Standard Co-training in Multiword Expression Detection

Senem Kumova Metin(✉)

Department of Software Engineering, Faculty of Engineering,
Izmir University of Economics, Sakarya Caddesi, No. 156, Izmir, Turkey
senem.kumova@ieu.edu.tr

Abstract. Multiword expressions (MWEs) are units in language where multiple words unite without an obvious/known reason. Since MWEs occupy a prominent amount of space in both written and spoken language materials, identification of MWEs is accepted to be an important task in natural language processing.

In this paper, considering MWE detection as a binary classification task, we propose to use a semi-supervised learning algorithm, standard co-training [1] Co-training is a semi-supervised method that employs two classifiers with two different views to label unlabeled data iteratively in order to enlarge the training sets of limited size. In our experiments, linguistic and statistical features that distinguish MWEs from random word combinations are utilized as two different views. Two different pairs of classifiers are employed with a group of experimental settings. The tests are performed on a Turkish MWE data set of 3946 positive and 4230 negative MWE candidates. The results showed that the classifier where statistical view is considered succeeds in MWE detection when the training set is enlarged by co-training.

Keywords: Multiword expression · Classification · Co-training

1 Introduction

A learning machine and/or the task of learning requires experience in other words a training phase to learn. The method to obtain the experience puts the machine learning methods into 3 main categories: supervised, unsupervised and reinforcement learning algorithms. In supervised learning, a labeled data set is given to the machine during training. Following, the machine that gained the ability to label a given sample, may classify the testing samples. In unsupervised learning, the labels of the samples are not provided to the machine in training phase. The machine is expected to learn the structure of samples and varieties in unlabeled sample set and to extract the clusters it self. In reinforcement learning, the machine interacts with the dynamic environment and aims to reach a predefined goal. The training of the machine is provided by the rewards and penalties.

The supervised methods require a sufficient amount of labeled samples for training to achieve in classification of unlabeled data. However, in many problems it is not possible to provide that sufficient amount of labeled samples or preparation of such a

P. Horain et al. (Eds.): IHCI 2017, LNCS 10688, pp. 178–188, 2017.
https://doi.org/10.1007/978-3-319-72038-8_14

sample set is over costing. In such cases, the machine may be forced to learn from unlabeled data. This is why, the notion of semi-supervised learning is defined as a halfway between supervised and unsupervised learning [2].

In semi-supervised learning methods, commonly training is performed iteratively. In first iteration, a limited number of labeled samples are given to the machine to learn. After first iteration, the machine labels the unlabeled samples. The samples that are labeled most reliably are added to the labeled set and the machine is re-trained by this enlarged labeled set in next iteration. After a number of iterations, it is accepted that the learning phase is finished and the machine is ready to label unlabeled data set. In other group of semi-supervised methods, some constraints are defined to supervise the training phase [2].

The earliest implementation of semi-supervised learning approach is probably the self-training [2]. In self-training, a single machine, trained by labeled sample set, enlarges its own labeled set iteratively, by labeling the unlabeled set. An alternative method to self-training, co-training, is proposed by Blum and Mitchell [1]. The co-training aims to increase the classification performance by employing two classifiers that considers different views of the data to label the unlabeled samples during training phase. There exist several implementations of the method that are used to solve different problems such as word sense disambiguation [3], semantic role labeling [4], statistical parsing [5], identification of noun phrases [6], opinion detection [7], e-mail classification [8] and sentiment classification [9].

In this study, we examine the effect of co-training in an important natural processing task: multiword expression detection. The notion of multiword expression may be explained in a variety of different ways. Simply, MWEs are word combinations where words unite to build a new syntactical/linguistic or semantic unit in language. Since the words may change their meaning or roles in text while they form MWE, detection of MWEs has a critical role in language understanding and language generation studies. For example, the expression “lady killer” is a MWE meaning “an attractive man”. But if the meanings of the composing words are considered individually, the expression refers to something completely different. In MWE detection, it is believed that the links between the composing words of MWEs are stronger than the links between random combinations of words. The strength of these links is measured commonly by statistical and/or linguistics features that may be extracted from the given text or a text collection (e.g. [10–13]).

In a wide group of studies that aim identification of MWEs, the regarding task is accepted as a classification problem and several machine-learning methods are employed. For example, in [13] statistical features are considered together by supervised methods such as linear logistic regression, linear discriminant analysis and neural networks. In [12], multiple linguistically-motivated features are employed in neural networks to identify MWEs in a set of Hebrew bigrams (uninterrupted two word combinations). Several experiments are performed on Turkish data set with linguistics features by 10 different classifiers (e.g. J48, sequential minimization, k nearest neighbor) in [14].

In this study, we aim to examine the performance change in MWE recognition when co-training is employed. The paper is organized as following. We first present the

semi-supervised learning and co-training in Sect. 2. In Sect. 3, experimental setup is given. In Sect. 4 results are presented. And the paper is concluded in Sect. 5.

2 Semi-supervised Learning: Co-training

Semi-supervised methods are proposed in order to overcome the disadvantages of supervised learning when there is a lack of sufficient amount of labeled samples. The methods are reported to succeed in some cases when some assumptions such as smoothness, clustering, manifold and transduction hold.

Semi-supervised methods are mainly categorized in four groups: generative, low-density, graph-based models and change of representation [2]. In generative models, the main aim is modeling the class conditional density. Co-training [1] and expected maximization [15] methods are well-known examples of generative models. On the other hand, low-density separation methods such as transductive support vector machine proposed by [16] try to locate decision boundaries in low density regions and away from the unlabeled samples. The methods presented in [17–19] are the examples of graph based methods where each node represents a sample and classification is performed by measuring the distance between nodes. In change of representation approach, a two-stage training is required. Since labeled samples are considered without their labels in the first stage, it is accepted that the representation of samples are changed by this way. In the second stage of training, unlabeled samples are excluded from the data set and supervised learning is performed with the new measure/kernel.

In this study, the semi-supervised method: co-training is implemented to identify MWEs. The co-training algorithm, given in Fig. 1, that will be named as standard co-training is proposed by [1].

Given
- Labeled data set L
- Unlabeled data set U

Repeat until there is no sample in set U:

Train first classifier, $f^{(1)}$, by labelled data set L, using only $x^{(1)}$ features
Train second classifier, $f^{(2)}$, by labelled data set L, using only $x^{(2)}$ features

Label/classify samples of unlabelled data set U by first classifier, $f^{(1)}$.
Label/classify samples of unlabelled data set U by second classifier, $f^{(2)}$.

Insert n number of negative and p number of positive samples that are most confidently labelled by first classifier to the labelled data set L.

Insert n number of negative and p number of positive samples that are most confidently labelled by second classifier to the labelled data set L.

Remove the samples that are inserted to the labelled set from the unlabelled set U.

Fig. 1. Standard co-training algorithm [1]

In standard co-training, the main aim is building a classifier trained by L number of labeled and U number of unlabeled samples where L is known to be a small number. In order to overcome the disadvantage of having a limited number of labeled samples, L, [1] proposed to split the feature vector in two groups of features where each group of features represents a different view of the regarding data set. Each group of features/split/view is used to train one of the classifiers. The assumptions that guarantee the success of co-training are explained as [1]

- Both groups of features (splits/views) must be available for classification.
- Given the label, the feature groups must be conditionally independent for each sample in the data set.

In several studies such as [6, 20], the researchers investigated to what degree these assumptions and the data set size effect the performance of co-training algorithm. For example, experimenting on the same problem mentioned in [1, 20] reported that even if the independency assumption is not satisfied, still co-training performs better than to alternatively proposed expected maximization algorithm since in each iteration all the samples are compared to others to determine the most confidently labeled ones in co-training.

The standard co-training algorithm is implemented to classify web pages in [1]. The first group of features is built by the words in web pages and the second group includes the words in the web links. In both classifiers, Naive Bayes algorithm is used and the tests are performed with $p = 1$ and $n = 3$. In [1], it is reported that the proposed co-training algorithm reaches to higher classification performance compared to supervised machine learning.

3 Experimental Setup

The experiments to examine performance of co-training in MWE detection require the following four tasks to be performed:

1. Two different views (two groups of features) of data set must be determined
2. The classifier pairs must be chosen
3. MWE data set composed of both positive and negative samples must be prepared/selected.
4. Labeled, unlabeled and testing data set sizes must be set.
5. Evaluation measures must be determined.

We propose to use linguistic and statistical features as two different views on MWE data set. In this study, the linguistic view includes 8 linguistic features listed below:

1. *Partial variety in surface forms (*$PVSF\_{m}$ and $PVSF\_{n)}$: In MWE detection studies, it is commonly accepted that MWEs are not observed in a variety of different surface forms in language. As a result, the histogram presenting the occurrence frequencies of different surface forms belonging to the same MWE is expected to be non-uniform [12]. We measured variety in surface forms in two different ways that are called as $PVSF\_{m}$ and $PVSF\_{n}$ features based on the surface form histogram,

similar to [12]. Briefly, the Manhattan distance between the actual surface form histogram of the MWE candidate and the possible/expected uniform histogram is employed as $PVSF\_{m}$. The ratio of $PVSF\_{\mathrm{m}}$ to total occurrence frequency of the candidate (in any form) is accepted as $PVSF\_{\mathrm{n}}$.

2. *Orthographical variety* (OV_h and OV_a): MWEs may hold orthographical changes due to the use of some punctuation marks such as hyphen. For example, expression "e mail" is commonly written as "e-mail". In this study we considered two punctuation marks and employed a Turkish corpus to obtain the feature values. The first punctuation mark is the hyphen. OV_h value is the proportion of the occurrence frequencies of candidate that is formed with a hyphen and without a hyphen. The second orthographical variety feature is OV_a. In this feature, the occurrences of the candidate with and without apostrophe symbol in the second composing word are counted. The ratio of the occurrences with and without apostrophe is employed as OV_a.
3. *Frozen Form:* It is a binary feature that is one if the MWE candidate has a single surface form in corpus and zero other vice.
4. *The ratio of Uppercase Letters:* The feature is simply the ratio of occurrence frequency of MWE candidate where capital letters are used to the total frequency of the candidate in the corpus.
5. *The suffix sequence (SS):* It is expected that a number of suffixes or suffix sequences are to be used with MWEs more than random word/word combinations. In order to determine such suffixes, a set of Turkish idioms is built. The suffixes of length [3 10] (in characters) that are commonly used with the idioms are determined in a Turkish corpus. And *SS* value of the MWE candidate is obtained by comparing the last *n* characters of the candidate with these suffix sequences. If there exists a match, the number of characters of regarding suffix is employed as *SS* feature value.
6. *Named Entity Words (NEW):* A list of words (3626 words) that are commonly used in Turkish named entities (e.g. personal names, locations, addresses) is prepared to obtain *NEW* feature values. The list includes 5 different categories of named entities. If a composing word of the given MWE candidate is observed in one of these categories, *NEW* value is increased by one. As a result, for each word in MWE candidate, *NEW* value may be increased to five theoretically.

The statistical view includes 18 features (Table 1). These features are known to be commonly used in many studies (e.g. [10, 13, 21]). In Table 1, w_1 and w_2 represent the first and the second word in given MWE candidate, respectively.

In Table 1, $P(w_1w_2)$ is the probability of co-occurrence of two words w_1 and w_2 sequentially. $P(w_1)$ and $P(w_2)$ are the occurrence probabilities of first and the second words. $P(w_i|w_j)$ gives the conditional occurrence probability of the word w_i given that the word w_j is observed. $f(w_1w_2), f(w_1), f(w_2)$ are occurrence frequency of the bigram w_1w_2, and the words w_1 and w_2 respectively. The different number of words following the bigram is represented by $v_f\ (w_1w_2)$, different number of words preceding and following the bigram is $v_b\ (w_1w_2)$ and $v_f\ (w_1w_2)$, respectively.

In this study, the classifiers *SMO* (Sequential Minimal Optimization) [22, 23], *J48* [24] and logistic regression (*Logistic*) [25] are employed in classifier pairs as presented in Table 2. A Turkish MWE data set that includes 8176 samples of MWE candidates

Table 1. Statistical features

Feature	Formula
Bigram-backward variety	$\frac{v_b(w_1w_2)}{f(w_1w_2)}$
Bigram-forward variety	$\frac{v_f(w_1w_2)}{f(w_1w_2)}$
Bigram–word forward variety	$\frac{v_f(w_1w_2)}{v_f(w_2)}$
Fager	$\frac{f(w_1w_2)}{\sqrt{(f(w_1w_2)+f(w_1\bar{w}_2).(f(w_1w_2)+f(\bar{w}_1w_2)}} - \frac{1}{2}\max(f(w_1\bar{w}_2)f(\bar{w}_1w_2))$
First Kulcznsky	$\frac{f(w_1w_2)}{f(w_1\bar{w}_2)+f(\bar{w}_1w_2)}$
Jaccard	$\frac{f(w_1w_2)}{f(w_1w_2)+f(w_1\bar{w}_2)+f(\bar{w}_1w_2)}$
Joint probability	$P(w_1w_2)$
Mutual dependency	$\log\frac{P(w_1w_2)^2}{P(w_1)P(w_2)}$
Normalized expectation	$\frac{2f(w_1w_2)}{f(w_1)+f(w_2)}$
Neighborhood unpredictability (*NUP*) [11]	$FNUP(w_1w_2) = 1 - \frac{v_f(w_1w_2)-1}{v_f(w_2)-1}$ $BNUP(w_1w_2) = 1 - \frac{v_b(w_1w_2)-1}{v_b(w_1)-1}$ $NUP(w_1w_2) = \sqrt{FNUP(w_1w_2)^2 + BNUP(w_1w_2)^2}$
Point-wise mutual information	$\log\frac{P(w_1w_2)}{P(w_1)P(w_2)}$
Piatersky-Shapiro	$P(w_1w_2) - P(w_1)P(w_2)$
R cost	$\log\left(1+\frac{f(w_1w_2)}{(f(w_1w_2)+f(w_1\bar{w}_2))}\right) + \log\left(1+\frac{f(w_1w_2)}{(f(w_1w_2)+f(\bar{w}_1w_2))}\right)$
S cost	$\log(1+\frac{\min(f(w_1\bar{w}_2),f(\bar{w}_1w_2)}{f(w_1w_2)+1})$
U cost	$\log(1+\frac{\min(f(w_1\bar{w}_2),f(\bar{w}_1w_2))+f(w_1w_2)}{\max(f(w_1\bar{w}_2),f(\bar{w}_1w_2))+f(w_1w_2)})$
Second Kulcznsky	$\frac{1}{2}(\frac{f(w_1w_2)}{(f(w_1w_2)+f(w_1\bar{w}_2))} + \frac{f(w_1w_2)}{(f(w_1w_2)+f(\bar{w}_1w_2))})$
Second Sokal-Sneath	$\frac{f(w_1w_2)}{f(w_1w_2)+2(f(w_1\bar{w}_2)+f(\bar{w}_1w_2))}$
Word forward variety	$\frac{v_f(w_2)}{f(w_2)}$

(3946 positive (MWE labeled) and 4230 negative (non MWE labeled)) is utilized in experiments.

Table 3 presents the sizes of labeled (L), unlabeled (U) and test (T) data sets. For example, in experimental setting no 1, 50 samples are used in labeled set, unlabeled set has 250 samples and test size is set as 100.

The evaluation of the classification is performed by F1 measure. F1 measure is given as

$$F1 = \frac{2TP}{2TP + FN + FP} \tag{1}$$

Table 2. Classifier pair

Classifier pair	Linguistics classifier	Statistical classifier
1	*J48*	*Logistic*
2	*SMO*	*SMO*

Table 3. Data sets

Setting no	L (Labeled set size)	U (Unlabeled set size)	T (Test set size)
1	50	250	100
2	100	200	100
3	200	100	100
4	50	700	250
5	100	650	250
6	200	550	250
7	500	250	250
8	50	950	300
9	100	900	300
10	200	800	300
11	500	500	300
12	750	250	300

where *TP* is the number of true positives (candidates that are both expected and predicted to belong to the same class MWE or non-MWE), *FN* is the number of false negatives, *FP* is the number of false positives.

4 Results

The performance of standard co-training, given in Fig. 1, is examined on test settings by repeating the same experiment 5 times (5 runs) for each setting. The numbers of positive (p) and negative samples (n) that will be inserted to the labeled data set in each iteration are set to one. And in each run of the tests, the data set is shuffled to build the labeled L, unlabeled U and test sets randomly. Table 4 gives the average evaluation results of the regarding tests. In Table 4,

- *Fi*, is the average F1 value that is obtained when classifier is trained by the labeled data set L,
- *Fc*, is the average F1 value that is obtained when classifier is trained with enlarged data set (U + L) (the resulting/final training set after co-training),
- *Fs*, is the average F1 value that is obtained when enlarged data set (U + L) is used in training with the actual (not expected) labels of the samples.
- *CP* column includes classifier pairs employed in the study. The first method in *CP* cells is the statistical classifier and the second method represents the linguistic classifier. For example, J48 is statistical and logistic is linguistic classifier.

Table 4. Testing results of standard co-training.

			Statistical Classifier Results			Linguistic Classifier Results		
CP	Test/U+L	L	*Fi*	*Fc*	*Fs*	*Fi*	*Fc*	*Fs*
J48-LOGISTIC	100/300	50	0,50	0,60	0,68	0,59	0,58	0,63
		100	0,58	0,62	0,68	0,60	0,57	0,63
		200	0,63	0,66	0,68	0,61	**0,63**	0,63
	250/750	50	0,52	0,57	0,65	0,61	0,53	0,62
		100	0,57	0,62	0,65	0,60	0,57	0,62
		200	0,55	0,62	0,65	0,60	0,58	0,62
		500	0,61	**0,67**	0,65	0,61	0,61	0,62
	300/1000	50	0,51	0,57	0,65	0,61	0,55	0,62
		100	0,56	0,61	0,65	0,63	0,56	0,62
		200	0,56	0,63	0,65	0,63	0,60	0,62
		500	0,57	**0,66**	0,65	0,63	**0,62**	0,62
		750	0,60	0,64	0,65	0,63	**0,62**	0,62
SMO-SMO	100/300	50	0,50	0,55	0,71	0,60	0,61	0,66
		100	0,56	0,63	0,71	0,63	0,63	0,66
		200	0,63	0,68	0,71	0,66	**0,66**	0,66
	250/750	50	0,52	0,55	0,68	0,63	0,58	0,66
		100	0,56	0,58	0,68	0,64	0,64	0,66
		200	0,56	0,60	0,68	0,66	0,65	0,66
		500	0,62	**0,68**	0,68	0,66	**0,66**	0,66
	300/1000	50	0,52	0,56	0,68	0,64	0,58	0,67
		100	0,55	0,59	0,68	0,65	0,62	0,67
		200	0,56	0,63	0,68	0,67	0,66	0,67
		500	0,57	**0,68**	0,68	0,67	**0,67**	0,67
		750	0,63	**0,69**	0,68	0,67	**0,67**	0,67

The shaded regions in Table 4 show the settings in which $Fi \geq Fc$, meaning that when training set is enlarged with co-training, F1 value increases. It is observed that standard co-training succeeds for all settings in statistical classifier. The cells that hold bold F1 values represent the settings where $Fc \geq Fs$, meaning that the training set that is enlarged by co-training is more successful in supervising the classifier when compared to the same data set with human annotated labels of samples.

Table 5 gives minimum, average and maximum F1 values of both classifiers for three different cases:

1. *Classification (L):* This is the case where labeled set L is employed in training
2. *Standard co-training:* Standard co-training is employed to enlarge the training set size to U + L.
3. *Classification (U + L):* Classifiers are trained by U + L samples that are labeled by human annotators.

Table 5. F1 results with and without co-training.

		Statistical classifier			Linguistic classifier		
	Classifier pair	*Min*	*Ave*	*Max*	*Min*	*Ave*	*Max*
Classification (L)	*J48-Logistic*	0,50	0,56	0,63	0,59	0,61	0,63
	SMO-SMO	0,50	0,57	0,63	0,60	0,65	0,67
Standard Co-training	*J48-Logistic*	0,57	0,62	0,67	0,53	0,59	0,63
	SMO-SMO	0,55	0,62	0,69	0,58	0,64	0,67
Classification (U + L)	*J48-Logistic*	*0,65*	0,66	0,68	0,62	0,62	0,63
	SMO-SMO	*0,68*	0,69	0,71	0,66	0,67	0,67

From Table 5, three important outputs are observed. These are:

1. Standard co-training succeeds in training for both classifier pairs in statistical classifier. On the other hand, it is observed that for linguistic classifier, co-training generates lower/equal F1 values when compared to training with a limited number of samples (L).
2. Overall, *SMO-SMO* classifier pair outperforms *J48-Logistic* classifier pair in terms of average and maximum F1 values.
3. The highest performance in co-training (0.69) is obtained with *SMO-SMO* pair. It is observed that the increase in F1 value reached to an acceptable level (0.69 – 0.63 = 0.06) for this classifier pair.

5 Conclusion

In this study, we present our efforts to improve the performance of MWE detection by the use of standard co-training algorithm. The results showed that especially for the classifier that employs statistical features in classification, performance is improved by co-training. As a further work, we plan to apply different versions of co-training and run the tests with different types of classifiers.

Acknowledgement. This work is carried under the grant of TÜBİTAK – The Scientific and Technological Research Council of Turkey to Project No: 115E469, Identification of Multi-word Expressions in Turkish Texts.

We thank to Mehmet Taze, Hande Aka Uymaz, Erdem Okur and Levent Tolga Eren for their efforts in labeling MWE data set.

References

1. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: Proceedings of the Eleventh Annual Conference on Computational Learning Theory - COLT 1998, pp. 92–100 (1998)
2. Olivier, C., Schölkopf, B., Zien, A.: Semi-Supervised Learning (2006)

3. Mihalcea, R.: Co-training and self-training for word sense disambiguation. In: Language Learning, pp. 182–183 (2003)
4. He, S., Gildea, D.: Self-training and Co-training for Semantic Role Labeling, New York, Primary Report 891, 13 (2006)
5. Sarkar, A.: Applying Co-Training methods to statistical parsing. In: ACL, pp. 175–182 (2001)
6. Pierce, D., Cardie, C.: Limitations of co-training for natural language learning from large datasets. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1–9 (2001)
7. Yu, N.: Exploring co-training strategies for opinion detection. J. Assoc. Inf. Sci. Technol. **65**, 2098–2110 (2014)
8. Kiritchenko, S., Matwin, S.: Email classification with co-training. In: Proceedings of the 2001 Conference on Centre for Advanced Studies on Collaborative Research, pp. 301–312 (2001)
9. Wan, X.: Co-training for cross-lingual sentiment classification (2009)
10. Metin, S.K., Karaoğlan, B.: Collocation extraction in Turkish texts using statistical methods. In: Loftsson, H., Rögnvaldsson, E., Helgadóttir, S. (eds.) NLP 2010. LNCS, vol. 6233, pp. 238–249. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14770-8_27
11. Metin, S.K.: Neighbour unpredictability measure in multiword expression extraction. Comput. Syst. Sci. Eng. **31**, 209–221 (2016)
12. Tsvetkov, Y., Wintner, S.: Identification of multiword expressions by combining multiple linguistic information sources. Comput. Linguist. **40**, 449–468 (2014)
13. Pecina, P.: A machine learning approach to multiword expression extraction. In: LREC Workshop Towards a Shared Task for Multiword Expressions, pp. 54–61 (2008)
14. Kumova Metin, S., Taze, M., Aka Uymaz, H., Okur, E.: Multiword expression detection in Turkish using linguistic features. In: 25th Signal Processing and Communications Applications Conference, pp. 1–3 (2017)
15. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using EM. Mach. Learn. **39**, 103–134 (2000)
16. Vapnik, V.N.: Statistical Learning Theory. Wiley, New York (1998)
17. Szummer, M., Jaakkola, T.: Information regularization with partially labeled data. Adv. Neural. Inf. Process. Syst. **15**, 1049–1056 (2002)
18. Zhu, X., Ghahramani, Z.: Learning from labeled and unlabeled data with label propagation. Neuroscience (2002)
19. Belkin, M., Niyogi, P.: Using manifold structure for partially labelled classification. In: Nips 2002, pp. 271–277 (2002)
20. Nigam, K., Ghani, R.: Analyzing the effectiveness and applicability of co-training. In: Proceedings of the Ninth International Conference on Information and Knowledge Management - CIKM 2000, pp. 86–93 (2002)
21. Metin, S.K., Karaoğlan, B.: Türkiye Türkçesinde Eşdizimlerin İstatistiksel Yöntemlerle Belirlenmesi. J. Soc. Sci. Turkic World. Summer, 253–286 (2016)
22. Platt, J.C.: Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines (1998)
23. Shevade, S.K., Keerthi, S.S., Bhattacharyya, C., Murthy, K.R.K.: Improvements to the SMO algorithm for SVM regression. IEEE Trans. Neural Netw. **11**, 1188–1193 (2000)
24. Quinlan, J.R.: C4.5: Programs for Machine Learning (1992)
25. Tabachnick, B.G., Fidell, L.S.: Using Multivariate Statistics. Allyn & Bacon, Boston (2007)