehlab M (2020) Multi-view deep network: a deep model based on learning features from heterogeneous neural networks for sentiment analysis. *IEEE Access* 4(8):86984–97
14. Sadr H, Pedram MM, Teshnehlab M (2019) A robust sentiment analysis method based on sequential combination of convolutional and recursive neural networks. *Neural Process Lett* 50(3):2745–61
15. Zhang S, Xu X, Pang Y, Han J (2020) Multi-layer attention based CNN for target-dependent sentiment classification. *Neural Process Lett* 51(3):2089–103
16. Yang H, Willis A, De Roeck A, Nuseibeh B (2012) A hybrid model for automatic emotion recognition in suicide notes. *Biomedical informatics insights*. 5:BII-S8948
17. Chaffar S, Inkpen D (2011) Using a heterogeneous dataset for emotion analysis in text. In: *Canadian conference on artificial intelligence*, May 25, Springer, Berlin, Heidelberg, pp 62–67
18. Strapparava C, Mihalcea R (2008) Learning to identify emotions in text. *Proceedings of the 2008 ACM symposium on applied computing*. *ACM*, pp 1556–1560
19. Maas AL, Daly RE, Pham PT, Huang D, Ng AY, Potts C (2011) Learning word vectors for sentiment analysis. *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*. *ACL*, pp 142–150
20. Zhang L (2013) Contextual and active learning-based affect-sensing from virtual drama improvisation. *ACM Trans Speech Lang Process (TSLP)* 9(4):1–25
21. Ahmad SN, Laroche M (2015) How do expressed emotions affect the helpfulness of a product review? Evidence from reviews using latent semantic analysis. *Int J Electron Commerce* 20(1):76–111
22. Zhe X, Boucouvalas AC (2002) Text-to-emotion engine for real time internet communication. In: *Proceedings of international symposium on communication systems, networks and DSPs*, July 15, pp 164–168

23. Donath J, Karahalios K, Viegas F (1999) Visualizing conversation. *J Comput-Mediated Commun* 4(4):JCMC442
24. Chaumartin FR UPAR7: a knowledge-based system for headline sentiment tagging
25. Liu H, Lieberman H, Selker T (2003) A model of textual affect sensing using real-world knowledge. In: *Proceedings of the 8th international conference on intelligent user interfaces*, January 12, pp 125–132
26. Gruber TR (1995) Toward principles for the design of ontologies used for knowledge sharing? *Int J Hum Comput Stud* 43(5–6):907–28
27. Grassi M (2009) Developing HEO human emotions ontology. In: *European workshop on biometrics and identity management*, September 16. Springer, Berlin, Heidelberg, pp 244–251
28. Ma C, Prendinger H, Ishizuka M (2005) Emotion estimation and reasoning based on affective textual interaction. In: *International conference on affective computing and intelligent interaction*, October 22. Springer, Berlin, Heidelberg, pp 622–628
29. Pestian JP, Matykiewicz P, Linn-Gust M, South B, Uzuner O, Wiebe J, Cohen KB, Hurdle J, Brew C (2012) Sentiment analysis of suicide notes: a shared task. *Biomed Inform Insights* 5:BII-S9042
30. Nikfarjam A, Emadzadeh E, Gonzalez G (2012) A hybrid system for emotion extraction from suicide notes. *Biomed Inform Insights* 5:BII-S8981
31. Erşahin B, Aktas Ö KD, Erşahin M (2019) A hybrid sentiment analysis method for Turkish. *Turk J Electr Eng Comput Sci* 27(3):1780–93
32. Sun X, Zhang C, Ding S, Quan C (2018) Detecting anomalous emotion through big data from social networks based on a deep learning method. *Multimedia Tools Appl* 1–22
33. De Albornoz JC, Plaza L, Gervás P, Díaz A (2011) A joint model of feature mining and sentiment analysis for product review rating. In: *European conference on information retrieval*, Springer, Berlin, Heidelberg, pp 55–66
34. Bao S, Xu S, Zhang L, Yan R, Su Z, Han D, Yu Y (2009) Joint emotion-topic modeling for social affective text mining. In: *2009 Ninth IEEE international conference on data mining*, IEEE, pp 699–704
35. Pang B, Lee L (2008) Opinion mining and sentiment analysis. *Found Trends Inf Retrieval* 2(1–2):1–35
36. Pang B, Lee L, Vaithyanathan S (2002) Thumbs up? Sentiment classification using machine learning techniques. *Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, Association for Computational Linguistics, vol 10, pp 79–86
37. Alm CO, Roth D, Sproat R (2005) Emotions from text: Machine learning for text-based emotion prediction. *Proceedings of the conference on human language technology and empirical methods in natural language processing*, ACL, 579–586
38. Williams CK, Rasmussen CE (2006) *Gaussian processes for machine learning*. MIT press, Cambridge
39. Cohn T, Specia L (2013) Modelling annotator bias with multi-task gaussian processes: an application to machine translation quality estimation. *Proceedings of the 51st annual meeting of the ACL*, vol 1, pp 32–42
40. Ebden M (2015) *Gaussian processes: a quick introduction*. arXiv preprint [arXiv:1505.02965](https://arxiv.org/abs/1505.02965), May 12
41. Angel Deborah S, Milton Rajendram S, Mirnalinee TT (2017) SSN_MLRG1 at SemEval-2017 task 4: sentiment analysis in twitter using multi-kernel Gaussian process classifier. *Proceedings of the 11th international workshop on semantic evaluation*. ACL, pp 709–712
42. Zhang Y, Yang Q (2017) A survey on multi-task learning. arXiv preprint [arXiv:1707.08114](https://arxiv.org/abs/1707.08114)
43. Alvarez Mauricio A, Rosasco L, Lawrence Neil D et al (2015) Kernels for vector-valued functions: a review. *Found Trends Mach Learn Now Publishers, Inc* 4(3):10177–10194
44. Alvarez MA, Rosasco L, Lawrence ND (2012) Kernels for vector-valued functions: a review. *Found Trends Mach Learn* 4(3):195–266
45. Gönen M, Alpaydın E (2011) Multiple kernel learning algorithms. *J Mach Learn Res* 12:2211–68
46. Wilson A, Adams R (2013) Gaussian process kernels for pattern discovery and extrapolation. *International conference on machine learning*, pp 1067–1075
47. Carlo S, Rada M (2007) Semeval-2007 task 14: affective text. *Proceedings of the 4th international workshop on semantic evaluation*, Association for Computational Linguistics, pp 70–74
48. Cortis K, Freitas A, Daudert T, Huerlimann M, Zarrouk M, Handschuh S, Davis B (2017) SemEval-2017 task 5: fine-grained sentiment analysis on financial microblogs and news. *11th International workshop on semantic evaluations (SemEval-2017)*. ACL, pp 519–535
49. Angel Deborah S, Milton Rajendram S, Mirnalinee TT (2017) SSN_MLRG1 at SemEval-2017 task 5: fine-grained sentiment analysis using multiple kernel Gaussian process regression model. *Proceedings of the 11th international workshop on semantic evaluation*. ACL, pp 823–826