Engineering Applications of Artificial Intelligence 62 (2017) 152–163

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Instance labeling in semi-supervised learning with meaning values of words

Berna Altınel[a,*], Murat Can Ganiz[a], Banu Diri[b]

[a] Engineering Faculty, Department of Computer Engineering, Marmara University, Turkey
[b] Engineering Faculty, Department of Computer Engineering, Yıldız Technical University, Turkey

ARTICLE INFO

Keywords:
Text classification
Semantic kernel
Semi-supervised learning
Instance labeling
Helmholtz principle

ABSTRACT

In supervised learning systems; only labeled samples are used for building a classifier that is then used to predict the class labels of the unlabeled samples. However, obtaining labeled data is very expensive, time consuming and difficult in real-life practical situations as labeling a data set requires the effort of a human expert. On the other side, unlabeled data are often plentiful which makes it relatively inexpensive and easier to obtain. Semi-Supervised Learning methods strive to utilize this plentiful source of unlabeled examples to increase the learning capacity of the classifier particularly when amount of labeled examples are restricted. Since SSL techniques usually reach higher accuracy and require less human effort, they attract a substantial amount of attention both in practical applications and theoretical research. A novel semi-supervised methodology is offered in this study. This algorithm utilizes a new method to predict the class labels of unlabeled examples in a corpus and incorporate them into the training set to build a better classifier. The approach presented here depends on a meaning calculation, which computes the words' meaning scores in the scope of classes. Meaning computation is constructed on the Helmholtz principle and utilized to various applications in the field of text mining like feature extraction, information retrieval and document summarization. Nevertheless, according to the literature, ILBOM is the first work which uses meaning calculation in a semi-supervised way to construct a semantic smoothing kernel for Support Vector Machines (SVM). Evaluation of the proposed methodology is done by performing various experiments on standard textual datasets. ILBOM's experimental results are compared with three baseline algorithms including SVM using linear kernel which is one of the most frequently used algorithms in text classification field. Experimental results show that labeling unlabeled instances based on meaning scores of words to augment the training set is valuable, and increases the classification accuracy on previously unseen test instances significantly.

1. Introduction

Text categorization is a popular task whose aim is to label documents according to predefined class labels. There is a big amount of textual data collected on the internet especially on social networks, microblogging sites, blogs, forums, news, etc. This tremendous amount of texts continues to enlarge by the contributions of millions of people every day. Automatically processing and extracting meaning from these great amounts of documents is one of the main difficulties not only for research platforms but also for commercial platforms. The text classification plays a very important role in several popular and widely used applications such as document filtering, sentiment classification, information extraction, summarization and question answering. It is also significant to remember that, one of these applications is likely to be a part of a very important military, health and security engineering problem in real world cases. Nevertheless a very big portion of the accumulated data consists of unlabeled samples.

Bag of Words (BOW) is traditional representation methodology of unstructured textual data in the literature. Each of these terms in the same document represents an independent dimension in a vector space (Salton and Yang, 1973). There is no order of terms in BOW feature demonstration. Also, a bag is able to be demonstrated as a vector as well as a group of bags is able to be demonstrated as a matrix. The rows of this matrix represent the documents and columns of this matrix represent the corresponding term frequencies of these documents; which is called Vector Space Model (VSM). This approach mainly emphasizes the frequency of terms. The BOW methodology makes the representation of words simpler in documents by disregarding the following semantic and syntactic relations between words in natural language: 1.) It assumes independency between words, since it ignores the semantic connections among words. This will be an important problem especially for the documents which include multi-word expressions. 2.) It processes polysemous words like a particular unit. For example, word "bank" could have two distinct meanings according

* Corresponding author.
E-mail addresses: berna.altinel@marmara.edu.tr (B. Altınel), murat.ganiz@marmara.edu.tr (M. Can Ganiz), banu@ce.yildiz.edu.tr (B. Diri).

http://dx.doi.org/10.1016/j.engappai.2017.04.003
Received 30 November 2016; Received in revised form 24 February 2017; Accepted 3 April 2017

to the context it appears; one is a financial institution and the other is a river side (Wang and Domeniconi, 2008). 3.) It maps synonymous terms into completely different entities (Salton and Yang, 1973). Each class of texts has two forms of vocabulary: i)"core" vocabulary that is related to the theme of that class, ii) "general" vocabulary which may have almost identical distributions in distinct classes like stop words as Steinbach et al. (2000) analyze and discuss. Therefore, two unlike documents which cover completely distinct topics and belong to different classes may have several general terms in common as well as may have high similarity value according to their BOW feature demonstration.

An expected output of accurate and efficient text classification algorithms is to label unlabeled textual materials based on specified classes that comprise of identical textual materials. On account of accomplishing this goal, there are various classification methods which based on distance or similarity measures. These similarity measures compare pairs of documents and compute their similarities. It is also known that vector space demonstration of texts results sparsity and high dimensionality. This is a very big difficulty especially when there are numerous class labels however an inadequate training data. Hence it is critical that a successful and accurate text classifier should scale well with the large number of classes and features under the circumstances of restricted training data. However, rather preferably, terms in documents convey semantic information, i.e., the sense carried by the words of the textual materials. Therefore, a perfect text classification system should be able to take advantage of this semantic information.

Semantic text classification groups the documents into meaningful classes. In these kinds of classifiers, semantic connections among the words and the documents are taken into consideration. The texts which are semantically correlated to each other are classified with the same class label while the texts which are semantically unconnected are classified with different class labels. Semantic classification algorithms can also help in detecting the subject of a class. Semantic classification methodologies focuses on meanings of the terms and therefore the semantic approach mostly uses a dictionary or statistical calculations extracted from the corpus to build the classifier and then classify the test instances.

Advantages of semantic text classification over traditional text classification could be listed as follows:

- Semantic text classification algorithms help in information and relationship detection among words of the texts.
- Semantic text classification algorithms can contribute semantically relating the classes to one another.
- Semantic text classification approach can give the opportunity to extract the latent relationships between words and documents.
- Semantic text classification algorithms can generate meaningful keywords for the existing classes.
- Common text classification methods have poor capabilities in explaining to users why a certain result is achieved because traditional text classification algorithms cannot relate semantically to nearby terms. As well, they cannot explain how the result clusters are related to one another. But on the other side, the good news is that semantic text classification algorithms have the capability to locate the instances semantically, explain and analyze the classification results.
- Traditional text classification methods focus on only syntax that produces poor classification results. So, semantic understanding of text is necessary to improve progress of the efficiency and accuracy of classification.
- Synonymy is a term or phrase that means exactly or nearly the same as another word or phrase in the same language even though they are written differently. Polysemy is the ability for a term or phrase to have more than one meaning. Many languages have several synonyms. For instance "peak-summit", "minuscule-minute" pairs are synonyms in English. There are also many polysemous terms in English. For example, the verb "to get" can mean "procure", "understand" (I get it), etc. Traditional text classifiers cannot make use of semantic approaches and they only concentrate on syntax in a document. Thus, they ignore the semantic connections between words and documents and they evaluate a word as it is independent from its context. Conversely, semantic text classification algorithms have the opportunity to handle synonymy and polysemy better than traditional text classification algorithms since they take advantages of semantic connections between words. Consequently, semantic approaches make semantic classification algorithms assess and interpret a word within its context.

In machine learning applications, especially in the field of text classification there are two conventional strategies; supervised learning and unsupervised learning. A sufficient amount of labeled data is required as training corpus to build the classifier in conventional supervised classification methods, which will be helpful to guess the class labels of the unlabeled instances. Conversely, unsupervised learning, only depends on unlabeled instances, and doesn't require class labels to build a classifier so; they attempt to explore the latent composition of unlabeled data to train a model (Zhu, 2005). Unfortunately most of the huge amount of accumulated data on the web is unlabeled. This restrict their usage in numerous machine learning applications like speech recognition, sentiment recognition and text classification. Moreover, assigning labels to them manually is expensive, tedious and time-consuming. Furthermore, to train a classifier with very little labeled data possibly will not yield adequate classification accuracy. Semi-supervised Learning (SSL) algorithms take advantages of both labeled and unlabeled instances to improve the classification performance. A lot of SSL algorithms have been suggested in the former decades, like co-training (Blum and Mitchell, 1998), self-training (Rosenberg, 2005; Yarowsky, 1995), graph-based methods (Zhu, 2005), semi-supervised support vector machines (Zhu, 2005), Estimation-Maximization (EM) with generative mixture models (Nigam et al., 2000), transductive support vector machines (Chapelle and Zien, 2005).

It is known that Latent Semantic Indexing (LSI) utilizes latent higher-order structure between terms and documents (Kontostathis and Pottenger, 2006). Higher-order relations in LSI get "hidden semantics". The LSI algorithm (Deerwester, 1990) is a very popular and commonly-used technique in the fields of text mining and information retrieval. There are several LSI-based classifiers. For instance, in (Zelikovitz and Hirsh, 2004) the authors propose an LSI-based k-Nearest Neighborhood (LSI k-NN) algorithm in a semi-supervised setting for short text classification which is one of the simple uses of LSI in text classification. In this work, the authors use the k-Nearest Neighborhood (k-NN) algorithm that is based on calculating similarities or distance between training instances and a test instance in the transformed LSI space. They set the number of neighbors to 30 and use the noisy-or operator. A similar approach is used in a supervised setting to build an LSI-based k-NN algorithm as one of the baseline algorithms in (Ganiz et al., 2011). In this study, the number of neighbors is set to 25, and the dimension parameter (k) of the LSI algorithm is optimized.

In a recent study (Altınel et al., 2015), a novel supervised semantic smoothing kernel for SVM is offered: Class Meaning Kernel (CMK). CMK uses Helmholtz principle (Balinsky et al., 2010, 2011a, 2011b, 2011c) for smoothening a document's words in BOW demonstration. Evaluation of CMK on experimental data reveals significant improvement in classification accuracy over linear kernel. This is very important since linear kernel is a benchmark algorithm for text classification field.

Inspired by the benefits of CMK over linear kernel, and concentrated on the truth that there is inadequate labeled samples in actual world cases, a non-iterative semi-supervised version of CMK is built, which is named Instance Labeling Based on Meaning (ILBOM). This

algorithm uses meaning values of words. The offered approach uses both labeled and unlabeled data for building a classification model. Initially, it smoothens the words of the labeled instances in BOW representation with the usage of meaning calculation like it is done in CMK (Altınel et al., 2015). Then, it tries to give appropriate labels to unlabeled samples. It achieved this labeling process by meaning calculations. Instances in unlabeled dataset are classified one by one by using an algorithm which is quite similar to maximum likelihood classifiers such as Naïve Bayes (NB). A similar algorithm Supervised Meaning Classifier (SMC) for supervised classification is published in a recent study (Ganiz et al., 2015). Similarly, meaning values of each word in the training set for each class are calculated. This constitutes the training phase. In the classification phase, for an unlabeled instance, meaning values of the terms for a particular class is summed up to obtain class membership value. The class with maximum membership score is chosen as the label of the instance. One of the main differences of the approach employed in this study is to incorporate term frequencies in the class membership calculations. In the experiments, this leads to better results. This smoothing process increases the importance of meaningful words (i.e., significant words) for each class whereas it reduces the importance of general words. This outcome is actually very pleasing since general words are not good at differentiating classes. This methodology decreases drawbacks of BOW feature representation. The main novelty of the suggested algorithm is the utilization of class-based meaning calculations in both smoothing process of the semantic kernel and labeling of unlabeled data. So a hybrid model called ILBOM is suggested. This model combines a slightly advanced version of SMC for the classification of unlabeled instances to in a single pass to significantly extend the original training set and use extended training set to train a SVM with CMK semantic kernel. It should be noted that both of the methods use class based meaning calculations but in a different way. It is observed that ILBOM yields to significant increases in the classification accuracy as a semi-supervised classifier.

The first benefit of the suggested method is its classification capability. To evaluate classification effectiveness of ILBOM several experiments are conducted on varied benchmark datasets. According to experimental results ILBOM has higher classification performance than the baseline algorithms. In linear kernel function, calculated similarity matrix contains information about only the shared terms. This methodology may be treated as first-order approach as its context includes just a single document only as it is mentioned in (Altınel et al., 2015). Nevertheless, ILBOM takes advantages of meaning calculation in the scope of classes. ILBOM utilizes semantic connection between two words which is achieved by class-based meaning values.

Another advantage of ILBOM is its comparatively low complexity as there is no need of an exterior knowledge source like Wikipedia or WordNet. Besides, as a semantic classifier; ILBOM is always up to date since it is built by corpus based statistics. Lastly, ILBOM is a hybrid non-iterative SSL algorithm which is much simpler than usually iterative SSL algorithms. The modified version of SMC is so powerful that it can sufficiently accurately assign class labels to large number of unlabeled instances in a single pass using relatively much smaller amount of training set. And even the huge noise introduced to augmented training set due to the scarcity of the original training set can be compensated by another powerful supervised algorithm, CMK, which uses semantic smoothing kernel to transform highly noisy training instances. This novel combination of two different supervised meaning based classifiers lead to an efficient and effective SSL algorithm.

The other benefit of ILBOM is its flexibility in combining with existing term-based similarity measures or using semantic resources like WordNet or Wikipedia.

The rest of this paper is prepared as followings: A short overview to SVM is given in Section 2. Also, previous semi-supervised methodologies in text classification field and meaning calculation are presented in Section 2. Section 3 details ILBOM. Experiment environment and experiment results are reported in Section 4. Lastly, there is a conclusion with a discussion on possible future directions of ILBOM is in Section 5.

2. Related work

2.1. SVM for text classification

SVM is commonly applied in text classification field as a machine learning algorithm (Boser et al., 1992; Vapnik, 1995). Its basic aim is to find the optimal separating hyperplane between two classes which has the maximal margin.

A kernel function in SVM likes a similarity function, as it computes the similarity scores of data points in the transformed space. The traditional kernel functions for the document vectors d_p and d_q:

$$\text{Linear kernel: } \kappa(d_p, d_q) = d_p d_q \tag{1}$$

$$\text{Polynomial kernel: } \kappa(d_p, d_q) = (d_p d_q + 1)^b,\ b > 1 \tag{2}$$

$$\text{Radial Basis Function (RBF) kernel: } \kappa(d_p, d_q) = \exp(\gamma \|d_p - d_q\|^2) \tag{3}$$

2.2. Semi-supervised learning algorithms

There is a scarcity of labeled training instances in practical real world cases. In these cases, unlabeled instances are utilized by SSL methods for creating better classifiers with higher accuracy. Self-training and co-training are two widely applied SSL algorithms. In self-training; a classification model is formed with labeled instances and then this classification model tries to label the unlabeled instances (Yarowsky, 1995). Then unlabeled instances that the classifier has high classification confidence are given labels and combined with the original labeled instances. Following this, the classifier is re-trained using this extended labeled-instances set. It is very easy to implement self-training algorithm; nevertheless as it is hard to guarantee the convergence of it, the algorithm is repeated up to a well-known iteration number or achieving a convergence standard.

Co-training is similar to self-training with the difference that it accepts that features can be separated into two isolated groups. According to co-training, two distinct classifiers are trained on these two subsets (Blum and Mitchell, 1998). After that, each classifier is tried to give labels to the unlabeled examples. At each iteration, unlabeled samples with the maximum classification confidence are chosen and sent to the labeled data set. Both classifiers are retrained on this extended data set, and the steps are re-performed a pre-defined number of times. As it is discussed in (Jin et al., 2011) the central logic behind co-training is that a classifier may give appropriate labels to some samples while it will be harder for the other classifier.

There are various types of self-training and co-training methodologies in the literature. One type of them uses all the unlabeled examples in each iteration, which means any selection standard is not used (Nigam et al. (2000); Nigam and Ghani (2000). Another kind of self-training and co-training methods uses active learning to choose unlabeled samples and then asks some human experts to label them which produces no mislabeled instances will occur (Muslea et al., 2002; Zhu, 2005). EM algorithm is used in an active learning framework in order to improve SSL in RBF and is applied to content-based image retrieval in Luo and Zhang (2008). A very similar method (with the addition of suitable preprocessing of the data) is described in Jiang (2009) for text classification. Liu et al. (2009) uses uncertainty sampling to choose unlabeled examples in all iterations. Then a cost-sensitive classifier is developed on the expanded labeled data. However active learning techniques are not easy to apply since they require human specialists.

Confidence selection is a common instance selection methodology

(Blum and Mitchell, 1998; Chapelle et al., 2006; Nigam and Ghani, 2000; Yarowsky, 1995). There are also other selection methods in the literature. One of them is the study presented by Wang et al. (2008). Their methodology is an adapted Value Difference Metric and depends on Decision Tree (DT) classifiers. They use the NB algorithm in order to classify sentences as objective or subjective. Their experimental evaluation on several datasets show that their algorithm works well on small datasets. A novel data editing method, named SETRED, is suggested in (Li and Zhou, 2005). SETRED takes advantages of the information of the neighbors of each self-labeled example to distinguish and delete the mislabeled instances. ISBOLD is presented by Guo et al. (2011) as a selection strategy. It is applied to avoid possible performance decrease in both co-training and self-training.

Li et al. (Li et al., 2008) presents a novel algorithm that includes three learners, namely h_1, h_2 and h_3. First of all, three classification models are built with using labeled examples. Then, these classifiers are utilized to give labels to the unlabeled samples; if two of them guess the same label; then that sample will be used to teach the third classifier. This procedure is repeated and the last estimation is completed with a majority vote among all the classification models.

In (Li et al., 2010), a *C4SVM* algorithm is offered. *C4SVM* algorithm is a kind of semi-supervised SVM and uses misclassification costs in its optimization function. In the literature, there are some algorithms in which a single base learner is applied and unlabeled examples are iteratively used depends on their own knowledge. Some systems use EM approach to give posterior parameters of a generative model. NB labels each unlabeled instance by using probability for each class (Nigam et al., 2000). Furthermore, there are systems uses the unlabeled documents to find a better configuration of Bayesian Network (Cohen et al., 2004). There are also some systems use transductive inference for SVM on a special test set (Joachims, 1999). The self-training algorithm (Nigam and Ghani, 2000) is an example of this kind. Nigam and Ghani (2000) use confidence selection in their all iterations.

Transductive Support Vector Machines (TSVMs) is a semi-supervised version of conventional SVM with the difference of usage of unlabeled data. The goal is to give labels to unlabeled instances and take advantages of these data in the training step such that a linear boundary will be exist which has the maximum margin on the labeled data. One of the recent studies is presented in (Li and Zhou, 2011) which is the implementation of semi-supervised support vector machines (S3VMs). Li et al. name their approach as S4VMs and explain that S4VMs tries to make use of many candidate low-density separators in contrast to common S3VMs which typically focus on approaching one optimal low density separator. Their comprehensive experiments validate the effectiveness of S4VMs.

In SSL, the selected unlabeled instances can finally help to build a better classification model. Nevertheless, there are some studies (Cozman, 2003; Guo et al., 2010) presenting that unlabeled samples are fairly often deleterious to the classification performance in various cases. For example, there is an extensive empirical study in (Guo et al., 2010) which was conducted on numerous popular SSL algorithms using different base Bayesian classifiers. They conduct a series of experiments on 26 UCI datasets. According to their experimental results, if the classifier has poor classification performance and erroneously gives labels to unlabeled samples, there will be stored mislabeled instances that causes the last performance will be jeopardized.

Prod. Type: Also there is a recent study (Schwenker and Trentin, 2014) in which several semi-supervised methods and applications are described.

2.3. Meaningfulness calculation and Its application fields

Helmholtz principle from Gestalt theory in image processing states that; “perceived geometric structure is perceptually meaningful if it has a very low probability to seem in noise” (Balinsky et al., 2011a). In other words, humans easily notice events with a large deviation from noise or randomness. There is a simple illustrative example in Fig. 1. In this figure, there is a group of five aligned dots in both Figs. 1(a) and (b); however they cannot be easily perceived in Fig. 1(a) because it contains high noise. The alignment probability of five dots increases due to the high noise, i.e. large number of randomly placed dots. In contrast, the image in the right hand side is the form of the image in the left hand side after removing randomly placed dots. It might be easier to notice the alignment pattern in Fig. 1(b) since it is not likely to happen circumstantially. Balinsky et al. (2011a) state that rapid and unusual modifications will not happen accidentally and they can be directly detected.

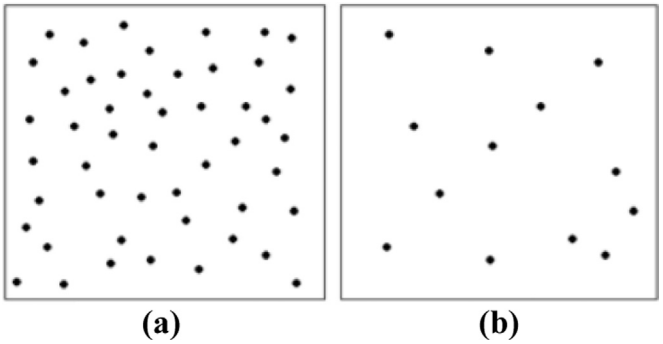

Fig. 1. The Helmholtz principle (Balinsky et al., 2011a) **(a)** A set of five aligned dots with great noise, (i.e. many arbitrarily placed dots). **(b)** A set of five aligned dots with low noise.

Thus, the above explanatory example and other examples in (Balinsky et al., 2011a) show that interesting events and meaningful features appears in large deviations from randomness.

The textual material comprises structures like sentences, paragraphs and documents. Balinsky et al. (2011a) try to define the meaningfulness of these structures by utilizing the Helmholtz principle. A meaning value is given to each word for modeling the meaningfulness of these buildings. According to their new algorithm for meaningful keyword extraction, there are two theories: 1) Keywords should be defined in the context of other documents like it is done in the TF-IDF method. 2) Subjects can be specified by “unusual activity”, so a new theme can be perceived by a harsh increase in the number of occurrences of definite terms. Balinsky et al. (2011a) mention that a sharp rise in frequencies can be used in quick modification discovery. A burst is a period of increased and quick modifications in an event as mentioned in (Kleinberg, 2002).

Depending on the studies given above, new algorithms are implemented for numerous corresponding application fields such as information extraction (Dadachev et al., 2012), document summarization (Balinsky et al., 2011b), rapid change recognition in data streams (Balinsky et al., 2010) and keyword extraction (Balinsky et al., 2011a).

According to the study (Balinsky et al., 2011a), the meaning value of a term w in a class c_j is computed with Eq. (4):

$$meaning(w, c_j) = -\frac{1}{m}\log\binom{k}{m} - [(m-1)\log N] \tag{4}$$

where w denotes a word, m shows the frequency of term w in class c_j, k indicates the frequency of term w in the whole dataset. $N = L / B$; L represents the length of the dataset and B represents the length of the class c_j in terms (Balinsky et al., 2011c). If a word's meaning score in a specific class is larger, then this means that this word is more informative for that class.

The meaning calculations are done in a supervised way, which means that a class of documents can be used as scope for computing meaning values of terms. A meaning value of a word essentially shows how high this word's frequency is likely to be in a class of documents

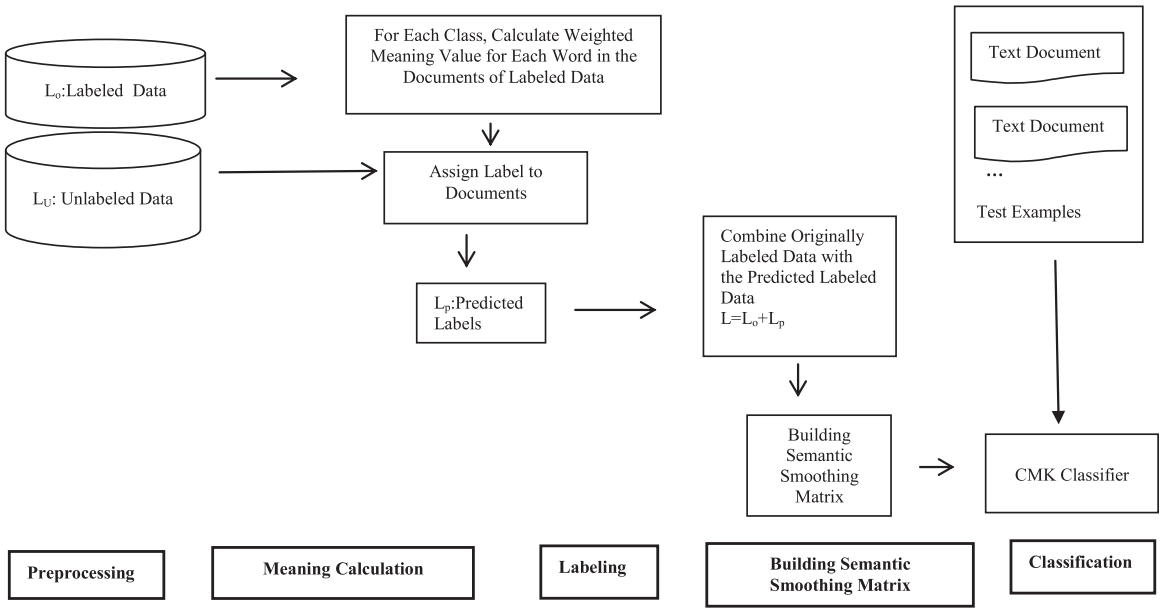

Fig. 2. ILBOM system.

compared to the other classes of documents. The meaning value of this word will be a large score if it is unexpected in a class of documents compared to the other classes of documents. From this perspective, meaning calculation seems to the Multinomial Naïve Bayes (MNB) where all the documents in a class are combined into a single document and then the probabilities are expected from this large class document. Meaning calculation is also similar to Term Frequency-Inverse Class Frequency (TF-ICF) technique which normalizes the term frequencies by the class frequencies.

There are also other studies which uses different semantic properties in order to classify texts. One of them is for rapid change detection (Anagnostopoulos, 2010) which presents a capture–recapture methodology that records the users' browsing behavior during their web search sessions. Their approach is actually based on a meta-search standardization, which is capable of self-adapting over the continuous changes that occur on the web which consequently gives users' navigation attitude. Their experimental results show an important improvement in compare to the existing approaches in the literature (Anagnostopoulos, 2010).

In (Charalampopoulos and Anagnostopoulos, 2011), there is a study which compares WEKA (Hall et al., 2009) clustering/classification algorithms for web page classification. Firstly, different web page classification algorithms (SVM, Latent Semantic Indexing, Page Brin Indexing, Classification using URL, Structure Based Classification, Classification using Link Information, Adaptive Classification with Multiple-Classification Ripple-Down Rules, Classification using Marginal Fisher Analysis and Minimax Probability Machine) are summarized/discussed including their advantages and disadvantages. Secondly, the authors attempt to contribute 4 unique characteristics of web pages such as text, graphics, links and references to the classification process. Then, they propose a method which cares text as a content and uses references in order to classify texts. They use k-NN, DT, neural network and Expectation Maximization (EM). The authors in (Charalampopoulos and Anagnostopoulos, 2011) get many useful conclusions from their experiments. One of those conclusions is that k-NN algorithm is not as good as the other classifiers in assigning labels to instances correctly for web page classification. They mention that no model is universal and every situation needs to be addressed separately. They also state that in web page classification, data must be modeled in relation to the processing algorithm, but evaluated alongside every other suggested approach.

In another recent study (Razis et al., 2016), an iterative algorithm is proposed towards the automatic labeling of Twitter accounts according to thematic categories derived from DBpedia properties. The authors present the motivation behind the selection of these thematic categories and discuss their evaluation. Then they suggest two generic and adaptable approach, namely URI Construction, for discovering the thematic description of Twitter accounts. URI Construction methodology is offered which requires the pattern recognition of the URI policy of each data source. One of the results of their analysis is that a generic search considering specific attributes (e.g. Display Name) is not sufficient. According to their experimental results, their approach results in much higher precision and true positives while resulting in lower number of false ones.

3. Instance Labeling Based on Meaning (ILBOM)

In text classification field, linear kernel has a widely usage as a kernel function in SVM. This is fairly because of the truth that the BOW demonstration of documents is quite high dimensional with thousands of features. Consequently, since linear kernel is the simplest kernel (i.e., it is based on only the shared terms between documents) that can be used in SVM, it is suitable for the high dimensionality of text classification. However, as it is discussed in (Altınel et al., 2015) this may cause an important problem since it really disregards the latent connections between the words in documents. Moreover, it will be very difficult to discover consistent patterns between instances when training set is inadequate since in that situation the feature vectors will be fairly sparse. All these reasons and examples conclude that using just inner product to calculate similarity between documents will not always generate adequately accurate similarity scores between these documents. Thus, to reach more accurate classification performance it is necessary to stress on core terms, those are strictly associated to the topic of that class (Steinbach et al., 2000). Therefore, with the motivation of handling these problems, researchers have been showing interest in semantic smoothing kernels which concentrate on the semantic relationships between words (Bloehdorn et al., 2006; Mavroeidis et al., 2005; Siolas and d'Alché-Buc, 2000).

In this article, a new semi-supervised semantic smoothing kernel is introduced. ILBOM comprises five modules as shown in Fig. 2 namely preprocessing, meaning calculation, labeling, building semantic smoothing matrix and classification.

3.1. Preprocessing

In this step stemming and stopword filtering is applied on the documents of the corpus. Additionally, infrequent terms whose frequency is less than three also filtered and removed. Moreover, attribute selection is applied and the most informative 2000 words are selected by Information Gain. These preprocessing parameters are optimized after a series of experiments in our previous studies (Altınel et al., 2014a, 2014b, 2015; Ganiz et al., 2009, 2011, 2015) and applied in this study. It is observed that this preprocessing improves the performance of the classifier since it reduces the noise; therefore, it is equally performed in all experiments that are reported in Section 4.

3.2. Meaning calculation

In this step, meaning values of the words are calculated as in Eq. (4) by using only the documents in the labeled set. This calculation produces $M_{labeled}$ class-based-term-meaning-matrix which demonstrates the meaningfulness of the terms for the labeled documents for each class. If a word's frequency in a class is one then its meaning score for that class is zero by Eq. (4). If the frequency of a term in a class is zero, its meaning value for that class is minus infinity after all the computations in Eq. (4). On account of making calculations more practical Altınel et al. (2015) give the following smallest score to that term based on the range of meaning scores for all of the terms in the dataset. Then, M is generated as a term-by-class matrix that contains the meaning scores of words in all classes of the dataset. Altınel et al. (2015) notice that these meaning scores are large for those terms that let them to differentiate between classes. Actually semantically correlated words of that class, i.e. “core” terms like it is stated in (Steinbach et al., 2000), gain significance while semantically isolated terms, i.e. “general” terms lose their significance. Consequently words are ranked according to their significance. For example, if the term “data” is highly present while the terms “knowledge” and “information” are less, semantic smoothing will increase the scores of the last two words since “data”, “knowledge” and “information” are powerfully correlated concepts. This new encoding enriches the document representation in compare to the customary TF-IDF representation as extra statistical information that is straightly computed from the training documents is added into the kernel.

ILBOM uses both labeled and unlabeled data. On account of including unlabeled samples into the classifier model in SVM, first it is required to assign labels to unlabeled documents. On account of this, weighted total meaning value of a document is calculated by using Eq. (5):

$$TM(d_i, c_j) = \sum_{n=1}^{t} meaning(w_n, c_j) \times tfw_{n,d_i} \quad (5)$$

where w shows a word, c indicates a class, $meaning(w_n, c_j)$ represents the meaning value of word w_n in a class c_j, $\sum_{n=1}^{t} meaning(w_n, c_j)$ shows the total meaning value of a document for class c_j, tfw_{n,d_i} shows the term frequency of word w_n in document d_i. The meaning calculation part of Eq. (5) is exactly done like it is presented in Eq. (4). The motivation behind Eq. (4) is based on Helmholtz principle and utilized in (Balinsky et al., 2010, 2011a, 2011b, 2011c) as mentioned in Section 2. This meaning calculation is also performed in (Altınel et al., 2015) to build semantic smoothing kernel for SMO and in (Ganiz et al., 2015) to label test instances.

The pseudocode of meaning calculation module of ILBOM is given in Fig. 3.

3.3. Labeling

$TM(d_i, c_j)$ matrix, represented in Fig. 4(a), includes $d_{ik} = [d_{i1},\ldots,d_{ik}]$ document vectors having the document d_i's total meaning scores for the all classes, respectively. In this labeling step, another column is added into this matrix as represented in Fig. 4(b). This new column is named as C_{max} and signifies the class number with the greatest score in $d_{ik} = [d_{i1}\ldots d_{ik}]$ document vector.

At the end of this labeling-step, suitable labels are given to all the unlabeled documents. Consequently, the enlarged labeled set is:

$$L = L_o + L_p \quad (6)$$

where L_o shows the original labeled instances, L_p signifies the instances which are labeled at this step and L is the total of L_o and L_p; respectively.

In labeling step, unlabeled instances are assigned labels in order to be added into labeled instances. This labeling procedure is actually done with the help of meaning calculation like it is applied in SMC (Ganiz et al., 2015). The pseudocode of labeling module of ILBOM is given in Fig. 5.

3.4. Building semantic smoothing matrix

At the beginning of this step L is ready as the composition of both L_o and L_p. A semantic smoothing matrix is formed by using L as in Eq. (7):

$$S = MM^T \quad (7)$$

where M matrix represents the meaningfulness of the terms in each class and S matrix shows the semantic relatedness between words. In other words; S matrix is calculated as semantic proximity matrix.

The pseudocode of this module of ILBOM is given in Fig. 6.

3.5. Classification

In classification step; the supervised CMK, which is proposed in a previous study (Altınel et al., 2015), is run for labeling test instances. In order to run CMK; S is used. S is generated with Eq. (7) in the step of building semantic smoothing matrix, as a semantic proximity matrix. Utilizing S matrix, CMK classifies the test instances with the help of kernel function shown in Eq. (8). A kernel function actually produces similarity values between instances. Thus, mathematically, the kernel or similarity score between two documents is calculated as:

$$k_{ILBOM}(d_1, d_2) = d_1 S S^T d_2^T \quad (8)$$

The S matrix in Eq. (8) modifies the orthogonality of the VSM, since this mapping leads term dependence (Wittek and Tan, 2009). After eliminating orthogonality, documents can be categorized as similar even though they do not have any common words.

4. Experiments

4.1. Datasets

Evaluation of ILBOM is done by performing a chain of experiments on numerous standard textual datasets. The first dataset is IMDB[1] which has 2000 reviews with two labels (i.e. *negative and positive*) about many movies in IMDB. The labels are balanced in labeled/unlabeled/test splits. Other two datasets are variants of well-known 20 Newsgroup[2] dataset. 20 Newsgroup dataset contains 20,000 newsgroup documents under 20 different category. Two subgroups, namely “POLITICS” and “SCIENCE”, are used from the 20 Newsgroup dataset. Each class in 20 Newsgroup dataset has the same number of documents. The fourth dataset is the Mini-Newsgroup [3] dataset. There are 20 classes and each class has 100 documents in Mini-Newsgroup dataset. Mini-Newsgroup dataset is also another subset of the 20 Newsgroup dataset. Table 1 shows the properties of all the datasets.

Module Meaning Calculation // Calculate the meaning value of the word *w* in class *j*

Input

$\overline{L_o}$: labeled documents set

Output

$\overline{M}_{w,j}$: matrix shows the meaning values of words for all classes(*w* ,*j*th element shows the meaning value of the word *w* in class *j)*

Local variables

$m_{w,j}$: total frequency of word *w* in class *j*

$k_{w,d}$: total frequency of word *w* in L

N : the length of the corpus / the length of the class

begin

for each word *w*

for each class *j*

$$\overline{M}_{w,j} = -\frac{1}{m}\log\binom{k}{m}\frac{1}{N^{m-1}}$$ //Calculate the meaning value of the word *w* in class *j*

end for

end for

end

Fig. 3. The pseudocode of meaning calculation module of ILBOM.

4.2. Experiment Setting and Evaluation

In order to see the behavior of ILBOM for each dataset, 1%, 2%, 3%, 4%, 5%, 7%, 10%, 15% and 30% data are separated as labeled set while 79%, 78%, 77%, 76%, 75%, 73%, 70%, 65% and 50% of the data are used as unlabeled set. The remaining 20% of the dataset is used as test portion.

SMO's misclassification cost (*C*) parameter is used as 1. 10 random runs are performed for each training set level by arbitrarily choosing the documents to form training set and their average score is reported. Standard deviations are also provided in the results tables. The performance gain between baseline algorithm and ILBOM is calculated as:

$$Gain_{ILBOM} = \frac{(P_{ILBOM} - P_x)}{P_x} \quad (9)$$

where P_{ILBOM} is the classification accuracy of *ILBOM* semantic smoothing kernel and P_x shows the classification accuracy of the SSL-Linear. The experimental results are reported in Tables 2–7. The first three columns in these results tables demonstrate the labeled, unlabeled and test splits in the experiments. The “Linear” columns in Tables 2–7 represent the classification accuracy of the first baseline algorithm which is SMO under linear kernel. The “SSL-Linear” column in Tables 2–7 represents the classification accuracy of the second baseline algorithm which is SSL-Linear kernel. The “CMK” column in Tables 2–7 represents the classification accuracy of the third baseline algorithm which is our previous work, CMK (Altınel et al., 2015). The “ILBOM” column in Tables 2–7 represents the classification accuracy of the proposed algorithm, ILBOM. The “Gain” column in Tables 2–7 demonstrates the (%) gain of ILBOM over SSL-Linear algorithm computed as in Eq. (9). Furthermore, for statistical significance Students *t*-Tests with significance level $\alpha=0.05$ are given. In the training sets, where ILBOM significantly differs from baseline algorithm according to Students *t*-Tests, this is indicated with “*”. ILBOM is integrated into the WEKA library.

4.3. Baseline algorithms

First baseline algorithm which is used to evaluate the results of ILBOM is the customary linear kernel. The experimental results of linear kernel on results tables are generated by building linear kernel classifier with using just labeled split and then running this classifier to classify test instances in the test split, so; the unlabeled split is not used for the experiments of linear kernel. Secondly, SSL-Linear is used as the baseline algorithm. SSL-Linear first classifies unlabeled documents with the help of linear kernel which is built by the labeled documents. Formerly, SSL-Linear merges these unlabeled documents with their assigned-labels and original labeled documents to construct the classifier again by using linear kernel. After that, it again tries to classify unlabeled documents with the last built model and compares the labels of each document. If a document is classified into a distinct class by the second classifier then its label is modified since the last model is more extended in compare to the first model. This self-training process continues until it reaches 100 iterations. Moreover, the results of ILBOM are compared to those of CMK, which is the third baseline. The experimental results of CMK on Tables 2–7 are produced by building CMK classifier (Altınel et al., 2015) with using only labeled split and then running this classifier to classify test instances in the test split. So; again the unlabeled split is not used for the experiments of CMK like linear kernel.

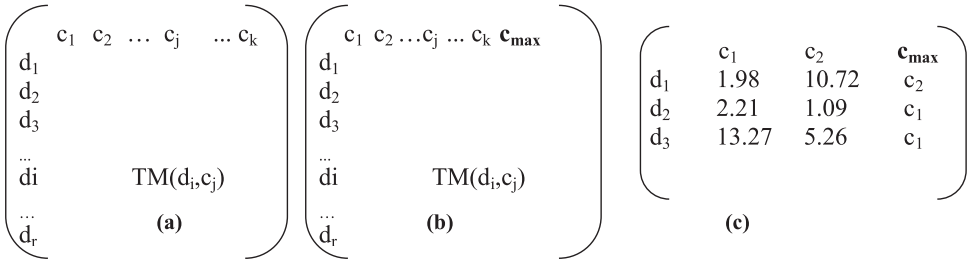

Fig. 4. (a) *TM (d_i, c_j)* matrix indicates the meaning-scores-sum of the words in the document d_i for the class c_j. **(b)** New column, c_{max}, shows the class number which the document has maximum meaning-score-sum of that document. **(c)**A simple demonstrative matrix represents the meaning-scores-sum and the related class labels for the documents of d_1, d_2 and d_3.

Module Labeling *//Labeling* Unlabeled Documents

Input

$\overline{L_u}$: Unlabeled documents set

$\overline{M}_{w,j}$: matrix shows the meaning values of words for all classes in $\overline{L_o}$

Output

$\overline{L_p}$: Matrix shows the unlabeled documents with their predicted labels

Local variables

$\overline{TM}_{d_i,c_j}$: Matrix shows the unlabeled documents with their total meaning values for all classes (i,j^{th} element shows the total meaning value of the document d_i in class c_j)

begin

$\overline{TM}_{d_i,c_j} = \overline{L_u}_{w,j} (\overline{M}_{w,j})$

for each document d_i in the $\overline{TM}_{d_i,c_j}$

L_{P_i} = label(max($\overrightarrow{TM_{d_i}}$)) *//Assigning the label of document d_i*

end for

end

Fig. 5. The pseudocode of labeling module of ILBOM.

4.4. Experimental results and discussion

ILBOM is superior to linear, SSL-Linear and CMK on 20NewsGroup-Science dataset which can be perceived from Table 2. The classification performance difference is definitely noticeable at smaller labeled set levels between 1% labeled and 15% labeled set percentage. For instance; the accuracies of ILBOM are 59.7%, 64.15%, 79.78%, 86.03% and 89.6% at labeled set levels 1%, 2%, 4%, 5% and 7%. However; at those labeled set levels, the classification accuracies of SSL-Linear are 50.03%, 57.1%, 66.45, 67.95% and 71.7%. ILBOM also outperforms previously suggested supervised semantic kernel, CMK, at all of the labeled set levels with the exception of labeled set level 30%. The maximum gain of ILBOM over SSL-Linear kernel on 20NewsGroup-Science dataset is 26.61% which is achieved at 5%

Table 1
Properties of datasets.

Dataset	#classes	#documents	#words
IMDB	2	2000	16,678
20NewsGroup-Politics	3	1500	2477
20NewsGroup-Science	4	2000	2225
Mini-NewsGroup	20	2000	12,112

1 http://www.imdb.com/interfaces
2 http:// www.cs.cmu.edu/~textlearning
3 http://archive.ics.uci.edu/ml/

Module Building Semantic Smoothing Matrix *//Building semantic smoothing kernel*

Input

$\overline{L}$: $\overline{L_o} + \overline{L_p}$ (All the labeled documents)

$\overline{M}_{w,j}$: matrix shows the meaning values of words for all classes(w,j^{th} element shows the meaning value of the word w in class j)

Output

$\overline{G}_{d_p,d_q}$: Gram matrix shows the kernel value between documents d_p and d_q

Local variables

$\overline{S}_{w_p,w_q}$: Semantic smoothing matrix shows the relatedness between words w_p and w_q

begin

$\overline{S} = \overline{M}\ \overline{M}^T$ *//Building semantic smoothing matrix*

for each document d_p in $\overline{L}$

for each document d_q in the training set

$\overline{G}_{d_p,d_q} = d_p S S^T d_q$ *//Calculating the kernel value between documents d_p and d_q*

end for

end for

end

Fig. 6. The pseudocode of building semantic smoothing kernel module of ILBOM.

Table 2
Experiment results of algorithms on 20NewsGroup-Science dataset.

Labeled Dat%	Unlabeled dat%	Test dat%	Baseline Algorithms			ILBOM	Gain of ILBOM over SSL-linear
			Linear	SSL-linear	CMK		
1	79	20	51.80 ± 5.33	50.03 ± 5.29	39.42 ± 6.78	**59.70 ± 21.63**	19.33*
2	78	20	59.10 ± 5.49	57.10 ± 6.01	50.30 ± 6.00	**64.15 ± 15.2**	12.35*
3	77	20	**66.03 ± 3.61**	64.83 ± 3.64	53.40 ± 7.78	63.93 ± 12.25	−1.39
4	76	20	69.05 ± 3.70	66.45 ± 3.59	60.50 ± 7.17	**79.78 ± 5.62**	20.06*
5	75	20	70.10 ± 4.34	67.95 ± 4.64	70.03 ± 5.07	**86.03 ± 3.77**	26.61*
7	73	20	72.72 ± 4.47	71.70 ± 3.59	78.53 ± 5.07	**89.60 ± 3.01**	24.97*
10	70	20	76.68 ± 2.07	74.58 ± 3.30	87.48 ± 4.81	**92.55 ± 1.23**	24.09*
15	65	20	83.53 ± 2.68	80.53 ± 2.79	89.95 ± 1.71	**94.38 ± 0.91**	17.20*
30	50	20	86.28 ± 2.27	83.70 ± 1.97	**95.28 ± 0.95**	94.98 ± 0.78	13.48*

labeled set level. This is of great significance as generally it is hard and costly to collect labeled documents in actual world scenarios. Also it must be noted that, ILBOM outperforms linear kernel at labeled set levels 1%, 2%, 4%, 5%, 7%, 10%, 15% and 30%.

Furthermore, the total kernel computation time of linear kernel, SSL-Linear, CMK and ILBOM are recorded in terms of seconds. All the experiments reported here are performed on an experimental environment which includes our experiment server, Turkuaz. Turkuaz uses WEKA library and it has two Intel(R) Xeon(R) CPUs at 2.66 GHz with 64 GB of memory. The computation time of linear kernel, SSL-Linear, CMK and ILBOM on 20NewsGroup-Science dataset is shown in Fig. 7. According to the experiment records, the computation time of linear kernel, SSL-Linear, CMK and ILBOM are 220, 1051, 1830 and 5490 s. This computational difference between ILBOM and CMK is appreciated as the labeling process of unlabeled data in ILBOM. Actually this might be one of the possible future tasks in order to improve ILBOM.

Additionally, the experiments for 20NewsGroup-Science dataset are repeated by performing the classical 10-fold Cross Validation (CV) technique, where in each different fold the labeled/unlabeled data population can be mixed, like it is done in the current version of ILBOM. The experimental results for 20NewsGroup-Science dataset with 10-fold CV technique are reported in Table 3. Although the classification results for the labeled set levels between 1% and 30% shows differences in compare to the random selection technique; the superiority of ILBOM over all of the baseline kernels still continues. According to Table 3, ILBOM outperforms to all three baseline kernels, namely linear kernel, SSL-Linear and CMK with the only exception of labeled set level 30%. Even though CMK gives higher classification accuracy than ILBOM at labeled set level 30%, ILBOM is still better than linear kernel and SSL-linear at this labeled set level. The highest gain of ILBOM over SSL-Linear kernel on 20NewsGroup-Science dataset with 10-fold CV technique is 26.33% which is achieved at 5% labeled set level.

According to Table 4, ILBOM's performance is greater than SSL-Linear's performance in all labeled set levels on 20NewsGroup-Politics dataset. Besides, ILBOM performs better than the other baseline kernels in all labeled set levels.

Moreover, as it is done for 20NewsGroup-Science dataset, the experiments for 20NewsGroup-Politics dataset are repeated by performing the classical 10-fold CV technique. The experimental results for 20NewsGroup-Politics dataset with 10-fold CV technique are reported in Table 5. Although the classification results for the labeled set levels between 1% and 30% shows differences in compare to the random selection technique; the superiority of ILBOM over all of the baseline kernels still continues. According to Table 5, ILBOM outperforms to all three baseline kernels, namely linear kernel, SSL-Linear and CMK. The highest gain of ILBOM over SSL-Linear kernel on 20NewsGroup-Science dataset with 10-fold CV technique is 27.36% which is achieved at 3% labeled set level.

According to the experimental results in Table 6, ILBOM is superior to all of the baseline algorithms on the IMDB dataset. For instance; the accuracies of ILBOM are 82.88%, 79.25%, 87.08% and 88.70% at labeled set levels 2%, 3%, 10% and 15% while the accuracies of SSL-linear are only 70.13.7%, 71.15%, 78.83% and 80.35%.

Table 7 also lists the experiment results on Mini-NewsGroup dataset. In overall, SSL-Linear is not as good as its supervised version (linear kernel) on all of the datasets in the experiments. In other words, interestingly, SSL-Linear cannot benefit from the unlabeled examples on 20NewsGroup-Science, 20NewsGroup-Politics, IMDB and Mini-NewsGroup datasets. On the other hand, ILBOM has the capability to benefit from unlabeled instances as perceived from the experimental results in Tables 2–7 since ILBOM outperforms to its supervised version, CMK, at almost all of the test cases. One possible explanation is that ILBOM takes advantages of meaning calculation to label unlabeled instances which is a good idea to capture enough latent semantics between documents and terms (Altınel et al., 2015) especially at low labeled percentages. Another point should be noticed is that gains of ILBOM over SSL-Linear on IMDB dataset are not as much as the gains on other datasets. This may be because of the relatively fewer number of classes in this dataset. However, meaning calculation seems to be more successful at larger number of classes according to the experimental results of CMK (Altınel et al., 2015). Furthermore,

Table 3
Experiment results of algorithms on 20NewsGroup-Science dataset (with 10-fold CV).

Labeled dat%	Unlabeled dat%	Test Dat%	Baseline algorithms			ILBOM	Gain of ILBOM over SSL-linear
			Linear	SSL-linear	CMK		
1	79	20	51.8 ± 5.33	50.58 ± 5.41	39.42 ± 6.78	**54.63 ± 23.4**	8.01*
2	78	20	61.93 ± 2	58.8 ± 3.36	47.98 ± 6.31	**64.68 ± 15.7**	10*
3	77	20	66.72 ± 4.76	64.33 ± 5.11	54.6 ± 7.86	**73.1 ± 11.61**	13.63*
4	76	20	70.22 ± 4.96	66.35 ± 5.08	64.55 ± 7.61	**83.08 ± 6.09**	25.21*
5	75	20	70.1 ± 4.34	68.1 ± 4.96	70.03 ± 5.07	**86.03 ± 3.77**	26.33*
7	73	20	74.9 ± 3.79	72.8 ± 3.73	80.28 ± 4.63	**89.6 ± 2.99**	23.08*
10	70	20	76.68 ± 2.07	75.22 ± 3.58	87.48 ± 4.81	**92.55 ± 1.23**	23.04*
15	65	20	81.1 ± 2.07	79.4 ± 2.59	90.88 ± 1.94	**94.23 ± 1.42**	18.68*
30	50	20	86.28 ± 2.27	84.13 ± 1.99	**95.28 ± 0.95**	94.98 ± 0.78	12.90*

Table 4
Experiment results of algorithms on 20NewsGroup-Politics dataset.

Labeled dat%	Unlabeled dat%	Test Dat%	Baseline algorithms			ILBOM	Gain of ILBOM over SSL-linear
			Linear	SSL-linear	CMK		
1	79	20	52.60 ± 5.69	51.33 ± 6.18	38.60 ± 2.26	**52.63 ± 7.76**	2.53
2	78	20	64.60 ± 6.34	62.20 ± 5.80	47.97 ± 4.64	**82.30 ± 7.78**	32.32*
3	77	20	69.60 ± 5.28	68.97 ± 5.89	64.60 ± 10.43	**86.37 ± 5.43**	25.23*
4	76	20	69.97 ± 5.68	69.07 ± 7.29	68.57 ± 12.7	**88.37 ± 5.05**	27.94*
5	75	20	73.23 ± 3.92	72.23 ± 3.29	78.03 ± 4.46	**88.70 ± 2.80**	22.80*
7	73	20	78.33 ± 5.30	76.87 ± 4.85	80.03 ± 5.24	**92.23 ± 2.26**	19.98*
10	70	20	82.00 ± 2.38	80.77 ± 1.60	87.13 ± 2.13	**93.40 ± 1.28**	15.64*
15	65	20	84.67 ± 4.92	83.93 ± 4.88	91.50 ± 1.69	**94.63 ± 1.62**	12.75*
30	50	20	90.07 ± 1.91	87.50 ± 2.64	94.43 ± 1.05	**95.33 ± 0.82**	8.95*

according to Table 7, ILBOM's gains over SSL-Linear at low labeled percentages are not as much as the gains at higher labeled percentages on Mini-NewsGroup dataset. This maybe because of the relatively smaller amount of labeled instances per class in this dataset. For instance there are 500 instances per class in 20-NewsGroup datasets such as 20NewsGroup-Science and 1000 instances per class in IMDB but there are only 100 instances per class in Mini-NewsGroup. Therefore this yields more misclassified unlabeled examples before building the model and those mislabeled examples degrade the classification performance. Those hidden relations may be very important; because the number of classes is fairly high and the number of documents per class is far smaller which generates high sparsity.

On 20NewsGroup-Science dataset, the only exception that ILBOM does not outperform SSL-Linear is labeled set level 3%. The classification performance of ILBOM is 63.93% while the classification performance of SSL-Linear is 64.83% and the classification performance of linear is 66.03% at labeled set level 3% on 20NewsGroup-Science dataset as it can be observed from Table 2. A similar exception could be observed on Mini-NewsGroup dataset at labeled set level 2%. The classification performance of ILBOM is 29.23% while the classification performance of SSL-Linear is 30.58% and the classification performance of linear is 38.42% at labeled set level 2% on Mini-NewsGroup dataset as it can be seen from Table 7. These exceptions might be explained with the possibility that the documents randomly selected at this labeled set level share a larger number of common terms since linear kernel is based on the dot product of common terms between documents. On the other hand, the meaning score calculation generates less meaning values for the terms which occur in most of the documents from different classes like stop words. However this test case shows that there might be some exceptional cases where terms do not need to be filtered as stop words. As a future direction; we would like to improve our approach so that it generates large meaning scores for important terms which occur in most of the documents from different classes but maybe with different senses; so they do not have to be handled as stop words.

The classification accuracies of ILBOM are also compared to those of SMC and CMK as shown in Fig. 8. The classification accuracies of ILBOM, SMC and CMK at training splits between 1% and 30% on IMDB dataset are represented on Fig. 8. The experimental results show that ILBOM outperforms SMC and CMK at all training set levels on IMDB dataset. The difference is especially perceptible at all training set levels as it can be easily perceived from Fig. 8. For instance, according to Fig. 8 the performance gains of ILBOM over SMC are 22.39%, 26.88% and 25.72% at training set levels 5%, 10% and 30%; respectively. Both SMC and CMK are supervised algorithms which utilize only labeled data. On the other hand ILBOM is a semi-supervised algorithm that utilizes labeled and unlabeled samples together. The experimental results clearly show the superiority of ILBOM to SMC on IMDB dataset. This proves that ILBOM obviously benefits from unlabeled data.

5. Conclusions and future directions

In this article, a novel hybrid semi-supervised classification methodology is suggested for text classification which is much simpler than traditional iterative SSL algorithms. In the first phase, a relatively much smaller amount of labeled data is used as training set in order to give class labels to a large number of unlabeled data. The modified version of SMC is a powerful classifier that it can accurately assign class labels to a great number of unlabeled instances in a single pass using comparatively much smaller amount of training set. And even the large amount of noise introduced to augmented training set due to the scarcity of the original training set can be compensated by another powerful supervised algorithm, CMK, which uses semantic smoothing kernel to transform highly noisy training instances. This novel combination of two different supervised, meaning based classifiers lead to an efficient and effective SSL algorithm for text classification. These two different classifiers are also corpus-based classifiers which are also called language-Independent systems since they are independent from any knowledge source such as WordNet, Wikipedia like knowledge-based systems. Moreover; because ILBOM is composed from corpus-based systems, it does not need the processing of a huge exterior knowledge source like a natural language processor that creates

Table 5
Experiment results of algorithms on 20NewsGroup-Politics dataset (with 10-fold CV).

Labeled dat%	Unlabeled dat%	Test Dat%	Baseline algorithms			ILBOM	Gain of ILBOM over SSL-linear
			Linear	SSL-linear	CMK		
1	79	20	56.97 ± 4.46	54.47 ± 4.46	42.9 ± 5.42	**62.67 ± 12.28**	15.05*
2	78	20	64.37 ± 6.53	64.2 ± 5.96	51.93 ± 6.95	**80.57 ± 7.5**	25.50*
3	77	20	67.07 ± 5.58	64.83 ± 5.52	60.47 ± 4.07	**82.57 ± 5.12**	27.36*
4	76	20	70.47 ± 4.28	69.23 ± 3.5	67.4 ± 5.76	**86.2 ± 5.25**	24.51*
5	75	20	73.23 ± 3.92	72.3 ± 3.29	78.03 ± 4.46	**88.7 ± 2.8**	22.68*
7	73	20	78.07 ± 3.86	76.03 ± 3.93	80.7 ± 4.93	**92.57 ± 2.33**	21.75*
10	70	20	82 ± 2.38	81.1 ± 2.09	87.13 ± 2.13	**93.4 ± 1.28**	15.17*
15	65	20	85 ± 3.29	82.77 ± 2.62	90.47 ± 2.08	**94.73 ± 1.59**	14.45*
30	50	20	90.07 ± 1.91	87.73 ± 2.47	94.43 ± 1.05	**95.33 ± 0.82**	8.66*

Table 6
Experiment results of algorithms on IMDB dataset.

Labeled dat%	Unlabeled dat%	Test Dat%	Baseline algorithms			ILBOM	Gain of ILBOM over SSL-linear
			Linear	SSL-linear	CMK		
1	79	20	65.75 ± 7.79	65.60 ± 8.37	61.33 ± 6.4	**70.43 ± 15.32**	7.36*
2	78	20	69.30 ± 3.28	70.13 ± 2.82	67.97 ± 6.57	**82.88 ± 6.52**	18.18*
3	77	20	72.80 ± 3.28	71.15 ± 3.83	74.25 ± 3.91	**79.25 ± 6.78**	11.38*
4	76	20	76.28 ± 2.27	75.70 ± 2.92	77.03 ± 3.57	**80.08 ± 6.56**	5.79*
5	75	20	77.03 ± 2.55	75.88 ± 2.78	80.33 ± 3.79	**82.68 ± 4.99**	8.96*
7	73	20	79.45 ± 1.76	78.53 ± 2.49	82.38 ± 1.88	**86.33 ± 1.84**	9.93*
10	70	20	79.70 ± 2.86	78.83 ± 3.28	84.65 ± 1.94	**87.08 ± 1.99**	10.47*
15	65	20	81.72 ± 2.47	80.35 ± 1.42	86.98 ± 1.57	**88.70 ± 2.18**	10.39*
30	50	20	85.80 ± 1.37	84.78 ± 1.29	90.88 ± 1.28	**91.40 ± 0.76**	7.81*

grammatical tags such as POS tags based on syntactic analysis. Besides, since corpus-based systems are generated from corpus-based statistics they are always updated. Finally, ILBOM is a corpus-based classifier and there is no coverage problem. This is mainly caused by the fact that the semantic relations between words are particular to the field of the dataset.

The experiment results show that ILBOM successfully benefits from unlabeled documents to advance the classification accuracy. The maximum gain of ILBOM over SSL-Linear kernel is achieved on Mini-NewsGroup dataset which has 20 classes. ILBOM reached this highest gain at 10% labeled set level and 70% unlabeled set level on Mini-NewsGroup dataset.

The proposed SSL method can be considered as a hybrid algorithm. It basically combines two previously published methods. These methods and their parameters are determined after extensive experimentation. Especially the order we apply these methods has a significant effect on the performance. These are a slightly modified version of SMC and CMK, respectively. As it is common in the SSL literature, the initial labeled dataset is chosen as a very small percentage of the benchmark datasets such as 1% or 2%. The SMC type algorithm we employ has a great performance and unique ability to learn a reasonable classifier model from very small training sets as can be noticed from the experimental results of SMC. As a result it is chosen for initial labeling of unlabeled data to extend training set. The CMK, our kernel based approach does not work as effective as SMC in this initial labeling stage. Although, SMC works very well with very small training sets compare to the many major text classification algorithms, it still introduces lots of noise to the extended training set. We haven't applied common practices in SSL such as quality check for initial labeling, selecting only the most confident classifications, or iterative improvement of initial labeling for reducing this noise. The main reason for this is the ability of CMK to handle highly noisy training data. Additionally, we would like to keep ILBOM as simple as possible to implement, compare and improve on following studies. Once again, we could have used common practices in SSL such as iterative improvement of labeling or an ensemble approach but our experimental results persuaded us that this straightforward combination produced good results as a novel semi-supervised text classification algorithm. In any way, we are planning to include these kind of improvements in ILBOM as a future work.

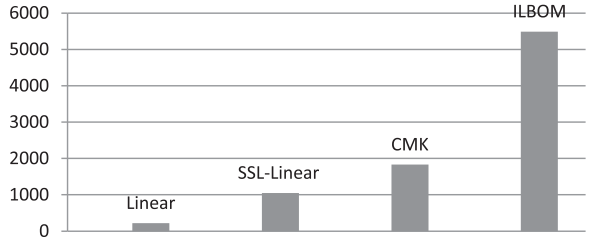

Fig. 7. Time unit of ILBOM and CMK.

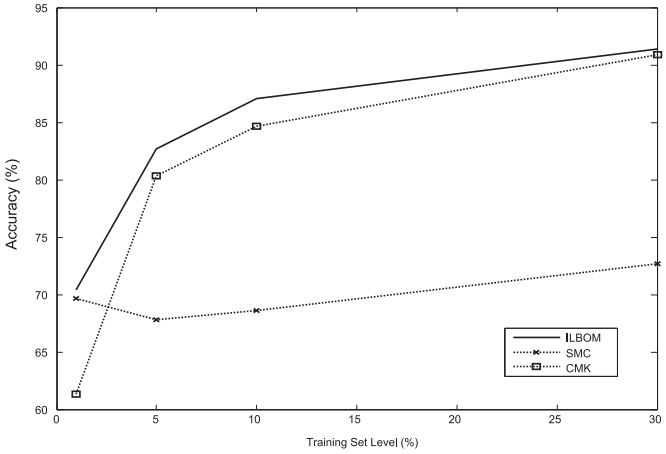

Fig. 8. Classification accuracies of ILBOM, SMC and CMK at different training splits on IMDB dataset.

We also would like to analyze the differences of the two algorithms we use. Both SMC and CMK are based on meaning calculations but

Table 7
Experiment results of algorithms on Mini-NewsGroup dataset.

Labeled dat%	Unlabeled dat%	Test Dat%	Baseline algorithms				Gain of ILBOM over SSL-linear
			Linear	SSL-linear	CMK	ILBOM	
1	79	20	**36.70 ± 4.24**	29.85 ± 4.04	19.23 ± 2.59	32.45 ± 8.83	8.71*
2	78	20	**38.42 ± 5.47**	30.58 ± 4.61	18.43 ± 2.48	29.23 ± 5.45	−4.41
3	77	20	**46.33 ± 4.32**	38.23 ± 4.71	26.63 ± 3.43	41.90 ± 4.89	9.60*
4	76	20	**46.70 ± 8.34**	38.78 ± 7.55	33.48 ± 4.19	45.63 ± 3.51	17.66*
5	75	20	**50.00 ± 5.49**	40.85 ± 5.63	35.78 ± 3.15	49.75 ± 4.17	21.79*
7	73	20	55.15 ± 4.72	47.70 ± 5.62	47.23 ± 3.18	**59.75 ± 3.88**	25.26*
10	70	20	57.03 ± 2.79	47.98 ± 3.37	53.65 ± 3.27	**64.03 ± 2.16**	33.45*
15	65	20	62.48 ± 4.28	55.48 ± 3.48	61.83 ± 3.24	**67.65 ± 2.85**	21.94*
30	50	20	69.55 ± 4.50	63.60 ± 4.30	70.68 ± 3.38	**72.88 ± 2.40**	14.59*

meaning scores of terms are used in very different ways. In SMC class based meaning scores of words are combined to form a score (or a likelihood if you will) for each class and used in Maximum Likelihood Estimation (MLE) setting much like NB. This is a simple yet effective classifier especially on small datasets. On the other hand, in CMK, meaning scores of words in the class scope is used to create a semantic kernel and that kernel is incorporated in SVM classifier. This can be seen as similar to latent semantic algorithm, however our approach is strictly supervised and it can be argued that we are mapping terms to class dimensions. The debate that this is one of the main reasons that CMK can handle highly noisy training sets because the similarity / distance calculations between documents (or support vectors) are improved largely by this class meaning kernel. As a summary, both approaches are based on meaning calculations in class context however, we believe the way they are implemented and combined makes ILBOM a novel semi-supervised approach and an interesting contribution to the related literature.

As an alternative future work related to ILBOM, it might be a good idea to analyze how the suggested methodology implicitly gets semantic information in the scope of a class when calculating the similarity between two documents. There is also a plan which includes building the iterative form of ILBOM and analyzing the performance differences especially at lower labeled set percentages. Additional item in our agenda is to expand our approach by adding further semantic-dimension(s) which will enrich the document representation and directly be contributing to the classification process. We think that this could be beneficial since it will expose hidden semantic relationships between documents and terms.

Acknowledgment

This work is supported in part by The Scientific and Technological Research Council of Turkey (TÜBİTAK) grant number 111E239. Points of view in this document are those of the authors and do not necessarily represent the official position or policies of TÜBİTAK.

References

Altınel, B., Ganiz, M.C., Diri, B., 2014a. A semantic kernel for text classification based on iterative higher–order relations between words and documents. In: Proceedings of the 13th International Conference on Artificial Intelligence and Soft Computing (ICAISC), Lecture Notes in Artificial Intelligence (LNAI), 8467, 505–517.

Altınel, B., Ganiz, M.C., Diri, B., 2014b. A simple semantic kernel approach for svm using higher-order paths. Proceedings of IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), 431-435.

Altınel, B., Ganiz, M.C., Diri, B., 2015. A corpus-based semantic kernel for text classification by using meaning values of terms. J. Eng. Appl. Artif. Intell.. http://dx.doi.org/10.1016/j.engappai.2015.03.015, (Elsevier).

Anagnostopoulos, I., 2010. A capture–recapture sampling standardization for improving Internet meta-search. Comput. Stand. Interfaces 32 (1), 61–70.

Balinsky, A., Balinsky, H., Simske, S., 2010. On the Helmholtz principle for documents processing. Proc. 10th ACM Doc. Eng. (DocEng).

Balinsky, A., Balinsky, H., Simske, S., 2011a. On the Helmholtz Principle for Data Mining. In: Proceedings of Conference on Knowledge Discovery, Chengdu, China.

Balinsky, A., Balinsky, H., Simske, S., 2011b. Rapid change detection and text mining. In: Proceedings of the 2nd Conference on Mathematics in Defense (IMA), Defense Academy, UK.

Balinsky, H., Balinsky, A., Simske, S., 2011c. Document sentences as a small World. In: Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 2583–2588.

Bloehdorn, S., Basili, R., Cammisa, M., Moschitti, A., 2006. Semantic kernels for text classification based on topological measures of feature similarity. In: Proceedings of the Sixth International Conference on Data Mining (ICDM), pp. 808–812.

Blum, A., Mitchell, T., 1998. Combining labeled and unlabeled data with co-training. Proceedings Conference on Computational Learning Theory, pp. 92–100.

Boser, B.E., Guyon, I.M., Vapnik, V.N., 1992. A training algorithm for Optimal margin classifier. In Proceedings of the 5th ACM Workshop. Comput. Learn. Theory, 144–152.

Chapelle, O., Zien, A., 2005. Semi-supervised classification by low density separation. Proc. Tenth Int. Workshop Artif. Intell. Stat..

Chapelle, O., Scholkopf, B., Zien, A., 2006. Semi-supervised learning. MIT Press, Cambridge.

Charalampopoulos, I., Anagnostopoulos, I., 2011. A comparable study employing weka clustering/classification algorithms for web page classification. In: Proceedings of the 15th Panhellenic Conference on Informatics (PCI), pp. 235–239. IEEE.

Cohen, W.W., Carvalho, V.R., Mitchell, T.M., 2004. Learning to classify email into "speech acts". EMNLP, 309–316.

Cozman, F.G., et al.2003. Semi-supervised learning of mixture models, In: Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003).

Dadachev, B., Balinsky, A. Balinsky, H., Simske, S., 2012. On the Helmholtz principle for data mining. In: International Conference on Emerging Security Technologies (EST), pp. 99–102.

Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R., 1990. Indexing by latent semantic analysis. J. Am. Soc. Inf. Sci. 41 (6), 391–407.

Ganiz, M.C., Lytkin, N.I., Pottenger, W.M., 2009. Leveraging higher-order dependencies between features for text classification. In: Proceedings of the Conference Machine Learning and Knowledge Discovery in Databases (ECML/PKDD), pp. 375–390.

Ganiz, M.C., George, C., Pottenger, W.M., 2011. Higher-order Naive Bayes: a novel non-IID approach to text classification. IEEE Trans. Knowl. Data Eng. (TKDE) 23 (7), 1022–1034.

Ganiz, M.C., Tutkan, M., Akyokus, S., 2015. A novel classifier based on meaning for text classification. InInternational Symposium on pp. 1–5. IEEE.

Guo, Y., Niu, X., Zhang, H., 2010. An extensive empirical study on semi-supervised learning. In: Proceedings of the 10th IEEE International Conference on Data Mining.

Guo, Y., Zhang, H., Liu, X., 2011. Instance selection in semi-supervised learning. In: Innovations in Intelligent SysTems and Applications (INISTA), Proceedings of the 24th Canadian Conference on Artificial Intelligence, pp. 158–169.

Hall, M, Frank, E. Holmes, G., Pfahringer, B., Reutemann, P., Witten, I. H., 2009. The WEKA Data Mining Software: An Update, SIGKDD Explorations, Volume 11, Issue 1.

Jiang, E.P., 2009. Semi-supervised text classification using RBF networks. In: Adams, N.M., Robardet, C., Siebes, A., Boulicaut, J.-F. (Eds.), IDA 5772. Springer, 95–106.

Jin, Y., Huang, C., Zhao, L., 2011. A semi-supervised learning algorithm based on modified self-training SVM. J. Comput. 6 (7), 1438–1443.

Joachims, T., 1999. Text Categorization with Support Vector Machines: learning with Many Relevant Features. Springer Berlin Heidelberg, 137–142.

Kleinberg, J., 2002. Bursty and Hierarchical Structure in Streams. In: Proceedings of the 8th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), 7(4), 373-397.

Kontostathis, A., Pottenger, W.M., 2006. A framework for Understanding LSI performance. J. Inf. Process. Manag., 56–73.

Li, M., & Zhou, Z. H., 2005. SETRED: Self-training with editing. In: Advances in Knowledge Discovery and Data Mining (pp. 611-621). Springer Berlin Heidelberg.

Li, K., Zhang, W., Ma, X., Cao, Z., Zhang, C., 2008. A novel semi-supervised SVM based on tri-training. In Intelligent Information Technology Application, Vol. 3, pp. 47–51.

Li, Y.F., Kwok, J.T., Zhou, Z.H., 2010. Cost-sensitive semi-supervised support vector machine. In: Proceedings of the 24th AAAI Conference on Artificial Intelligence, pp. 500–505.

Li, Y.F., Zhou, Z.H., 2011. Towards making unlabeled data never hurt. IEEE Trans. Pattern Anal. Mach.e Intell. 37/1, 175–188.

Liu, A., Jun, G., Ghosh, J., 2009. A self-training approach to cost sensitive uncertainty sampling. Mach. Learn. 76, 257–270.

Luo, Z.P., Zhang, X.-.M., 2008. A semi-supervised learning based relevance feedback algorithm in content-based image retrieval. In: Chinese Conference on Pattern Recognition (CCPR '08), pp. 1–4.

Mavroeidis, D., Tsatsaronis, G., Vazirgiannis, M., Theobald, M., & Weikum, G., 2005. Word sense disambiguation for exploiting hierarchical thesauri in text classification. In: Knowledge Discovery in Databases: PKDD. Springer Berlin Heidelberg, pp. 181–192.

Muslea, I., Minton, S., Knoblock, C.A., 2002. Active semi-supervised learning robust multi-view learning. ICML 2, 435–442.

Nigam, K., et al., 2000. Text classification from labeled and unlabeled documents using EM. Mach. Learn. 39 (2/3), 103–134.

Nigam, K., Ghani, R., 2000. Analyzing the effectiveness and applicability of co-training. In: Proceedings of the 9th ACM International Conference on Information and Knowledge Management.Washington, DC, pp. 86–93.

Razis, G., Anagnostopoulos, I., & Saloun, P., 2016. Thematic labeling of Twitter accounts using DBpedia properties. In: IEEE 11th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP), 2016, pp. 106-111.

Rosenberg, C., et al., 2005. Semi-supervised self-training of object detection models. Seven-. IEEE Workshop Appl. Comput. Vision..

Salton, G., Yang, C.S., 1973. On the specification of term values in automatic indexing. J. Doc. 29 (4), 11–21.

Schwenker, F., Trentin, E., 2014. Partially supervised learning for pattern recognition. Pattern Recognit. Lett. 37, 1–3.

Siolas, G., d'Alché-Buc, F., 2000. Support vector machines based on a semantic kernel for text categorization. In: Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN), Vol. 5, pp. 205–209.

Steinbach, M., Karypis, G., Kumar, V., 2000. A comparison of document clustering techniques. Proc. KDD Workshop Text. Min..

Vapnik, V.N., 1995. The Nature of Statistical Learning Theory. Springer, New York.

Wang, B., Spencer, B., Ling, C.X., Zhang, H., 2008. Semi-supervised self-training for sentence subjectivity classification. In: Proceedings of the 21st Canadian Conference on Artificial Intelligence, pp. 344–355.

Wang, P., Domeniconi, C., 2008. Building semantic kernels for text classification using wikipedia. In: Proceedings of the 14th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), pp. 713–721.

Wittek P., Tan, C., 2009. A Kernel-Based Feature Weighting For Text Classification. In: Proceedings of IEEE IJCNN-09, International Joint Conference on Neural Networks, pp. 3373–3379.

Yarowsky, D., 1995. Unsupervised word sense disambiguation rivaling supervised methods. Proc. 33rd Annu. Meet. Assoc. Comput. Linguist., 189–196.

Zelikovitz, S., Hirsh, H., 2004. Transductive LSI for short text classification problems. In: FLAIRS conference pp. 556–561.

Zhu, X.J., 2005. Semi-Supervised Learning Literature Survey (Technical Report) Department of Computer Sciences. University of Wisconsin at Madison, Madison, WI.