Journal of Theoretical and Applied Information Technology
31st October 2019. Vol.97. No 20

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

THE USE OF PARTICLE SWARM OPTIMIZATION TO OBTAIN N-GRAM OPTIMUM VALUE FOR MOVIE GENRE CLASSIFICATION

EDI NOERSASONGKO[1], D. SALVANA ERVAN[1], H. AGUS SANTOSO[1], CATUR SUPRIYATO[1], FARIKH AL ZAMI[1], M.A. SOELEMAN[1]

[1] *Faculty of Computer Science, Dian Nuswantoro, Semarang, Indonesia*

E-mail: [1]edi-nur@dosen.dinus.ac.id, [1]deny.ervan@gmail.com, [1]heru.agus.santoso@dsn.dinus.ac.id,[1]catur.dinus@gmail.com, [1]alzami@dsn.dinus.ac.id,[1]arief22208@gmail.com

ABSTRACT

Document Classification that until now still be done by experts / human in the field related to the document. In recent years researchers included in this study proposed a variety of methods to solve the problem of document classification automatically. In this research, we used classification of movie genre based on synopsis as research object. Previous research proposed a model with a statistical approach with the Naive Bayes algorithm that proved to get the best results compared to other classification algorithms. Several studies have proposed adding a selection of N-gram features to pre-processing. The resulting classification becomes better than before. However, there is a weakness of N-gram between is the value of n which is determined still randomly or by trial and error. With these weaknesses, in this study proposes to optimize the N-gram model to obtain optimum n values using the Particle Swarm Optimization algorithm. Acquisition of an optimal N-gram n value will improve document performance and classification results with Naive Bayes. Based on the proposed model can then be used as a document classification model with different objects

Keywords: *Document Classification, N-gram, Particle Swarm Optimization, Naive Bayes*

1. INTRODUCTION

Nowadays, text classification and categorization is applied in many fields, such as: spam-email filtering, news categorization, information retrieval, summarization, movies genre classification and so on [1]. According to Sebastiani [2], Text classification and categorization is the method assigning natural language text into one or more categories based on their content. The text classification process have five sequentially steps: 1) document pre-processing; 2) document modelling; 3) feature selection and projection; 4) machine learning algorithm to obtain a learning model; 5) and evaluation methods[3].

Regarding the text classification process, in text classification and categorization, the research is mainly divided into two parts, one focused on text representation schemes and the latter focused on classification or categorization technique. In text classification and categorization, the text representation schemes usually utilized vector space model (VSM) which used for word-weighting. *Retrieval* results obtained from VSM are the most relevant documented *query* documents. In the VSM, documents and *queries* are represented as vectors in the vector space arranged in the *term* index, then modelled by geometric equations. Then, in form of classification or categorization technique, it can divided by three approach: *machine learning* approach, *lexicon-based* approach, and approach techniques that apply both or *hybrid approach* [4].

In the document pre-processing, the broader representation VSM is using keywords or phrases, which commonly known as unigram, bigram, trigram, n-gram [5][6][7]. In the literatures, many researchers predefined the gram for document pre-processing, some used trial and error for choosing the right gram. The right gram could affect in improvement of text classification performance. Thus, pre-processing will be more optimal in generating data or document classification if it succeeds in obtaining the n value of the optimum *N-gram*.

In this paper, we present an approach obtaining optimum N-gram by utilizing particle swarm optimization (PSO) which help the classifier to gain best performance in classification accuracy. In here we used naïve Bayes algorithm as base classifier due to naïve Bayes is one of classifier which usually used for text mining, then movies genre classification is used as base problem. A main

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

contribution in this paper is in matter of text representation schemes, PSO could help reduce the time and cost to obtain the optimum N-gram rather than trial and error.

The rest of paper is organized as follows: section 2 introduces related work; section 3 present the theoretical basis; section 4 present methodology; experiment and discussion are reported in section 5, then in section 6 presents about conclusion and future works.

2. RELATED WORK

Text classification have many research categories, such as: domain category, classification purpose category and classification task category. The detailed is described as follows.

1) in domain category, many researcher put research in domain of industry: Ittoo *et.al* put a discussions about challenges in industrial applications such as heterogeneous data source, artifacts, de-facto standard and quality of results [8]; Kumar and Ravi discussed various application of text mining in finance, such as: forex rate prediction, stock market prediction, customer relationship management and cyber security where most of those relies on news reports and social media or twitter sentiments [9]; Chaix *et.al* proposed text mining tools which successfully extract information in food microbiology in a large collection of PubMed abstracts of scientific papers especially for the phenotypes and habitat of microorganisms [10]; Basto-Fernandez *et.al* proposed multi-objective evolutionary algorithm to solve problem of three-way classification (which helps in mitigating security risks and information loss) and the postprocessing of results on SPAM filters [11]; and so on;

2) in classification purpose category, Giatsogloua *et.al* proposed approach where textual documents are used for polarity classification model by represented it as vector representations as word embeddings based, lexicon based and hybrid vectorizations in English and Greek language [12]; Kang *et.al* applied an ensemble text-based hidden markov model to new sentiments analysis method which used sequence of words training text for implicit opinions classification [13]; Catal and Nangir investigates Turkish sentiment problems using ensemble stacking classifier and giving note that parameter optimization of each classifier should be addressed carefully[14]; Garcia-Pablos *et.al* proposed almost unsupervised system based on topic modelling performs aspect-term, aspect category classification and opinion-word separation and sentiment polarity classification for any given domain and language [15];

3) in classification tasks, many research is based on binary class[16], multi class[17], multi label[18][19] and hierarchical[20];

One of the existing document classification methods is with statistical approaches with Naive Bayes and Bayesian Nets [21][22]. Among the existing classification algorithms, according to a survey of previously conducted researches, the probabilistic approach of Naive Bayes machine learning algorithms obtains optimal accuracy for document classification cases, derived from Naive Bayes dependencies.

In matter of text representation, the *N-gram* and *Bayesian Classification* studies were conducted by *Ross Clement* and *David Sharp*[23] who used the dataset of the *Internet Movie Database* (IMDb). The study aims to identify the *authors* of the *movie review* of movies taken as samples in several topics. By identifying the *term or frequency* value of each author sample they write each time a review will be found the pattern then movie *review* on the movie document is classified according to its *authorship*. The results obtained in the study are the *authorship* of each *movie review* posted in each IMDb movie document.

Subsequent research into the classification of movie genres was performed [24] by experimented with comparing several classifier algorithms that processed the movie's synopsis data to get the movie classification by genre automatically. The dataset used in that report is public data from IMDb. Several classifier models were compared: Support Vector Machine (SVM) conducted two experiments with no weighting and by increasing the weight of the penalty, K-Nearest Network (KKN) conducted 2 experimental values k, the Parametric Mixture Model (PMM) experiments with Maximum Likelihood Estimate (MLE) and Maximum a Posteriori (MAP) and Neural Network (NN) computations performed experiments with and without Principal Component Analysis (PCA).

Most common methods in selecting number of n-grams for vector space model is used trial and error, for this case is using cross-validation along with distance measurements[25]. If the document is big enough, the process selecting the right grams using brute search could take long time[26][27].

In recent years until now the Particle Swarm Optimization (PSO) has contributed a lot to researchers who are researching on parameter modification fields, or in order to optimize globally because these algorithms can adapt dynamically. The original (PSO) framework was designed by

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Kennedy and Eberhart [28] which is used for continuous nonlinear optimization of functions with evolutionary techniques. This is found through simulating the social behavior of bird flocks. This herd consists of several particles of a certain volume and velocity, each of which is a viable solution in the solution space. The optimal solution obtained from this algorithm is to observe the particle movements in the solution space. Due to the convenience of realization and promising optimization capabilities in various issues, PSO has been contributing in many researches related field related to optimization. Most studies focus on improving performance to modify parameters, optimize variable diversity, neural network training, and others [29].

Thus, from related work that previously described, the application of the PSO in this study is used to evaluate the resulting n values on the N-gram model with the learning or validation function formed from the classification model using Naive Bayes, which implicitly could be used to obtain best n-grams rather than brute search method in matter of speed of time.

3. THEORITICAL BASIS

3.1 Naïve Bayes

The Naive Bayes model works with conditional probabilities derived from a statistical approach known as "Bayes Theorem", in which as naive refers to the "assumption" that all attributes of the examples are independent of each other given context categories. Because the assumption of independence of parameters for each attribute can be studied separately and this greatly simplifies learning especially when the number of attributes is large [30]. In this context text classification, the probability that a document c belongs to a class a is computed by Bayes theorem as follows:

$$P(a|c) = \frac{P(c|a)P(a)}{P(c)} \quad (1)$$

Given a training set with class labels and E as a test case with n values of attributes (a_1, a_2, …, a_n). So, the Naive Bayes equation to classify E is:

$$c(E) = arg_{c\epsilon C} max\ P(c)P(a_1, a_2, \ldots, a_n|c) \quad (2)$$

The central assumption of Naive Bayes classification is that, in each class, attribute values are all independent of each other. Independent probability law:

$$P(a_1, a_2, \ldots, a_n|C) = \prod_{i=1}^{n} P(a_i|C) \quad (3)$$

By replacing (3) in (2), the resulting classification equation as shown below:

$$C_{NB}(E) = \arg_{c\epsilon C} \max P(C) \prod_{i=1}^{n} (a_i|C) \quad (4)$$

Where C_{NB} denotes the classification produced by Naive Bayes in test case E.
From equation (4), each probability can be determined directly from the training data. At the time of training Naive Bayes produced a one-dimensional table of estimated probability classes, indexed by classes, and the two-dimensional attribute-value conditional probability estimates table, indexed by classes and value-attributes.

Although independence is generally an unfavorable assumption, in practice Naive Bayes is often used as a comparison of algorithms with more complex techniques but the performance of Naive Bayes often yields surprising accuracy [22]. This has also been demonstrated by researchers on a large scale comparing the Naive Bayes algorithm with state-of-the-art decision-tree decision induction, instance-based learning, and rule induction [31].
Compared to other classifiers, Naive Bayes requires relatively little data for training. It trains very quickly, requires little storage space during the training process and classification, is easy to implement, and lacks many parameters like Neural Networks (NN) and Support Vector Machines (SVM) [32][33].

3.2 N-Gram

N-grams is defined as a sub-sequence of n characters of a given word. Namely, in the N-gram approach character, consecutive letters consecutively from a selected word, starting with the first letter. Then, the second consecutive letter n successively from the selected word begins with the second letter. This process continues until the selected N-gram is the last n of the given word [34].
The comparison process for query per *query* varies according to the number of words in the *consecutive query*. To determine the consecutive query equation, we define the threshold value. The N-gram character method makes a decision by comparing this value with the similarity ratio, which is defined as the identical ratio of *N-gram* compared to the total

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

number of *N-grams*. The ratio of similarities can be calculated as follows [34]:

$$similarity\ ratio = \frac{\delta}{\min(\alpha, \beta)} \quad (5)$$

Where:

δ = number of identical $n-gram$
α = number of n − gram for $word_A$
β = number of n − gram for $word_B$

where $word_A$ is the first word and $word_B$ is the second word used for comparison of *N-gram* characters.

3.3 PARTICLE SWARM OPTIMZATION

Particle swarm optimization have many advantages, such as: only have several parameter to adjust which is more convenient and simpler than other optimization algorithm , have higher efficiency and higher chance in finding global optima, did not overlap and mutate, fast in convergence and cheap in computational time.

Suppose the search space is *D*-dimensional, and the position of i *swarm* particles can be represented by a *D*-dimensional vector, $\boldsymbol{x}_i=(x_{i1}, \ldots, x_{id}, \ldots, x_{iD})$. The velocity (change of the position) of the particle x_i can be represented in the other D-dimensional vectors $\boldsymbol{v}_i=(v_{i1}, \ldots, v_{id}, \ldots, v_{iD})$. The best positions previously visited by the ith particles are denoted as $\boldsymbol{p}_i=(p_{i1}, \ldots, p_{id}, \ldots, p_{iD})$. If the topology is defined in such a way that all particles are assumed neighbours and g as the index of the particle to visit the best position in the *swarm*, then $\boldsymbol{p}_g$ becomes the best solution found so far, and the particle velocity and the new position will be determined according to the following two equations:

$$V_{id} = rho_{id}V_{id} + c_1r_1(p_{id} - x_{id}) + c_2r_2(p_{gd} + x_{id}) \quad (6)$$

$$x_{id} = x_{id} + v_{id} \quad (7)$$

where c_1 and c_2 are the acceleration coefficients set the best relative to global and local velocity, r_1 and r_2 are two random numbers in [0, 1]. In addition, the maximum allowable velocity $\boldsymbol{V}_{max}$ velocity clamps the particle velocity in each dimension. If acceleration causes speed at dimensions exceeding $\boldsymbol{V}_{max}$ to be determined by the user, then the speed at that dimension will be limited to $\boldsymbol{V}_{max}$.. In later studies, to ensure convergence, an algorithmic analysis of the mathematical aspects was given by [35], which proposed the use of χ a constriction factor; *Constriction Factor Method (CFM)* algorithm. $\varphi = c_1 + c_2$, when:

$$X = \begin{cases} \dfrac{2k}{\varphi - 2 + \sqrt{\varphi^2 - 4\varphi}}, & for\ \varphi > 4 \\ k, & otherwise \end{cases} \quad (8)$$

Where k is positive constant, then the Eq.6 becomes

$$V_{id} = X(v_{id} + c_1r_1(p_{id} - x_{id}) + c_2r_2(p_{gd} - x_{id})) \quad (9)$$

3.4 CONFUSION MATRIX

Confusion matrix is a concept of *machine learning* that contains information about the actual classification and prediction made by the classification system [36]. A *Confusion Matrix* has two dimensions, one is the dimension that is indexed by the actual / actual class of an object, the other is indexed by the *classifier* class. Figure 1. shows the basic form of the *Confusion Matrix* for the *multiclass* classification model with classes A_1, A_2. In *Confusion Matrix* N_{ij} represents the actual number of data samples belonging to class A_1 but belongs to the class A_j.

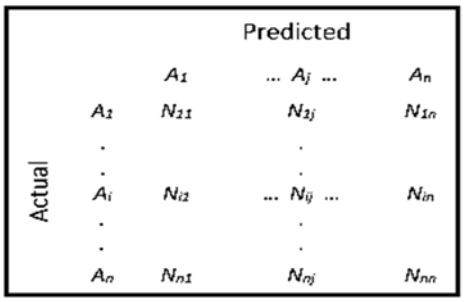

Figure 1: Confusion Matrix

Classification performance measurement can be defined by *Confusion Matrix*. Some of these steps include[37]:

Accuracy is the correct proportion of the number of predictions:

$$accuracy = \frac{\sum_{i=1}^{n} N_i}{\sum_{i=1}^{n}\sum_{j=1}^{n} N_{ij}} \quad (10)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Precision is a measure of the accuracy divided according to predetermined prediction class:

$$precision_i = \frac{N_{ii}}{\sum_{k=1}^{n} N_{ki}} \quad (11)$$

Recall is a measure of the ability of a prediction model to select an instance of a particular class of a set of data, defined by the formula:

$$recall_i = \frac{N_{ii}}{\sum_{k=1}^{n} N_{ik}} \quad (12)$$

4. METHODOLOGY

To improve the performance of Naive Bayes algorithm for document classification can be done in preprocessing data, one of them is by implementing N-gram model. In the process, N-gram will parse data which in case of classification of documents in this research is text, into several syllables based on the number of parser coefficients. N-gram model will be able to increase fault tolerance of writing on test data to be classified. So that the acquisition probability with the essence of the same word meaning is still detected in the probability calculation between the data with the term label used as reference classification document.

In previous studies, the N-gram n value was determined by trying several possibilities to obtain the best classification results. This is then the weakness of the N-gram model is the determination of n or parser coefficients that are still determined at random or by trial and error method. The determination of the optimum will have an impact on the performance and accuracy of the classification of documents produced by Naive Bayes.

PSO has the advantage to solve the problem that has nonlinear characteristics and not too high complexity. So, in the case of this research the researchers implement to determine the value of an optimum N-gram. The input given to the PSO is the position, value, and velocity of the particles which are initialized, the position and the value of the particles can be determined randomly with the initial zero velocity. The particle value is the result of the performance of the Naive Bayes classification model, which in the PSO implementation of the classification model is called the fitness function. The value of each particle is then used as a reference benchmark to determine the best particle position as a particle velocity determinant and subsequent particle position.

In essence the PSO process is to find the best position of the random positional colonies of particle positions based on the movement of the particles. These factors make the PSO process is iterative to get the particle position marked with the colony of the particles to form or be in an almost identical position. Convergence of the final position data of the particles is the best value or position obtained by the PSO.

The result of convergence of data or value is the value of N-gram optimum generated by PSO and Naive Bayes as learning model of classification. The optimum n value generated by the PSO will improve document classification performance, and time cost due to the more efficient determination of N-grams.

Proposed method: obtaining optimum n-gram	
Input: training document, testing document	
Output: GBest position where position is n-gram number	
Fitness function: accuracy result using n-gram Naïve Bayes	
initialize: number particle; particle position; particle velocity; max iter; PBest=0; GBest=0;	
Steps:	
1	Until achived max iteration
2	n = GBest position
3	Initialize: number particle; particle position; particle velocity; max iter; PBest=0; GBest=0;
4	For iter
5	For each particle
6	Calculate particle velocity
7	Calculate particle's new position
8	Assign new PBest with fitness
9	If new GBest > old GBest, then GBest = new GBest
10	Else GBest = old GBest
11	End For
12	If particle position is convergence
13	Break; get GBest; n = GBest position
14	Else continue to iter for calculate velocity and position
15	End For

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

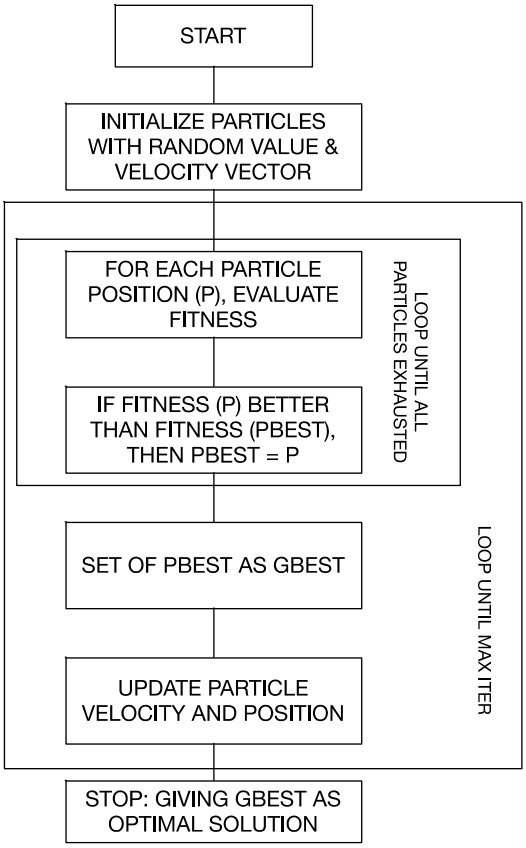

Figure 2: Basic Flow PSO

our movie classification is using text mining step as follows: 1) text preprocessing which cleaning the regular expression and special characters; 2) data transformation and weighting; 3) obtaining optimum n-gram using PSO with naïve Bayes as base classifier.

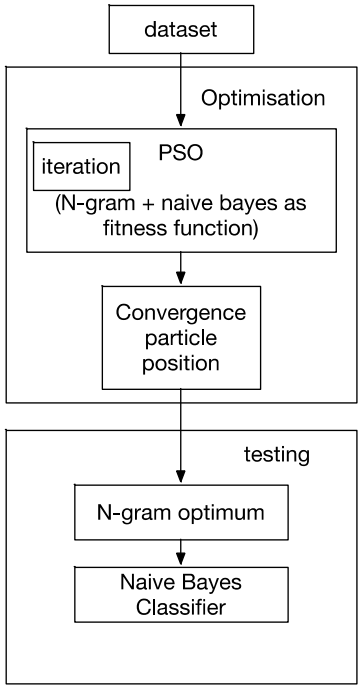

Figure 3: Proposed method

4.1 DATASET

The dataset used in this study is *public* data in the form of text obtained from *CMU Movie Summary Corpus* [38]. The movie data consists of various languages and countries, but all the plot synopsis is in English. The data *record* of 42,306 *movie plot summaries* will be divided into two data in the study, as training data and test data. The dataset consists of attributes: *Wikipedia movie id, freebee movie id, movie name, movie release date, movie box office revenue, movie runtime, movie languages, movie countries, movie genres.* The process of data extraction is the selection of attributes according to the needs of this research that includes attributes: movie title, synopsis or movie summary, and movie genres

4.2 N-GRAM OPTIMIZATION

This research proposes selection of *N-gram* model features with optimized value of n. The optimum value of n is obtained by implementing *Particle Swarm Optimization* (PSO) algorithm and classification process with *Naive Bayes* algorithm to obtain better classification model. Pre-processing using an *N-gram* model in this experimental study will compare the *N-gram* implementation with and without the optimization of the value of n used as the coefficient of *parser* value.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The optimum value n is obtained by applying the PSO algorithm. Before *generating* the value of n processed using PSO, a classification experiment was conducted using *Naive Bayes N-gram* to obtain a data classification model. Then the classification model will be used as *fitness function* in the optimization process. PSO algorithm will calculate each particle position based on its speed movement and validated with *fitness* function. The process is done iteratively and compares each position formed to obtain the best particle position according to PSO that is convergent, as the *parser* coefficient or the optimum value of *N-gram*

4.3 Classification

The document classification algorithm which in this case study is a movie genre based on the synopsis using *Naive Bayes*, as many researchers in this *field* implement the algorithm [39][40][41]. Models that have been obtained from the *training* stage will be tested at the *testing* stage with new data to determine the performance of the training results.

At the stage before data classification, *N-gram* optimization model has been done with the output of optimum n value. The coefficient of the *parser* will then be implemented and tested. The optimum *N-gram* parser coefficient is used to *parse* the word *term label* and movie synopsis which in the next process by *Naive Bayes* can be determined the probability of the tendency of the classification label

5. EXPERIMENTS

The data used in this study comes from CMU Movie Summary Corpus [38] which contains attributes of movie title, synopsis, genre, release year, and country of origin. But in this study only required the attributes of synopsis, movie genre, and movie title as his id. Researchers used the sample as a training data of 50 movies each and their synopsis of the five genres to be experimented at the next stage. Genres used as classification labels in this research experiment are movie science fiction, biography, action, and thriller.

We utilized Weka 3.81 and hypertext processor language *(*PHP*)* as main program. The initial stage of this experiment is preprocessing data aimed at clearing data. Preprocessing is done before classification process or before optimization process with PSO. This is done to remove text with spelling that is nonstandard that can interfere with data processing. The steps in preprocessing are: first from the raw data cleared regular expression or special character in the data synopsis used as experimental training data. Next is a cleanup of links, URLs or links if found in the synopsis data.

Data that has been through the preprocessing stage is done by weighting by transforming data from text form to weight based on term frequency. The transformation results make the word a feature for each data. In this study, because the experiments performed will compare and in the end is to find the optimum value of N-gram, the transformation phase and weighting of the text into features will be done 6 times for each N-gram n value of 1 to 6.

No.	Review	Genre
1	John Person is an out-of-work actor living in Los Angeles, with a credit card debt of over $27,000. Across the hall from him lives his friend, Grace. One night, his nebbish neighbor Neely, who wears a neck brace, invades his apartment with an unusual request: deliver a large blue suitcase to the truck stop of Baker, California, where it will be picked up by a man named "Cowboy", for which he will be given $25,000.....	scifi
2	On July 3, 1984 at 5:30PM EDT, at the Uneeda medical supply warehouse in Louisville, Kentucky, a foreman named Frank tries to impress the company's newest employee, Freddy, by showing him military drums that accidentally wound up in the basement of the building. The drum contains the remains of an army experiment gone wrong that inspired the film Night of	scifi

Figure 4: Dataset preprocessing

No.	Review	Regular Expression at Special Character
1	John Person is an out-of-work actor living in Los Angeles, with a credit card debt of over $27,000. Across the hall from him lives his friend, Grace. One night, his nebbish neighbor Neely, who wears a neck brace, invades his apartment with an unusual request: deliver a large blue suitcase to the truck stop of Baker, California, where it will	$, “

Figure 5: Cleaning Regular Expression or Special Character

In this experiment carried out is to find the optimum N-gram value, the transformation stage and the weighting of the text into features will be performed 6 times for each of N-Grams.
Moreover, we set PSO parameter as follows:

Table 1 PSO parameter

Parameter	value
C_1	1
C_2	1
r_1	0.1
r_2	0.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2 Feature Weighting Unigram

	escapes	from	jail	along	with
action	1.00	2.00	1.00	2.00	6.00
action	0.00	2.00	0.00	0.00	1.00
action	0.00	0.00	0.00	0.00	0.00
action	0.00	1.00	0.00	0.00	2.00
action	0.00	2.00	0.00	0.00	1.00
...	...	...	...	...	...
biography	0.00	2.00	0.00	0.00	1.00
biography	0.00	1.00	0.00	0.00	2.00
biography	0.00	1.00	0.00	0.00	5.00
biography	0.00	1.00	0.00	0.00	1.00
biography	1.00	2.00	0.00	0.00	4.00
...	...	...	...	...	...
scifi	0.00	3.00	0.00	1.00	2.00

Table 3 Feature Weighting Biigram

	escapes from	from jail	jail along	along with	with raghu
action	1.00	1.00	1.00	2.00	1.00
action	0.00	0.00	0.00	0.00	0.00
action	0.00	0.00	0.00	0.00	0.00
action	0.00	0.00	0.00	0.00	0.00
action	0.00	0.00	0.00	0.00	0.00
...	...	...	...	...	...
biography	0.00	0.00	0.00	0.00	0.00
biography	1.00	0.00	0.00	0.00	0.00
biography	0.00	0.00	0.00	1.00	0.00
biography	0.00	0.00	0.00	0.00	0.00
biography	0.00	0.00	0.00	0.00	0.00
...	...	...	...	...	...
scifi	0.00	0.00	0.00	0.00	0.00
scifi	0.00	0.00	0.00	0.00	0.00

Table 4 Feature Weighting Trigram

	escapes from jail	from jail along	jail along with	along with raghu	with raghu cell
action	1.00	1.00	1.00	1.00	1.00
action	0.00	0.00	0.00	0.00	0.00
action	0.00	0.00	0.00	0.00	0.00
action	0.00	0.00	0.00	0.00	0.00
action	0.00	0.00	0.00	0.00	0.00
...	...	...	...	...	...
biography	0.00	0.00	0.00	0.00	0.00
biography	0.00	0.00	0.00	0.00	0.00
biography	0.00	0.00	0.00	0.00	0.00
biography	0.00	0.00	0.00	0.00	0.00
biography	0.00	0.00	0.00	0.00	0.00
...	...	...	...	...	...

Based on the classification result, the resulting confusion matrix is as follows:

Table 5 Confusion Matrix Unigram Naive Bayes

		Predicted class			
		action	biography	scifi	Thriller
Actual class	action	73	6	17	4
	bio	7	87	4	2
	scifi	4	2	92	2
	thriller	6	3	15	76

Table 6 Confusion Matrix bigram Naive Bayes

		Predicted class			
		action	biography	scifi	thriller
Actual class	action	80	7	8	5
	biography	5	91	3	1
	scifi	6	1	91	2
	thriller	11	5	6	78

Table 7 Confusion Matrix trigram Naive Bayes

		Predicted class			
		action	biography	scifi	thriller
Actual class	action	73	9	14	4
	biography	2	82	11	5
	scifi	4	3	86	7
	thriller	7	4	10	79

Table 8 Confusion Matrix 4-gram Naive Bayes

		Predicted class			
		action	biography	scifi	thriller
Actual class	action	61	16	22	1
	biography	5	80	15	0
	scifi	3	12	82	3
	thriller	7	13	15	65

Table 9 Confusion Matrix 5-gram Naive Bayes

		Predicted class			
		action	biography	scifi	thriller
Actual class	action	43	22	35	0
	biography	0	81	16	3
	scifi	6	8	84	2
	thriller	0	26	18	56

Table 10 Confusion Matrix 6-gram Naive Bayes

		Predicted class			
		action	biography	scifi	thriller
Actual class	action	22	36	42	0
	biography	0	85	14	1
	scifi	4	5	88	3
	thriller	0	27	36	37

After testing with training data with several experiments with different N-Gram values, the following results were obtained:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 11 Result of Naïve Bayes N-gram Classification

	Accuracy	Precision	Recall
1-gram	82%	82%	83,05%
2-grams	85%	85%	85,22%
3-grams	80%	80%	80,69%
4-grams	72%	72%	75,44%
5-grams	66%	66%	76,36%
6-grams	58%	58%	69,82%

Based on the results of the Naïve Bayes classification with the N-Gram trial from unigram to 6-grams, the results obtained with the best performance and accuracy are the bigram model (2-grams) Naïve Bayes.

Table 12 The PSO Process Gets the Best Particle Position

iteration	position	G-Best	particle value
1	[1,1,4,5,6]	1	82
2	[1,1,4,5,6]	1	82
3	[1,1,4,5,5]	1	82
4	[1,1,4,5,5]	1	82
5	[1,1,4,4,5]	1	82
6	[1,1,4,4,4]	1	82
7	[1,1,4,4,4]	1	82
8	[1,1,3,4,4]	1	82
9	[1,1,3,4,4]	1	82
10	[1,1,3,3,4]	1	82
11	[1,1,3,3,4]	1	82
12	[1,1,3,3,3]	1	82
13	[1,1,3,3,3]	1	82
14	[1,1,3,3,3]	1	82
15	[1,1,3,3,3]	1	82
16	[1,1,3,3,3]	1	82
17	[1,1,2,3,3]	2	85
18	[1,1,2,3,3]	2	85
19	[1,1,2,3,3]	2	85
20	[1,1,2,3,3]	2	85
21	[1,1,2,2,3]	2	85
22	[1,1,2,2,3]	2	85
23	[1,2,2,2,3]	2	85
24	[1,2,2,2,2]	2	85
25	[1,2,2,2,2]	2	85
26	[1,2,2,2,2]	2	85
27	[2,2,2,2,2]	2	85

Based on the PSO process up to the 27th iteration and get the convergence of the particle position is 2, which means it converge at 2-gram position. Then the best position is the optimum value of *N-gram* generated through the PSO calculation. Here are the results of the *Naive Bayes 2-gram* model classification test can be summarized as follows:

		Predicted class			
		action	biography	scifi	thriller
Actual class	action	80	7	8	5
	bio	5	91	3	1
	scifi	6	1	91	2
	thriller	11	5	6	78

	Accuracy	Precision	Recall
Bigram	85%	85%	85,22%

6. DISCUSSION

From experiments sections, we know that PSO able to obtain better result of selecting n-grams by using iteration. This approach is implicitly better and faster than brute search method due PSO could converge faster to obtain the desired results for big volume of text mining data.

The disadvantages of our method are relied on the nature of PSO itself, due to low convergence rate in the iterative process and easy to fall into local minimum. Due to this PSO nature, adaptive particle swarm optimization can be used.

7. CONCLUSION AND FUTURE WORKS

From the results of the research and testing of the *Naive Bayes N-gram* classification model, and the optimization of *N-gram* value with PSO algorithm, at the end of the report we can give the following conclusion: 1) The proposed method for improving Naive Bayes N-gram model classification by determining the optimum value of N-gram using PSO algorithm has been successfully implemented to solve the genre classification case based on the review or synopsis; 2) Based on the experimental results, it is known that the effect of the application of PSO to determine the optimum value of N-gram in its implementation in the case of classification with Naive Bayes. Naive Bayes N-gram model with several experiments of N-gram value yielded the best accuracy result of 85% with N-gram 2 value. Time cost for obtaining optimum N-gram values can be summarized by implementing PSOs previously obtained randomly and trial and error. In this study the best particle position is 2 which is the next N-gram optimum for the classification process with Bigram Naive Bayes model.

Based on previous relevant research related to movie genre classification based on its synopsis, this research shows an increase of Precision value from 73,4% to 85%, Recall value from 67,63% to 85,22%, and best accuracy value obtained from this research is 85%.

Our research used particle swarm optimization which have disadvantages which has a low convergence rate in the iterative process and easy to fall into local minimum, thus adaptive particle swarm optimization can be used to elevate those disadvantages.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Future works that we will pursue is feature selection for text classification, which will take part not only in term frequency, but also using document frequency, mutual information, odds ratio and so on. In matter of feature projection, we also will take part and see how well the singular value decomposition, supervised clustering for dimensionality reduction and so on. In matter of classification, we will explore the possibilities using ensemble method not only in random forest, but also in adaboost, bagging or meta-classifiers.

It also worth mention, that movie genres classification could go to multi label learning, due to most movie have at least 2 genres categories such as thriller with horror, romance with drama, superhero with science fiction, etc.

Finally, the next step in our future research will take part in sentiment analysis, which is hot topic right now, we are using text classification research as base research to gaining more understandability and knowledge in regards of text mining field. Thus, polarity and sentiment analysis also can be applied in CMU movie corpus datasets.

REFERENCE

[1] F. Elghannam, "Text representation and classification based on bi-gram alphabet," *J. King Saud Univ. - Comput. Inf. Sci.*, Jan. 2019.

[2] F. Sebastiani, "Machine learning in automated text categorization," *ACM Comput. Surv.*, vol. 34, no. 1, pp. 1–47, Mar. 2002.

[3] C. C. Aggarwal and C. Zhai, "A Survey of Text Classification Algorithms," in *Mining Text Data*, Boston, MA: Springer US, 2012, pp. 163–222.

[4] D. Maynard and A. Funk, "Automatic Detection of Political Opinions in Tweets," 2012, pp. 88–99.

[5] F. Xie, X. Wu, and X. Zhu, "Efficient sequential pattern mining with wildcards for keyphrase extraction," *Knowledge-Based Syst.*, vol. 115, pp. 27–39, Jan. 2017.

[6] L. H. Lee, D. Isa, W. O. Choo, and W. Y. Chue, "High Relevance Keyword Extraction facility for Bayesian text classification on different domains of varying characteristic," *Expert Syst. Appl.*, vol. 39, no. 1, pp. 1147–1155, Jan. 2012.

[7] F. Figueiredo, L. Rocha, T. Couto, T. Salles, M. A. Gonçalves, and W. Meira Jr., "Word co-occurrence features for text classification," *Inf. Syst.*, vol. 36, no. 5, pp. 843–858, Jul. 2011.

[8] A. Ittoo, L. M. Nguyen, and A. van den Bosch, "Text analytics in industry: Challenges, desiderata and trends," *Comput. Ind.*, vol. 78, pp. 96–107, May 2016.

[9] B. S. Kumar and V. Ravi, "A survey of the applications of text mining in financial domain," *Knowledge-Based Syst.*, vol. 114, pp. 128–147, Dec. 2016.

[10] E. Chaix, L. Deléger, R. Bossy, and C. Nédellec, "Text mining tools for extracting information about microbial biodiversity in food," *Food Microbiol.*, vol. 81, pp. 63–75, Aug. 2019.

[11] V. Basto-Fernandes, I. Yevseyeva, J. R. Méndez, J. Zhao, F. Fdez-Riverola, and M. T.M. Emmerich, "A spam filtering multi-objective optimization study covering parsimony maximization and three-way classification," *Appl. Soft Comput.*, vol. 48, pp. 111–123, Nov. 2016.

[12] M. Giatsoglou, M. G. Vozalis, K. Diamantaras, A. Vakali, G. Sarigiannidis, and K. C. Chatzisavvas, "Sentiment analysis leveraging emotions and word embeddings," *Expert Syst. Appl.*, vol. 69, pp. 214–224, Mar. 2017.

[13] M. Kang, J. Ahn, and K. Lee, "Opinion mining using ensemble text hidden Markov models for text classification," *Expert Syst. Appl.*, vol. 94, pp. 218–227, Mar. 2018.

[14] C. Catal and M. Nangir, "A sentiment classification model based on multiple classifiers," *Appl. Soft Comput.*, vol. 50, pp. 135–141, Jan. 2017.

[15] A. García-Pablos, M. Cuadros, and G. Rigau, "W2VLDA: Almost unsupervised system for Aspect Based Sentiment Analysis," *Expert Syst. Appl.*, vol. 91, pp. 127–137, Jan. 2018.

[16] H. Altınçay and Z. Erenel, "Ternary encoding based feature extraction for binary text classification," *Appl. Intell.*, vol. 41, no. 1, pp. 310–326, Jul. 2014.

[17] S. Kang, S. Cho, and P. Kang, "Multi-class classification via heterogeneous ensemble of one-class classifiers," *Eng. Appl. Artif. Intell.*, vol. 43, pp. 35–43, Aug. 2015.

[18] B. Al-Salemi, S. A. Mohd Noah, and M. J. Ab Aziz, "RFBoost: An improved multi-label boosting algorithm and its application to text categorisation," *Knowledge-Based Syst.*, vol. 103, pp. 104–117, Jul. 2016.

[19] H. Elghazel, A. Aussem, O. Gharroudi, and W. Saadaoui, "Ensemble multi-label text

categorization based on rotation forest and latent semantic indexing," *Expert Syst. Appl.*, vol. 57, pp. 1–11, Sep. 2016.

[20] Y. Du, J. Liu, W. Ke, and X. Gong, "Hierarchy construction and text classification based on the relaxation strategy and least information model," *Expert Syst. Appl.*, vol. 100, pp. 157–164, Jun. 2018.

[21] V. K. A and G. Aghila, "A Survey of Naive Bayes Machine Learning approach in Text Document Classification," Mar. 2010.

[22] K. M. Al-Aidaroos, A. A. Bakar, and Z. Othman, "Naive bayes variants in classification learning," in *2010 International Conference on Information Retrieval & Knowledge Management (CAMP)*, 2010, pp. 276–281.

[23] R. Clement, "Ngram and Bayesian Classification of Documents for Topic and Authorship," *Lit. Linguist. Comput.*, vol. 18, no. 4, pp. 423–447, Nov. 2003.

[24] K. wing Ho, "Movies genres classification by synopsis," 2011.

[25] L. Khreisat, "A machine learning approach for Arabic text classification using N-gram frequency statistics," *J. Informetr.*, vol. 3, no. 1, pp. 72–77, Jan. 2009.

[26] A. Dey, M. Jenamani, and J. J. Thakkar, "Senti-N-Gram : An n -gram lexicon for sentiment analysis," *Expert Syst. Appl.*, vol. 103, pp. 92–105, Aug. 2018.

[27] B. J. Marafino, W. John Boscardin, and R. Adams Dudley, "Efficient and sparse feature selection for biomedical text classification via the elastic net: Application to ICU risk stratification from nursing notes," *J. Biomed. Inform.*, vol. 54, pp. 114–120, Apr. 2015.

[28] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, pp. 1942–1948.

[29] Mei-Ping Song and Guo-Chang Gu, "Research on particle swarm optimization: a review," in *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.04EX826)*, vol. 4, pp. 2236–2241.

[30] Sang-Bum Kim, Kyoung-Soo Han, Hae-Chang Rim, and Sung Hyon Myaeng, "Some Effective Techniques for Naive Bayes Text Classification," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 11, pp. 1457–1466, Nov. 2006.

[31] P. Domingos and M. Pazzani, "On the Optimality of the Simple Bayesian Classifier underZero-One Loss. Machine Learning," *Mach. Learn.*, vol. 29, pp. 103–130, 1997.

[32] S. B. Kotsiantis, I. D. Zaharakis, and P. E. Pintelas, "Machine learning: a review of classification and combining techniques," *Artif. Intell. Rev.*, vol. 26, no. 3, pp. 159–190, Nov. 2006.

[33] O. Nasraoui, *Web data mining*, vol. 10, no. 2. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009.

[34] B. C. Gencosman, H. C. Ozmutlu, and S. Ozmutlu, "Character n-gram application for automatic new topic identification," *Inf. Process. Manag.*, vol. 50, no. 6, pp. 821–856, Nov. 2014.

[35] M. Clerc, "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization," in *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, pp. 1951–1957.

[36] C. Sammut and G. I. Webb, Eds., *Encyclopedia of Machine Learning and Data Mining*. Boston, MA: Springer US, 2017.

[37] X. Deng, Q. Liu, Y. Deng, and S. Mahadevan, "An improved method to construct basic probability assignment based on the confusion matrix for classification problem," *Inf. Sci. (Ny).*, vol. 340–341, pp. 250–261, May 2016.

[38] D. and Bamman, B. and O'Connor, and N. A. Smith, "Learning Latent Personas of Film Characters," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2013, pp. 352–361.

[39] Y. Ji and L. Shang, "RoughTree A Classifier with Naive-Bayes and Rough Sets Hybrid in Decision Tree Representation," in *2007 IEEE International Conference on Granular Computing (GRC 2007)*, 2007, pp. 221–221.

[40] R. Abraham, J. B. Simha, and S. S. Iyengar, "Medical Datamining with a New Algorithm for Feature Selection and Naive Bayesian Classifier," in *10th International Conference on Information Technology (ICIT 2007)*, 2007, pp. 44–49.

[41] Z. Xie, W. Hsu, Z. Liu, and M. L. Lee, "SNNB: A Selective Neighborhood Based Naïve Bayes for Lazy Learning," 2002, pp. 104–114.