Knowl Inf Syst
DOI 10.1007/s10115-012-0477-x

REGULAR PAPER

Robust personalizable spam filtering via local and global discrimination modeling

Khurum Nazir Junejo · Asim Karim

Received: 29 June 2010 / Revised: 9 November 2011 / Accepted: 19 November 2011

Abstract Content-based e-mail spam filtering continues to be a challenging machine learning problem. Usually, the joint distribution of e-mails and labels changes from user to user and from time to time, and the training data are poor representatives of the true distribution. E-mail service providers have two options for automatic spam filtering at the service-side: a single global filter for all users or a personalized filter for each user. The practical usefulness of these options, however, depends upon the robustness and scalability of the filter. In this paper, we address these challenges by presenting a robust personalizable spam filter based on local and global discrimination modeling. Our filter exploits highly discriminating content terms, identified by their relative risk, to transform the input space into a two-dimensional feature space. This transformation is obtained by linearly pooling the discrimination information provided by each term for spam or non-spam classification. Following this local model, a linear discriminant is learned in the feature space for classification. We also present a strategy for personalizing the local and global models using unlabeled e-mails, without requiring user's feedback. Experimental evaluations and comparisons are presented for global and personalized spam filtering, for varying distribution shift, for handling the problem of gray e-mails, on unseen e-mails, and with varying filter size. The results demonstrate the robustness and effectiveness of our filter and its suitability for global and personalized spam filtering at the service-side.

Keywords E-mail classification · Local/global models · Distribution shift · Personalization · Gray e-mail

K. N. Junejo · A. Karim (✉)
Department of Computer Science,
LUMS School of Science and Engineering, Lahore, Pakistan
e-mail: akarim@lums.edu.pk

K. N. Junejo
e-mail: junejo@lums.edu.pk

Published online: 23 February 2012

1 Introduction

The problem of e-mail spam filtering has continued to challenge researchers because of its unique characteristics. A key characteristic is that of distribution shift, where the joint distribution of e-mails and their labels (spam or non-spam) changes from user to user and from time to time for all users. Conventional machine learning approaches to content-based spam filtering are based on the assumption that this joint distribution is identical in the training and test data. However, this is not true in practice because of the adversarial nature of e-mail spam, differing concepts of spam and non-spam among users, differing compositions of e-mails among users, and temporal drift in distribution.

Typically, a single spam filter is developed from generic training data, which is then applied to e-mails for all users. This global approach cannot handle the distribution shift problem. Specifically, the generic training data do not represent the e-mails received by individual users accurately as they are collected from multiple sources. A global filter is unlikely to provide accurate filtering for all users unless it is robust to distribution shift.

Recently, there has been significant interest in personalized spam filtering for handling the distribution shift problem (see [15]). In this approach, local or personalized filters are built for users from generic training data and their e-mails. For personalized spam filtering to be successful, it must (1) provide higher filtering performance when compared to global filtering, (2) be automatic in the sense that users' feedback on the labels of their e-mails is not required, and (3) have small memory footprints. The last requirement is critical for e-mail service providers (ESPs) who serve thousands and millions of users. In such a setting, implementing personalized spam filters at the service-side is constrained by the fact that all filters must reside in memory for real-time filtering.

Motivated by the flexibility of the LeGo (global models from local patterns) framework [44,56], the success of relative risk in the biomedical domain for identifying risk factors [29,52], and the robustness of statistical modeling for text classification [35,38], we develop a robust filter for global and personalized spam filtering. The filter is a robust text classifier capable of producing accurate classifications under different problem settings. The filter is also personalizable or adaptable in a semi-supervised fashion, satisfying the three desirable characteristics mentioned in the previous paragraph. The key ideas of the filter that contribute to its robustness and personalizability include: (1) supervised discriminative term weighting, which quantifies the discrimination information that a term provides for one class over the other. These weights are used to discover significant sets of terms for each class. (2) A linear opinion pool or ensemble for aggregating the discrimination information provided by terms for spam and non-spam classification. This allows a natural transformation from the input term space to a two-dimensional feature space. (3) A linear discriminant to classify the e-mails in the two-dimensional feature space. Items (1) and (2) represent a local discrimination model defined by two features, while item (3) represents a global discrimination model of e-mails. The performance of our filter is evaluated on six datasets and compared with four popular classifiers. Extensive results are presented with statistical evidence, demonstrating the robustness and personalizability of the filter. In particular, our filter performs consistently better than other classifiers in situations involving distribution shift. It is also shown to be scalable with respect to filter size and robust to gray e-mails.

We make the following contributions in this paper. (1) We define and discuss the challenges in spam filtering from a statistical point of view, highlighting issues like distribution shift and gray e-mails (Sect. 2). (2) We describe global and personalized spam filtering from an ESP's perspective and relate these filtering options to generative, discriminative, supervised, and semi-supervised learning (Sect. 2). (3) We present a new spam filter for global and

personalized spam filtering based on local and global discrimination modeling and compare it with popular generative, discriminative, and hybrid classifiers (Sect. 4). (4) We introduce a general classifier framework that emerges by generalizing our filters (Sect. 4). (5) We evaluate and compare our filter's performance with others on global and personalized spam filtering (Sect. 6). (6) We evaluate the performance of our filter for varying distribution shift, gray e-mail identification, robustness, and scalability (Sect. 6). The related work in spam filtering and data mining literature is discussed in Sect. 3, and the evaluation setup is described in Sect. 5.

2 The nature of the spam filtering problem

In this section, we define and describe the nature of e-mail classification, highlighting the key challenges encountered in the development and application of content-based spam filters. We introduce the typical e-mail classification problem, discuss the issue of distribution shift and gray e-mails, explore the applicability of global and personalized spam filtering options, and define semi-supervised global and personalized e-mail classification.

2.1 E-mail classification

The classification of e-mails into spam or non-spam based on the textual content of the e-mails given a set of labeled e-mails represents a prototypical supervised text classification problem. The idea is to learn a classifier or a filter from a sample of labeled e-mails, which when applied to unlabeled e-mails assigns the correct labels to them.

Let X be the set of all possible e-mails and $Y = \{+, -\}$ be the set of possible labels with the understanding that spam is identified by the label $+$. The problem of supervised e-mail classification can then be defined as follows:

Definition 1 (*Supervised e-mail classification*) Given a set of training e-mails $L = \{(x_i, y_i)\}_{i=1}^{N}$ and a set of test e-mails U, both drawn from $X \times Y$ according to an unknown probability distribution $p(x, y)$, learn the target function $\bar{\Phi}(x) : X \to Y$ that maximizes a performance score computed from all $(x, y) \in U$.

The joint probability distribution of e-mails and their labels, $p(x, y)$, completely defines the e-mail classification problem. It captures any selection bias and all uncertainties (e.g. label noise) in the concept of spam and non-spam in L and U. Thus, the e-mail classification problem can be solved by estimating the joint distribution from the training data. The learned target function can then be defined as

$$\bar{\Phi}(x) = y = \underset{v \in \{+,-\}}{\operatorname{argmax}} \bar{p}(x, v)$$

where $\bar{p}(x, v)$ denotes the estimate of the joint probability $p(x, v)$.

Since $p(x, y) = p(x|y)p(y)$ and $p(x, y) = p(y|x)p(x)$, it is customary and easier to estimate the component distributions on the right-hand sides rather than the full joint distribution directly. Given these decompositions of the joint distribution, the learned target function can be written in one of the following ways:

$$\bar{\Phi}(x) = y = \underset{v \in \{+,-\}}{\operatorname{argmax}} \bar{p}(x|v)\bar{p}(v) \tag{1}$$

$$\bar{\Phi}(x) = y = \underset{v \in \{+,-\}}{\operatorname{argmax}} \bar{p}(v|x) \tag{2}$$

Equation 1 represents a generative approach to supervised learning where the class prior and class-conditional distributions are estimated. It is called generative because these two distributions can be used to generate the training data. Equation 2 represents a discriminative approach to supervised learning where the posterior distribution of the class given the e-mail is estimated directly. Notice that in the discriminative approach, estimating the prior distribution of e-mails $\bar{p}(x)$ is not necessary because the classification of a given e-mail x depends on the posterior probability of the class given the e-mail only.

The performance score quantifies the utility of the learned target function or classifier. Typically when evaluating spam filters, the performance score is taken to be the accuracy and/or AUC value (area under the ROC curve) of the classifier. The AUC value is considered as a more robust measure of classifier performance since it is not based on a single decision boundary [6,17]. In supervised learning, only the labeled e-mails in L are available to the learner, and the learned target function is evaluated on the e-mails in U.

2.2 Distribution shift and gray e-mails

In the previous subsection, it was assumed that the training and test data, L and U, follow the same probability distribution $p(x, y)$. This assumption is not valid in many practical settings, where the training data come from publicly available repositories and the test data represent e-mails belonging to individual users. More specifically, the joint distribution of e-mails and their labels in L, $p_L(x, y)$ is not likely to be identical to that in U, $p_U(x, y)$. This arises from the different contexts (e.g. topics, languages, concepts of spam and non-spam, preferences) of the two data sets. Similarly, if U_i and U_j are the test sets belonging to user i and j, respectively, then we cannot expect the joint distribution of e-mails and their labels in these two sets to be identical.

Definition 2 (*Distribution shift*) Given sets of labeled e-mails L, U_i, and U_j, there exists a distribution shift between any two sets if any of the probability distributions $p(x, y)$, $p(x|y)$, $p(y|x)$, $p(x)$, and $p(y)$ are not identical for the two sets. The extent of distribution shift can be quantified in practice by information divergence measures such as Kullback–Leibler divergence (KLD) and total variation distance (TVD).

Because of distribution shift, it is likely that the learned target function will misclassify some e-mails, especially when the change in distribution occurs close to the learned decision boundary. In any case, a distribution shift will impact performance scores like the AUC value that are computed by sweeping through all decision boundaries.

Distribution shift can be quantified by KLD [9,47] and TVD [41], which are defined and used in Sect. 6. The extent of distribution shift can also be visualized from simple frequency graphs. Figure 1 illustrates shift in the distribution $p(x|y)$ between training data and a specific user's e-mails (test data). The top two plots show the probability of terms in spam (left plot) and non-spam (right plot) e-mails in the training data, while the middle two plots show the same distributions for an individual user's e-mails. The bottom two plots show the corresponding difference in distributions between the training and individual user's e-mails. Figure 2 shows the difference in distributions of $p(x|y)$ over two time periods. These figures illustrate the presence of distribution shift in practical e-mail systems.

A specific consequence of distribution shift is that of gray e-mail. If the distribution $p(y|x)$ in U_i and U_j is different to such an extent that the labels of some e-mails are reversed in the two sets, then these e-mails are called gray e-mails. They are referred to as gray because they are considered non-spam by some users and spam by other users.

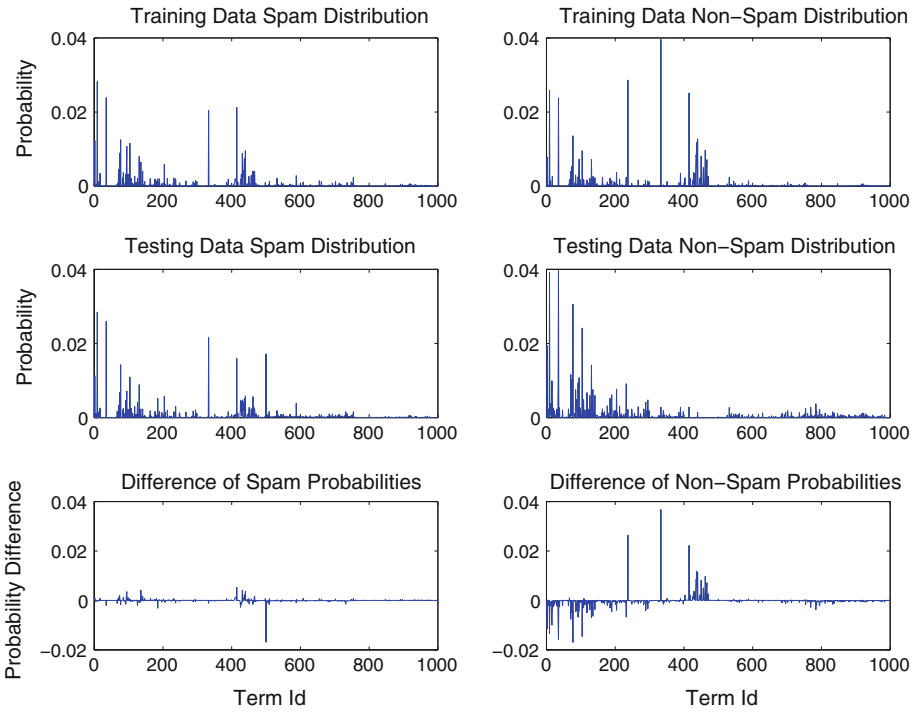

Fig. 1 Shift in $p(x|y)$ between training and test data (individual user's e-mails) (ECML-A data)

Definition 3 (*Gray e-mail*) An e-mail x is called a gray e-mail if its label in U_i and U_j is different. Gray e-mails can be identified in practice by finding highly similar e-mails in U_i and U_j that have different labels.

Typical examples of gray e-mails include mass distribution e-mails (e.g. newsletters, promotional e-mails) that are considered as spam by some but non-spam by others.

2.3 Global versus personalized spam filtering

The issue of global versus personalized spam filtering is important for e-mail service providers (ESPs). ESPs can serve thousands and millions of users and they seek a practical trade-off between filter accuracy and filter efficiency (primarily related to filter size). A personalized spam filtering solution, for example, may not be practical if its implementation takes up too much memory for storing and executing the filters for all users. A global spam filtering solution, on the other hand, may not be practical if it does not provide accurate filtering for all users. Thus, robustness and scalability are two significant characteristics desirable in a spam filter or classifier. If a filter is robust, then a global solution may provide acceptable filtering; otherwise, a personalized solution may be needed, but then it has to be scalable for it to be practically implementable.

In order to gain a better understanding of the suitability of personalized and global spam filtering solutions, it is important to know the settings under which each can be applied. Table 1 shows the learning settings under which global and personalized spam filtering can be applied. The first column indicates that a supervised global filter is possible, whereas

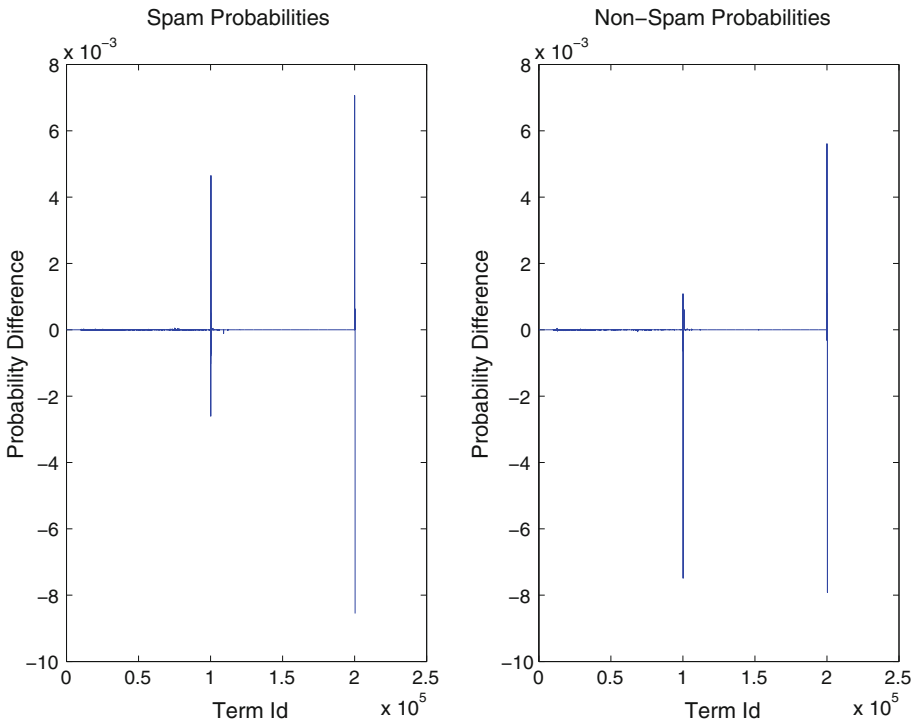

Fig. 2 Difference in $p(x|y)$ for e-mails from two different time periods (ECUE-1 data)

Table 1 Global and personalized spam filtering options (S'vised = Supervised)

	S'vised	Semi-S'vised	Semi-S'vised + Feedback	Hybrid
Global	✓	✓	✓	✓
Personalized	X	✓	✓	

supervised personalized filtering is not possible. This is because only the training data (L) are available to the learner while the labels of the e-mails belonging to individual users are not. Semi-supervised learning can be adopted for both global and personalized filtering solutions. Under this learning setting, the e-mails belonging to the users (without their labels) are available during the learning process in addition to the training data L. The actual way in which the unlabeled e-mails are utilized can vary from algorithm to algorithm. The third column indicates that both personalized and global filtering can be applied when users are requested to label some e-mails that belong to them. These labeled e-mails are then used during the learning process to build global or personalized filtering solutions. This strategy, however, is not automatic and places an additional burden on users in providing feedback on the received e-mails. Certain hybrid global/personalized and supervised/semi-supervised filtering solutions are also possible.

In this work, we focus on *automatic* supervised and semi-supervised learning for global and personalized spam filtering. This is the most desirable setting for an ESP. Semi-supervised global and personalized spam filtering can be defined as follows:

Definition 4 (*Semi-supervised global and personalized e-mail classification*) Given a set of labeled e-mails L (training data) and $M \geq 1$ sets of labeled e-mails U_i (e-mails belonging to user $i = 1, 2, \ldots, M$) drawn from $X \times Y$ according to (unknown) probability distributions $p_L(x, y)$ and $p_{U_i}(x, y)$, respectively, then

(a) a semi-supervised global filter learns the single target function $\bar{\Phi}(x) : X \rightarrow Y$ from L and U_i (without their labels) that maximizes a performance score computed from all $(x, y) \in \bigcup U_i$;
(b) a semi-supervised personalized filter learns M target functions $\bar{\Phi}_i(x) : X \rightarrow Y$ from L and U_i (without their labels) that, respectively, maximizes a performance score computed from all $(x, y) \in U_i$.

The joint probability distributions $p_L(x, y)$ and $p_{U_i}(x, y)$ are likely to be different from one another because of the differing contexts of the training data and the user's e-mails. Given this reality, it is unlikely that a single global filter trained on L only (supervised learning) will perform highly for all users, unless it is robust to shifts in distributions. Similarly, a semi-supervised approach, whether global or personalized, is likely to do better than a supervised approach as it has the opportunity to adapt to the users (unlabeled) e-mails. However, such an approach needs to be scalable and effective for its implementation at the ESP's end.

For semi-supervised learning, in addition to the training data L, the e-mails in U (without their labels) are also considered by the learner. In such a setting, the learned target function can be evaluated on the e-mails in U, as in the supervised learning setting. A better evaluation strategy, especially considering that a spam filter may not be updated continuously, is to evaluate on a randomly sampled hold out from U, $G \subset U$, with the learner seeing the unlabeled e-mails $U' = U \backslash G$, and the learned target function tested on the e-mails in G, which can be referred to as the generalization data.

2.4 Semi-supervised global versus personalized spam filtering

Which semi-supervised filtering solution is better: global or personalized? This question is difficult to answer in general as it depends on the algorithm and its robustness and scalability characteristics. It also depends upon the extent of the distribution shifts among L and U_i. In this section, we try to get a better understanding of this question under certain extreme distribution shift scenarios.

Let $p_{U_i}(x)$ and $p_{U_i}(y|x)$ be probability distributions in set U_i belonging to user i. Given this probabilistic viewpoint, four different scenarios can be defined: (1) $p_{U_i}(x) \ncong p_{U_j}(x), \forall i \neq j$ and $p_{U_i}(y|x) \ncong p_{U_j}(y|x), \forall i \neq j$, (2) $p_{U_i}(x) \cong p_{U_j}(x), \forall i, j$ and $p_{U_i}(y|x) \ncong p_{U_j}(y|x), \forall i \neq j$, (3) $p_{U_i}(x) \ncong p_{U_j}(x), \forall i \neq j$ and $p_{U_i}(y|x) \cong p_{U_j}(y|x), \forall i, j$, and (4) $p_{U_i}(x) \cong p_{U_j}(x), \forall i, j$ and $p_{U_i}(y|x) \cong p_{U_j}(y|x), \forall i, j$. The binary operator $\cong$ indicates that the shift between the two probability distributions is minor (i.e. the two distributions are approximately identical). Similarly, the binary operator $\ncong$ indicates that the shift between the two probability distributions is significant. For presentation convenience, operators $\cong$ and $\ncong$ are read as "identical" ("no distribution shift") and "not identical" ("distribution shift exists"), respectively. Whenever a distribution is identical among the users, it is assumed that it is the same distribution as that in the training data. Whenever a distribution is not identical among the users, it is assumed that all are different from that in the training data.

Scenario 1 is a difficult setting in which the users' e-mails and labels given e-mails both follow different distributions. For this scenario, it is expected that a personalized spam filtering solution will perform better as learning multiple joint distributions is difficult for a single algorithm, as would be required in a global filtering solution. An extreme case of this scenario

is when the vocabularies of e-mails for all the users are disjoint (e.g. when different users use different languages). In this setting, the size of the global filter will be identical to the sum of the sizes of the personalized filters. A similar statement can be made about scenario 2 regarding filtering performance. However, for this scenario, the size of each personalized filter will be identical to the size of the global filter. The problem of gray e-mails is likely to be present in scenarios 1 and 2, since in these scenarios, the distribution $p_{U_i}(y|x)$ is different between the users. The prevalence of gray e-mails suggests the preference of a personalized filtering solution. Scenario 3 does not involve concept shift while the distribution of e-mails is not identical across all users. For this scenario, which manifests as covariate shift in vector space representations, a global filter (built using all unlabeled e-mails) is expected to perform better especially when the distribution of e-mails is not very different. Scenario 4 represents a conventional machine learning problem with no distribution shift among the training and test data. Here, a global filter will be better in terms of both performance and space.

The scenarios described in the previous paragraph are based on a discriminative viewpoint of the e-mail classification problem. Similar scenarios can also be defined with a generative viewpoint using probability distributions $p(x|y)$ and $p(y)$.

Equations 1 and 2 define generative and discriminative probabilistic classifiers, respectively, for spam filtering. When there is no distribution shift, the probability estimates on the right-hand sides are based on the training data. On the other hand, when distribution shift exists the probability estimates must be based on the test data (U_i). However, the labels in the test data are not available to the learner, and it provides information about $p(x)$ only. When there is no shift in $p(y|x)$ and training data are abundant, a discriminant function learned from the training data will perform well on the test data, irrespective of the shift in $p(x)$. However, when the distribution $p(y|x)$ changes from training to test data, the discriminant function will require adaptation. Similarly, when the data generation process changes from training to test data, a generative classifier will require adaptation. Our approach, which is described in detail in Sect. 4, uses local and global discrimination models with semi-supervised adaptation of the models on the test data. The local model takes advantage of generative probabilities to discover discriminating patterns. The global model then performs pattern-based classification. This approach is flexible and is better able to adapt to distribution shift between training and test data.

3 Related work

In this section, we build the contextual background and motivation of our work by discussing the related work in the literature. For presentation convenience, we divide the discussion into two subsections: the first subsection covers work that focuses on spam filtering and the second covers work that focuses on relevant data mining and machine learning approaches.

3.1 Spam filtering

E-mail spam continues to be a menace that costs users, businesses, and service providers billions of dollars annually [4,26]. The problem of e-mail spam has engendered an industry sector that provides anti-spam products and services [49]. The spam generating community is also vibrant, as it develops new strategies and tools to distribute spam [68]. Many technological and non-technological measures have been developed to combat spam [15]. Among the technological measures, content-based filtering has proven to be a critical anti-spam measure. However, content-based spam filtering is challenging due to the nature of

e-mail spam. Some of these challenges are highlighted by Fawcett [24] as: (1) changing proportions of spam and non-spam e-mails, with time and with usage context (e.g. specific user). (2) Unequal and uncertain costs of misclassifications, making it difficult to evaluate the utility of spam filtering solutions. (3) Differing concepts of spam and non-spam e-mails among users, and (4) adversarial nature of spam, where spammers are continuously trying to evade spam filters. Challenges 1, 3, and 4 can be grouped under the general challenge of handling distribution shift in e-mail systems, as discussed in the previous section. In this work, we do not directly address challenge 2, although we do use the AUC (area under the ROC curve) value, in addition to filtering accuracy, for evaluating filtering performance. The AUC value is considered to be a robust measure of classifier performance that is often utilized for evaluating and comparing spam filters [6,17].

In recent years, there has been growing interest in personalized spam filtering where spam filters are adapted to individual user's preferences [12,13,27,37,39,66]. The adaptation is done in an effort to handle distribution shift between training e-mails and individual user's e-mails, which can cause a single global filter for all users to perform poorly. Personalized spam filtering solutions can be deployed at the client-side (user) or at the service-side (e-mail service provider or ESP). Effective service-side personalized spam filtering requires that the personalized filter be lightweight and accurate for it to be practically implementable for millions and billions of users served by the ESP [46].

Some solutions for personalized spam filtering rely upon user feedback regarding the labels of their e-mails [12,27,66,67]. This strategy burdens the e-mail user with the additional task of aiding the adaptation of the spam filter. Semi-supervised approaches that do not require user feedback have also been presented [7,13,37,39]. Junejo et al. [39] and Junejo and Karim [37] describe a statistical approach, called PSSF (personalized service-side spam filter) that transforms the input space into a two-dimensional feature space in which a linear discriminant is learned. Personalization is done by updating the transformation and linear discriminant on the (unlabeled) user's e-mails. In this paper, we extend our earlier work by developing a theoretical understanding of PSSF and presenting comprehensive evaluations of PSSF on various distribution shift scenarios including gray e-mail problem. To the best of our knowledge, this is the first time that a specific filter has been evaluated under different settings in a single paper. Semi-supervised learning approaches for handling covariate shift have been presented by Bickel and Scheffer [7] and Bickel et al. [8]. Their approaches assume that the vocabulary of terms and the probability distribution $p(y|x)$ are identical in the training and test data. Our approach makes none of these assumptions and can handle vocabulary changes easily since term space is transformed into a feature space via information pooling.

The notion of concept drift is closely related to distribution shift with the implicit understanding that distribution shift occurs continuously over time. Delany et al. [20] propose a case-based reasoning approach for handling concept drift in e-mail systems. Their strategy for maintaining the case base requires knowledge of the correct labels. Ensemble approaches are known to be robust to drifting data. The ensemble update process can be supervised [21] or semi-supervised [13,40]. Cheng and Li [13] present a graph-based semi-supervised approach combining support vector machine (SVM), naive Bayes classifier, and rare word distribution for personalized spam filtering. Katakis et al. [40] address the problem of recurring contexts in e-mail systems by presenting an ensemble method in which each classifier in the ensemble models a particular context. Our local model uses an ensemble approach for constructing features from discriminating terms while we adopt the naive semi-supervised learning approach for adaptation [70].

3.2 Data mining and machine learning

The data mining and machine learning literature has extensive coverage of learning paradigms and problem settings appropriate for spam filtering. A comprehensive discussion of all related areas is beyond the scope of this paper. We focus on key contributions and areas that have direct relevance to our work in this paper.

The literature on text classification has direct relevance to content-based spam filtering. A survey of supervised text classification is given by Sebastiani [64]. Supervised text classifiers can be based on generative or discriminative learning. The most common generative classifier is naive Bayes [45,65]. This classifier results from the application of the Bayes rule with the assumption that each variable is independent of the others given the class label. Another successful probabilistic classifier, which has similarities to naive Bayes [36], is maximum entropy [59]. The maximum entropy classifier estimates the joint probability distribution by maximizing its entropy constrained by the empirical distribution. The most popular discriminative text classifier is SVM [33,35,61]. SVM, which is based on statistical learning theory and structural risk minimization, learns a maximum margin linear discriminant in a (possibly) high-dimensional feature space. The balanced winnow is another example of a discriminative classifier that learns a linear discriminant in the input space by minimizing the mistakes made by the classifier [18].

Over the years, there has been some interest in combining generative and discriminative learning for classification [30,38,58,62]. These classifiers try to exploit the strengths of generative and discriminative learning by first learning the data distribution and then building a discriminative classifier using the learned distribution. Several variants of this general concept have been explored with promising results. Our work extends the supervised text classification algorithm presented by Junejo and Karim [38] to semi-supervised classification of e-mails. Although the algorithms presented in this work are not truly hybrid generative/discriminative in nature, they have close correspondence to such algorithms, as discussed in Sect. 4.

The semi-supervised learning paradigm was originally proposed for improving classifier performance on a task when labeled data are limited but unlabeled data are plentiful for the task. For this setting, several approaches have been proposed such as generative mixture models, self-training, co-training, and graph-based propagation [72,73]. In recent years, however, all approaches that rely upon labeled and unlabeled data are considered to be semi-supervised, regardless of whether their problem settings and motivations are identical to those proposed for the paradigm originally. As such, semi-supervised learning has been applied to problems involving distribution shift. Xue and Weiss [70] investigate quantification and semi-supervised learning approaches for handling shift in class distribution. Bickel et al. [8] present a discriminative learning approach for handling covariate shift. It should be noted that for handling covariate shift, quantification is not necessary as $p(x)$ can be estimated from the unlabeled data. Another machine learning paradigm with relevance to personalized spam filtering is transfer learning. Transfer learning involves the adaptation of a classifier for one task to a related task, given data for both tasks. Many transfer learning approaches are semi-supervised requiring only unlabeled data for the new task [63,69].

Recently, there has been much interest in the domain adaptation problem, where a classifier is adapted from a source domain to a target domain [32]. The target domain data may be labeled or unlabeled. Blitzer et al. [10] present a domain adaptation approach based on learning frequently occurring features (pivot features) in the source and target domains. The weights of the learned classifiers are then used to transform the input representation into a lower-dimensional feature space in which the task classifier is built. The importance of

having a feature space in which the two domains are less different is highlighted by Ben-David et al. [5].

Building global models from local patterns is a promising approach to classification [11, 43,44,56]. This approach, often called the LeGo (from local patterns to global models) framework for data mining, focusses on finding relevant patterns in data that are then used as features in global models of classification. Local patterns can be model-independent or model-dependent [11]. Also, it is desirable that these patterns form a non-redundant and optimal set or pattern team [43]. Our local patterns (significant term sets) are model-dependent as they are based on discrimination information. Furthermore, our local patterns form pattern teams based on the relative risk statistical measure. Our global model is a linear classifier that operates on features derived from the pattern teams. We also relate our local and global approach to hybrid generative/discriminative models for classification.

In data mining research, there is growing reliance on statistically sound measures for quantifying the relevance of patterns [52]. Efficient algorithms for discovering risk patterns, defined as itemsets with high relative risk, are discussed by Li et al. [53], while direct discovery of statistically sound association rules is presented by Hämäläïnen [28]. The use of statistical measures of association, such as relative risk, odds ratio, and risk difference, is motivated from biomedical studies where they have been used widely for some time [29,50]. Such measures have also been used in the language processing literature for quantifying term association [14].

A statistical model of text classification tasks is presented by Joachims [35]. Based on this model, he develops a characterization of the generalization performance of an SVM. He highlights that redundancy (multiple terms describing a specific class), discriminative power of terms (and term sets), and high frequency terms are desirable for robust classification. Statistical measures for term selection have been investigated by Forman [25], while term weighting with a relative risk variant for naive Bayes classification has been proposed by Kim et al. [42]. The idea of discriminative term weighting for text classification is introduced by Junejo and Karim [38]. Their algorithm, called DTWC, uses relative risk, log relative risk, or Kullback–Leibler divergence (KLD) to quantify the discrimination information provided by terms, which is then used to construct a two-dimensional feature space for linear classification. They report robust performance on text classification tasks, especially when using the relative risk discriminative term weighting strategy. More recently, Malik et al. [55] adopt a similar approach for text classification, relying upon information gain for term weighting and classifying directly from feature scores. The work presented in this paper is motivated by the robustness of relatively simple statistical models for text classification, and it extends the DTWC algorithm for semi-supervised learning and evaluation under different distribution shift scenarios.

4 DTWC/PSSF: a Robust and personalizable spam filter

In this section, we describe a robust and personalizable content-based spam filter suitable for global and personalized filtering of e-mails at the ESP's end. The filter is robust in the sense that its performance degrades gracefully with increasing distribution shift and decreasing filter size. The filter can be used for global as well as personalized filtering and can take advantage of each user's e-mails in a semi-supervised fashion without requiring feedback from the user.

The filter relies upon local and global discrimination models for its robustness and scalability. The key ideas of the supervised filter are the following: (1) identification of significant

content terms based on the discrimination information provided by them, quantified by their relative risk in spam and non-spam e-mails (called their discriminative term weights), (2) discrimination information pooling for the construction of a two-dimensional feature space, and (3) a discriminant function in the feature space. In a semi-supervised setting, either or both the local model (the discriminative term weights and the two scores) and the global model (discriminant function) are updated using the e-mails of each user to personalize the filter.

We name the supervised filter as DTWC (discriminative term weighting-based classifier) and its semi-supervised extension as PSSF (personalized service-side spam filter) in accordance with our previous work [37,38].

We start the discussion on DTWC/PSSF by describing the e-mail representation in the following section. The local and global models in DTWC/PSSF are presented next. These models are discussed for a supervised setting first followed by their extension to a semi-supervised setting for building personalized filters. Finally, we interpret and compare our algorithms with popular generative, discriminative, and hybrid classifiers.

4.1 E-mail representation

DTWC/PSSF is a content-based spam filter. For mathematical convenience, it is assumed that e-mails are represented in a vector space whose dimensions are the key attributes of the classification problem. The ith e-mail is defined by the vector $\mathbf{x}_i = \langle x_{i1}, x_{i2}, \ldots, x_{iT} \rangle$, where T is the size of the vector space and $x_{ij} \geq 0, \forall(i, j)$ is the value of the jth attribute. Please note that, unlike in the previous sections, an e-mail is denoted by the boldface $\mathbf{x}$. Typically, each attribute is a distinct term or token (e.g. word) in the e-mails' contents. Its value is defined by a term weighting strategy [64]. Our formulation assumes a binary (0/1) weighting strategy, although this can be extended to term count, term frequency, and term-frequency–inverse-document-frequency term weighting strategies easily. For simplicity of presentation, we also use the notation x_j (with a single subscript) to denote the jth term of an e-mail. Moreover, we denote the probability of occurrence of term j as $p(x_j)$.

4.2 Local patterns of spam and non-spam e-mails

DTWC/PSSF is based on a novel local model of spam and non-spam e-mails. The terms in the vocabulary are partitioned into significant spam and non-spam terms depending on their prevalence in spam and non-spam e-mails, respectively. The e-mails are considered to be made up of significant spam and non-spam terms, with each term expressing an opinion regarding the classification of the e-mail in which it occurs. The overall classification of an e-mail is based on the aggregated opinion expressed by the significant terms in it. The discriminative term weight quantifies the discrimination information (or opinion) of a term, while a linear opinion pool is formed to aggregate the opinions expressed by the terms. The aggregated opinions (one for spam and one for non-spam class) represent local patterns or features of an e-mail, learned from the training data, that are input to a global classification model (described in the next subsection).

4.2.1 Significant spam and non-spam terms

A term j in the vocabulary is likely to be a spam term if its probability in spam e-mails, $p(x_j|y = +)$, is greater than its probability in non-spam e-mails, $p(x_j|y = -)$. A term j is a significant spam (non-spam) term if $p(x_j|y = +)/p(x_j|y = -) > t$ ($p(x_j|y = -)/$

$p(x_j|y = +) > t$), where $t \geq 1$ is a term selection parameter. Given the above, the index sets of significant spam and non-spam terms (Z^+ and Z^-) can be defined as follows:

$$Z^+ = \left\{ j \left| \frac{p(x_j|y=+)}{p(x_j|y=-)} > t \right. \right\}, \quad \text{and} \tag{3}$$

$$Z^- = \left\{ j \left| \frac{p(x_j|y=-)}{p(x_j|y=+)} > t \right. \right\}, \tag{4}$$

where index j varies from 1 to T. It should be noted that $Z^+ \cap Z^- = \emptyset$, indicating that a hard partitioning of terms is done. However, $|Z^+ \cup Z^-|$ is generally not equal to T.

The probability ratios in Eqs. 3 and 4 are the relative risks of term j in spam and non-spam e-mails, respectively. We discuss this aspect in more detail in the following subsection.

The parameter t serves as a term selection parameter and can be used to tune the size of the spam filter. If $t = 1$, then all spam and non-spam terms are retained in the significant term model. As the value of t is increased, less significant (or less discriminating, as explained in the next subsection) terms are removed from the model. This is a supervised and more direct approach for term selection when compared to the common techniques used in practice like term document frequency and principal component analysis. Effective term selection is important for creating lightweight personalized filters for large-scale service-side deployment.

Alternatively, significant spam and non-spam terms can be selected by the conditions $p(x_j|y = +) - p(x_j|y = -) > t'$ and $p(x_j|y = -) - p(x_j|y = +) > t'$, respectively (see Eqs. 3 and 4). Here, $t' \geq 0$ is a term selection parameter, which has the same semantics as the parameter t discussed above.

The probabilities $p(x_j|y)$ are estimated from the training data (the set L of labeled e-mails) as the fraction of e-mails in which term j occurs:

$$\bar{p}(x_j|y) = \frac{\sum_{i \in L^y} x_{ij}}{|L^y|},$$

where L^y denotes the set of e-mails in L belonging to class $y \in \{+, -\}$ and $|L^y|$ is the number of e-mails in the set. To avoid division by zero, we assign a small value to the probabilities that are coming out to be zero.

4.2.2 Discriminative term weighting

E-mails are composed of significant spam and non-spam terms. Each term in an e-mail is assumed to express an opinion about the label of the e-mail—spam or non-spam. This opinion can be quantified by discrimination information measures that are based on the distribution of the term in spam and non-spam e-mails.

If an e-mail $\mathbf{x}$ contains a term j (i.e. $x_j = 1$), then it is more likely to be a spam e-mail if $p(x_j|y = +)$ is greater than $p(x_j|y = -)$. Equivalently, an e-mail $\mathbf{x}$ is likely to be a spam e-mail if the relative risk for spam of a term j in it is greater than one. This can be expressed mathematically as

$$\frac{p(y=+|x_j)}{p(y=-|x_j)} \propto \frac{p(x_j|y=+)}{p(x_j|y=-)} > 1 \tag{5}$$

Given this observation, we define the discriminative term weight w_j for terms $j = 1, 2, \ldots, T$ as

$$w_j = \begin{cases} p(x_j|y=+)/p(x_j|y=-) & \forall j \in Z^+ \\ p(x_j|y=-)/p(x_j|y=+) & \forall j \in Z^- \end{cases} \tag{6}$$

The discriminative term weights are always greater than or equal to 1. The larger the value of w_j the higher is the discrimination information provided by term j. The inclination of the discrimination information is determined by whether the term is a significant spam term or a significant non-spam term.

It is worth pointing out the distinction between discriminative term weights (DTWs) and (document) term weights (TWs). First, DTWs quantify the discrimination information that terms provide for classification, while TWs quantify the significance of a term within a document. Second, DTWs are defined globally for every term in the vocabulary, while TWs are defined locally for every term in a document. Third, DTWs are computed in a supervised fashion rather than the usual unsupervised computation of TWs. DTWs are not a substitute for TWs; they are defined for a different purpose of classification rather than representation.

Relative risk or risk ratio has been used in medical domains for analyzing the risk of a specific factor in causing a disease [29,52]. This is done as part of prospective cohort studies, where two groups of individuals, with one group exposed and the other unexposed to the factor, are observed for the development of the disease. The relative risk of the factor is determined by the ratio of the proportion of exposed individuals developing the disease to the proportion of exposed individuals not developing the disease. Here, we adopt the relative risk for identifying terms that are likely to make an e-mail to be a spam or non-spam e-mail. We also use the relative risk to quantify the discrimination information provided by a term.

It is interesting to note that the sets of significant spam and non-spam terms (Z^+ and Z^-) represent pattern teams [43] for a given value of t. To see this, define $q(Z) = \sum_{j \in Z} w_j$ to be the quality measure of a set of patterns (terms) defined by the index set Z. Then, it follows from Eq. 3 that there exists no other set of patterns of size $|Z^+|$ with a higher value of $q(Z^+)$. A similar reasoning shows that the set of patterns defined by Z^- is a pattern team. In other words, these sets are optimal and non-redundant with respect to the quality measure. This quality measure quantifies the collective discriminative power of a set of terms, and a pattern team with respect to this measure and a value of t will retain maximum discrimination power. For each e-mail, a feature is constructed from each pattern team by utilizing this definition of the quality measure, as discussed next.

4.2.3 Linear opinion pooling

The classification of an e-mail depends on the prevalence of significant spam and non-spam terms and their discriminative term weights. Each term $j \in Z^+$ in e-mail $\mathbf{x}$ expresses an opinion regarding the spam classification of the e-mail. This opinion is quantified by the discriminative term weight w_j. The aggregated opinion of all these terms is obtained as the linear combination of individual terms' opinions:

$$Score^+(\mathbf{x}) = \frac{\sum_{j \in Z^+} x_j w_j}{\sum_j x_j} \tag{7}$$

This equation follows from a linear opinion pool or an ensemble average, which is a statistical technique for combining experts' opinions [2,31]. Each opinion (w_j) is weighted by the normalized term weight ($x_j / \sum x_j$) and all weighted opinions are summed yielding an

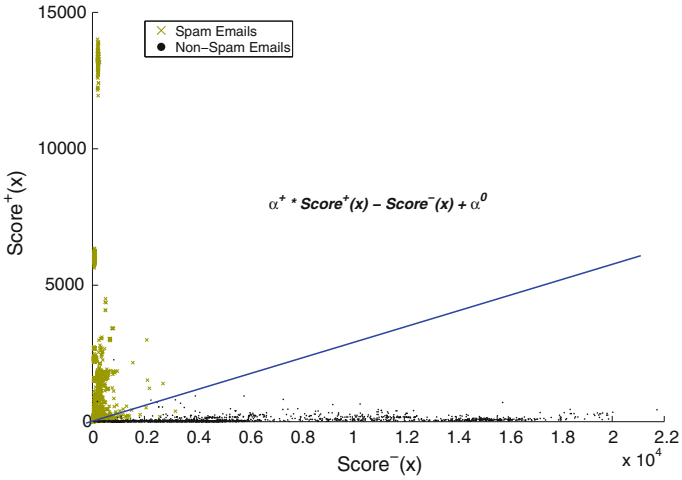

Fig. 3 The two-dimensional feature space and the linear discriminant function for e-mail classification

aggregated spam score ($Score^{+}(\mathbf{x})$) for the e-mail. If a term i does not occur in the e-mail (i.e. $x_i = 0$), then it does not contribute to the pool. Also, terms that do not belong to set Z^{+} do not contribute to the pool. Similarly, an aggregated non-spam score can be computed for all terms $j \in Z^{-}$ as

$$Score^{-}(\mathbf{x}) = \frac{\sum_{j \in Z^{-}} x_j w_j}{\sum_j x_j}. \tag{8}$$

These scores represent constructed features based on a local model of discrimination. E-mail classification can be done directly using these scores as follows:

$$y(\mathbf{x}) = \underset{v \in \{+,-\}}{\operatorname{argmax}}\, Score^{v}(\mathbf{x}). \tag{9}$$

In other words, the classification label of a new e-mail $\mathbf{x}$ is determined by the greater of the two scores, spam score or non-spam score.

4.3 Global discriminative model of spam and non-spam e-mails

This section describes a discriminative model that is built using the features found in the local model. The use of both local and global learning allows greater flexibility in updating model parameters in the face of distribution shift. It also contributes to the robustness of the classifier to distribution shift.

The local model yields a two-dimensional feature space defined by the scores, $Score^{+}(\mathbf{x})$ and $Score^{-}(\mathbf{x})$. In this feature space, e-mails are well separated and discriminated, as illustrated in Fig. 3. DTWC/PSSF classifies e-mails in this space by a linear discriminant function:

$$f(\mathbf{x}) = \alpha^{+} \cdot Score^{+}(\mathbf{x}) - Score^{-}(\mathbf{x}) + \alpha^{0}, \tag{10}$$

where α^{+} and α^{0} are the slope and bias parameters, respectively. The discriminating line is defined by $f(\cdot) = 0$. If $f(\cdot) > 0$, then the e-mail is likely to be spam (Fig. 3).

The global discrimination model parameters are learned by minimizing the classification error over the training data. This represents a straightforward optimization problem that can

be solved by any iterative optimization technique [54]. DTWC/PSSF's learned target function is defined as

$$\bar{\Phi}(\mathbf{x}) = y = \begin{cases} + & f(\mathbf{x}) > 0 \\ - & \text{otherwise} \end{cases} \tag{11}$$

The supervised spam filter DTWC is shown in Algorithm 1.

Algorithm 1 DTWC—Supervised Spam Filter

Input: L (training data – labeled e-mails), U (test data – unlabeled e-mails)
Output: labels for e-mails in U

On training data L
-Build local model
form index sets Z^+ and Z^- (Eqs. 3 and 4)
compute $w_j, \forall j \in (Z^+ \cup Z^-)$ (Eq. 6)
transform $\mathbf{x} \mapsto [Score^+(\mathbf{x})\ Score^-(\mathbf{x})]^T, \forall \mathbf{x} \in L$ (Eqs. 7 and 8)

-Build global model
learn parameters α^+ and α^0 (see Eq. 10)

On test data U
-Apply learned models
for $\mathbf{x} \in U$ **do**
compute $Score^+(\mathbf{x})$ and $Score^-(\mathbf{x})$ (Eqs. 7 and 8)
compute $f(\mathbf{x})$ (Eq. 10)
output $\bar{\Phi}(\mathbf{x})$ (Eq. 11)
end for

4.4 Personalization

The previous subsections described a supervised algorithm for spam filtering. This can be used to build a single global filter for all users given training data (L). A global filtering solution, however, is not appropriate when there is a distribution shift between the training data and the individual user's e-mails. In such a situation, which is common in e-mail systems, personalization of the filter is required for each user. Since the e-mails of users are unlabeled, a semi-supervised approach is needed, as discussed in Sect. 2.

DTWC is adapted to an individual user's e-mails (U_i) as follows. Use the learned target function from training data to label the e-mails in U_i. Since labels are now available, update the local and global models to form a new target function. Use this updated target function to finally label the e-mails in U_i. This approach corresponds to the naive semi-supervised learning (SSL) used in Xue and Weiss [70]. The motivation for using naive SSL instead of more sophisticated approaches (e.g. self-training SSL) are twofold. First, most of the other approaches are proposed for situations where the training and test data follow the same distribution. In e-mail systems, on the other hand, the distribution of e-mails in training data can be significantly different from that in the test data, and this difference is in general not easily quantifiable. Second, in e-mail systems, distribution shift occurs continuously over time and it is better to personalize a filter from one time period to another rather than from training data to current time period. The naive SSL approach decouples the adaptation from the training data and as such can be performed on the client-side in addition to the service-side.

Algorithm 2 PSSF1/PSSF2—Personalized Spam Filter

Input: L (training data – labeled e-mails), U_i (test data – unlabeled e-mails)
Output: labels for e-mails in U_i

$U_i' \leftarrow U_i$ labeled with DTWC(L, U_i)

On labeled e-mails U_i'
-Build local model
-Build global model **[PSSF2 Only]**

On test data U_i
-Apply learned models

The local model can be updated incrementally as new e-mails are seen by the filter capturing the changing distribution of e-mails received by the user. The global model can be rebuilt at periodic intervals (e.g. every week) to cater for significant changes in the distribution of e-mails.

The semi-supervised version of our filter is called PSSF (personalized service-side spam filter). We further differentiate between two variants of PSSF as PSSF1 or PSSF2. PSSF1 is the variant in which only the local model (the spam and non-spam scores) is updated over the user's e-mails (U_i) and the global model is not updated from the model learned over the training data. PSSF2 is the variant in which both the local and global models are updated over the user's e-mails (U_i). PSSF1/PSSF2 is described in Algorithm 2.

4.5 Interpretations and comparisons

In this section, we provide a broader interpretation of our spam filter by comparing it with generative, discriminative, and hybrid classifiers. We also propose a general framework for building adaptable classifiers and relate it to the local pattern discovery and global modeling framework of data mining.

4.5.1 Naive Bayes classifier

The naive Bayes classifier is popularly used for text and content-based e-mail classification. Using the Bayes rule, the odds that an e-mail $\mathbf{x}$ is spam rather than non-spam can be written as

$$\frac{p(y=+|\mathbf{x})}{p(y=-|\mathbf{x})} = \frac{p(\mathbf{x}|y=+)p(y=+)}{p(\mathbf{x}|y=-)p(y=-)}$$

Assuming that the presence of each term is independent of others given the class, the e-mail risk on the right-hand side become a product of terms' risks. The naive Bayes classification of the e-mail $\mathbf{x}$ is spam (+) when

$$\frac{p(y=+)}{p(y=-)} \prod_j \left(\frac{p(x_j|y=+)}{p(x_j|y=-)} \right)^{x_j} > 1 \tag{12}$$

Equivalently, taking the log of both sides, the above expression can be written as

$$\log \frac{p(y=+)}{p(y=-)} + \sum_j x_j \log \frac{p(x_j|y=+)}{p(x_j|y=-)} > 0 \tag{13}$$

This equation computes an e-mail score, and when this score is greater than zero, the naive Bayes classification of the e-mail is spam. Notice that only those terms are included in the summation for which $x_j > 0$.

Comparing the naive Bayes classifier, as expressed by Eq. 13, with DTWC/PSSF yields some interesting observations. The global discriminative model of DTWC/PSSF is similar to Eq. 13 in that the structure of the spam and non-spam score computations (Eqs. 7 and 8) is similar to the summation in Eq. 13, and the bias parameter α^0 corresponds to the first term in Eq. 13.

However, there are also significant differences between DTWC/PSSF and naive Bayes. (1) DTWC/PSSF partitions the summation into two based on discrimination information and then learns a linear discriminative model of the classification. Naive Bayes, on the other hand, is a purely generative model with no discriminative learning of parameters. (2) DTWC/PSSF, as presented in this work, involves summation of terms' relative risks rather than terms' log relative risks as in naive Bayes. However, as discussed in Junejo and Karim [38], other measures of discrimination information can be used instead of relative risk. (3) The spam and non-spam scores in DTWC/PSSF are normalized (for each e-mail) using the $L1$ norm. This normalization arises naturally from linear opinion pooling. Document length normalization is typically not done in naive Bayes classification, and when it is done, the $L2$ norm is used. It has been shown that performing $L1$ document length normalization improves the precision of naive Bayes for text classification [45].

4.5.2 Discriminative and hybrid classifiers

Popular discriminative classifiers learn a hyperplane or linear discriminant in the space representing the objects to be classified (e-mails in our case). Let $\phi : X \rightarrow V$ be the function that maps an e-mail $\mathbf{x}$ from the T-dimensional input space to $\mathbf{v}$ in a d-dimensional feature space. Then, a hyperplane in the feature space is defined by

$$\sum_{j=1}^{d} \alpha_j v_j + \alpha_0 = 0, \tag{14}$$

where $\alpha_j (j = 1, 2, \ldots, d)$ are the parameters of the hyperplane.

DTWC/PSSF's global model is also a linear discriminant. However, this discriminant function is learned in a two-dimensional feature space defined by spam and non-spam scores and has only two parameters. Input-to-feature space transformation is typically not done for discriminative classifiers like balanced winnow/perceptron and logistic regression. In SVM, this transformation is done implicitly through the inner product kernel $k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$, where $\phi(\cdot)$ is the function that maps from input to feature space.

The input-to-feature space transformation in DTWC/PSSF can be written as

$$\phi(\mathbf{x}) = [\phi_1(\mathbf{x})\ \phi_2(\mathbf{x})]^T = \left[Score^+(\mathbf{x})\ Score^-(\mathbf{x})\right]^T, \tag{15}$$

where the scores are defined in Eqs. 7 and 8. This represents a linear mapping from a T-dimensional input space to a two-dimensional feature space. The kernel is then defined as follows (after substitution and using vector notations):

$$k(\mathbf{x}, \mathbf{x}') = \phi^T(\mathbf{x})\phi(\mathbf{x}') = \bar{\mathbf{x}}^T \mathbf{W}^+ \bar{\mathbf{x}'} + \bar{\mathbf{x}}^T \mathbf{W}^- \bar{\mathbf{x}'}, \tag{16}$$

where $\mathbf{W}^+ = \mathbf{w}^+ \mathbf{w}^{+T}$ and $\mathbf{W}^- = \mathbf{w}^- \mathbf{w}^{-T}$ are $T \times T$-dimensional matrices and $\bar{\mathbf{x}} = \mathbf{x}/\|\mathbf{x}\|_{L1}$, $\bar{\mathbf{x}'} = \mathbf{x}'/\|\mathbf{x}'\|_{L1}$. The elements of vector $\mathbf{w}^+$ (vector $\mathbf{w}^-$) are equal to w_j (Eq. 6)

when $j \in Z^+$ ($j \in Z^-$) and zero otherwise. Noting that terms in the vocabulary are hard-partitioned, we can write

$$k(\mathbf{x}, \mathbf{x}') = \bar{\mathbf{x}}^T \mathbf{W} \bar{\mathbf{x}'} \tag{17}$$

where $\mathbf{W} = \mathbf{W}^+ + \mathbf{W}^-$.

The following observations can be made from the above discussion. (1) DTWC/PSSF performs a linear transformation from the input space to a *lower* dimension feature space. This feature space is formed in such a way that the discrimination between spam and non-spam e-mails is enhanced. Recently, it has been shown that feature space representations are critical to making classifiers robust for domain adaptation [1,5]. (2) The matrices $\mathbf{W}$, $\mathbf{W}^+$, and $\mathbf{W}^-$, which are symmetric, are smoothing matrices. The associated kernel is positive semi-definite. (3) The transformation is supervised, requiring information about class labels.

Jaakkola and Haussler [30] discuss a hybrid classifier in which the kernel function is derived from a generative model. The input-to-feature space transformation is based on the Fisher score and the resulting kernel is a Fisher kernel. Our input-to-feature space transformation is based on discrimination scores computed from discrimination information provided by the terms in the e-mails.

Raina et al. [62] present another hybrid classifier in which the input space (defined by the terms) is partitioned into two sets (based on domain knowledge) and the weights for each set of class-conditional distributions are learned discriminatively. DTWC/PSSF's global model parameters are similar in purpose; however, the two sets of class-conditional distributions in DTWC/PSSF correspond to the sets of significant spam and non-spam terms, which are determined from labeled e-mails.

4.5.3 Classifier framework

DTWC/PSSF are specific algorithms based on a general classifier framework. This framework emerges when the specific decisions made for DTWC and PSSF are generalized. Specifically, the classifier framework contains the following components: (1) Identification of descriptors or experts (we use terms), (2) quantification of descriptors' or experts' discrimination opinions (we use terms' relative risk), (3) combining experts' opinions (we use a linear opinion pool), (4) transformation from input to feature space (we do a linear transformation), (5) discriminative classification in the feature space (we use a linear discriminant function), and (6) classifier adaptation for semi-supervised learning (we adapt naive SSL). This is a rich framework worth exploring in future.

5 Evaluation setup: datasets and algorithms

We present extensive evaluations of DTWC, PSSF1, and PSSF2 on six spam filtering datasets and compare their performances with four other classifiers. We present results for a single global filter trained in a supervised fashion on the training data, as well as for personalized filters trained in a semi-supervised fashion. We also evaluate the robustness and scalability characteristics of our algorithms in several ways: (1) by varying distribution shift, (2) by evaluating performance on gray e-mails, (3) by testing on unseen e-mails, and (4) by varying filter size. For all the algorithms, we report filtering performance with the percent accuracy and the percent AUC value. Finally, we present significance test results showing that our algorithms are significantly better than the others.

Table 2 Evaluation datasets and their characteristics

	ECML-A	ECML-B	ECUE-1	ECUE-2	PU1	PU2
No. of training e-mails	4,000	100	1,000	1,000	672	497
No. of users/test sets	3	15	1	1	1	1
No. of e-mails per user/test set	2,500	400	2,000	2,000	290	213
Distribution shift	Yes		Yes (temporal)		No	

In this section, we describe the datasets and the setup for the comparison algorithms. The details of the various evaluations and their results are presented in the next section.

5.1 Datasets

Our evaluations are performed on six commonly used e-mail datasets: ECML-A, ECML-B, ECUE-1, ECUE-2, PU1, and PU2. Some characteristics of these datasets, as used in our evaluations, are given in Table 2. The selected datasets vary widely in their characteristics. Two datasets have distribution shift between training and test sets and among different test sets. Two datasets have concept drift from training to test sets, and the remaining two datasets have no distribution shift between training and test sets. The number of e-mails in training and test sets varies greatly in the selected datasets. The number of e-mails varies from 100 to 4,000 in the training sets and from 213 to 2,500 in the test sets. In some datasets, the size of the training set is larger than the size of the test set, while in others, it is the opposite. These datasets are described in greater detail subsequently.

5.1.1 ECML-A and ECML-B datasets

The ECML-A and ECML-B datasets correspond to the datasets for task A and task B, respectively, released for the 2006 ECML-PKDD Discovery Challenge [6]. Both datasets contain a generic set of e-mails for training and several sets of e-mails belonging to individual users for testing. All sets have an equal number of spam and non-spam e-mails. Distribution shift exists between the training and test sets and among the test sets.

The composition of the training sets is as follows: 50% spam e-mails sent by blacklisted servers of the Spamhaus project, 40% non-spam e-mails from the SpamAssassin corpus, and 10% non-spam e-mails from about 100 different subscribed English and German newsletters. The composition of the test sets (users' e-mails) is more varied with non-spam e-mails from distinct Enron employees in the Enron corpus and spam e-mails from various sources.

The number of e-mails in the training and test sets is much larger in ECML-A than in ECML-B. Furthermore, in ECML-B, the number of e-mails in the training set is less than the number of e-mails in the test sets. As such, the ECML-B dataset represents a more challenging personalized spam filtering problem than that captured in the ECML-A dataset.

E-mails in these datasets are represented by a list of tokens and their counts within the e-mail content (including the Subject field in the headers). This representation is based on a common dictionary (vocabulary) for all e-mails in ECML-A and a common dictionary for all e-mails in ECML-B. For more details regarding these datasets, including preprocessing steps, refer to Bickel [6].

5.1.2 ECUE-1 and ECUE-2 datasets

The ECUE-1 and ECUE-2 datasets are derived from the ECUE concept drift 1 and 2 datasets, respectively [19]. Each dataset is a collection of e-mails received by one specific user over the period of one year. The order in which the e-mails are received is preserved in these datasets. The training sets contain 1,000 e-mails (500 spam and 500 non-spam) received during the first 3 months. The test sets contain 2,000 e-mails (1,000 spam and 1,000 non-spam) randomly sampled from the e-mails received during the last 9 months. As such, concept drift exists from training to test sets in these datasets.

These datasets are not preprocessed for stop words, stemming, or lemmatization. E-mail attachments are removed before parsing but any HTML text present in the e-mails is included in the tokenization. A selection of header fields, including the Subject, To and From, is also included in the tokenization. These datasets contain three types of features: (a) word features, (b) letter or single character features, and (c) structural features, e.g. the proportion of uppercase or lowercase characters. For further details, refer to Bickel [19].

5.1.3 PU1 and PU2 datasets

The PU1 and PU2 datasets contain e-mails received by a particular user [3]. The order in which the e-mails are received is not preserved in these datasets. Moreover, only the earliest five non-spam e-mails from each sender are retained in the datasets. Attachments, HTML tags, and duplicate spam e-mails received on the same day are removed before preprocessing. The PU1 dataset is available in four versions depending on the preprocessing performed. We use the version with stop words removed. The PU2 dataset is available in the bare form only, i.e. without stop word removal and lemmatization.

The PU1 dataset contains 481 spam and 618 non-spam e-mails available in 10 partitions or folders. We select the first 7 folders for training and the last 3 for testing. Within the training and test sets, we retain 336 and 145 e-mails, respectively, of each class for our evaluation. The PU2 dataset is also available in 10 folders with the first 7 folders selected for training and the last 3 for testing. There are 497 training e-mails (399 are non-spam) and 213 test e-mails (171 are non-spam). For this dataset, we do not sample the e-mails to achieve even proportions of spam and non-spam e-mails because doing so produces very small training and test sets.

5.2 Algorithms

We compare the performance of our algorithms with Naive Bayes (NB), Maximum Entropy (ME), Balanced Winnow (BW), and Support Vector Machine (SVM). Two of these are generative (NB and ME) and two are discriminative (SVM and BW) in nature. For NB, ME, and BW, we use the implementation provided by the Mallet toolkit [57]. For SVM, we use the implementation provided by SVM^{Light} [34].

E-mails are represented as term count vectors for the NB, ME, BW, and SVM classifiers. The default algorithm settings provided by Mallet are adopted for NB, ME, and BW. The SVM (using SVM^{Light}) is tuned for each dataset by evaluating its performance on a validation set that is a 25% holdout of the training set. The SVM^{Light} parameter C that controls the trade-off between classification error and margin width is tuned for each dataset and each evaluation. Similarly, we evaluate the performance of SVM with both linear and non-linear kernels and find the linear kernel to be superior. This observation is consistent with that reported in the literature [23,51,71]. We perform e-mail length normalization using $L2$ (Euclidean) norm.

Table 3 Global spam filtering results for ECML-A dataset

User	DTWC		NB		ME		BW		SVM	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
1	**91.00**	**96.35**	81.24	82.71	62.20	77.32	61.00	66.01	64.56	72.55
2	**92.36**	**97.37**	83.80	86.40	68.16	81.19	64.76	69.46	70.08	79.24
3	87.52	**94.59**	**87.88**	94.40	78.92	91.13	73.44	78.52	80.44	90.59
Avg	**90.29**	**96.10**	84.30	87.83	69.76	83.21	66.40	71.33	71.69	80.79

Bold values signify maximum or best performance

This improves performance slightly from the unnormalized case, as observed by others as well [23,60,71]. We keep the remaining parameters of SVM^{Light} at default values.

Semi-supervised versions of the comparison algorithms are obtained by adopting the naive SSL approach (see Sect. 4.4). The semi-supervised versions of these algorithms are identified as NB-SSL, ME-SSL, BW-SSL, and SVM-SSL.

6 Results and discussion

We evaluate DTWC, PSSF1, and PSSF2 under several settings: global spam filtering, personalized spam filtering, varying distribution shift, gray e-mails, generalization to unseen data, and scalability. We also present statistical significance test results for the tested algorithms on global and personalized spam filtering.

6.1 Global spam filtering

In this section, we evaluate the performance of DTWC, NB, ME, BW, and SVM for global spam filtering. Each algorithm is trained on the training data and evaluated on the test set(s). As such, this represents a supervised learning setting where test data are not available to the algorithms during training. For this and subsequent evaluations (unless specified otherwise), the term selection parameter t' of DTWC/PSSF is kept equal to zero (or, equivalently, t is kept equal to one). The effect of varying the term selection parameter is discussed in Sect. 6.7.

Tables 3, 4, 5, and 6 show the percent accuracy and AUC values of the algorithms on ECML-A, ECML-B, ECUE (ECUE-1 and ECUE-2), and PU (PU1 and PU2) datasets, respectively. Out of the 44 results (percent accuracy and AUC values), DTWC outperforms the rest of the algorithms in 29 results. The next best algorithm is ME with 13 out of 44 winning results, all of them on the ECML-B dataset.

DTWC's performance is significantly better than the other algorithms on ECML-A, ECUE-1, ECUE-2, PU1, and PU2 datasets. The ECML-A dataset involves a distribution shift from training to test sets, where the test sets correspond to e-mails received by different users. For this dataset, the average percent accuracy and average AUC value of DTWC are 5.99 and 8.27% higher, respectively, than those of the next best filter (NB). The ECUE-1 and ECUE-2 datasets also involve a distribution shift from training to test sets in the form of concept drift. For these datasets, the percent accuracy and AUC values of DTWC are at least 3.95 and 6.14% higher, respectively, than the next best results. The PU1 and PU2 datasets involve no distribution shift as their training and test sets are drawn randomly from

Table 4 Global spam filtering results for ECML-B dataset

User	DTWC		NB		ME		BW		SVM	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
1	**76.00**	72.16	57.00	50.24	67.75	**87.88**	39.75	35.63	49.75	57.8
2	**74.50**	73.52	54.50	45.22	67.75	**82.69**	26.50	21.14	44.25	51.73
3	**84.50**	**91.24**	67.00	69.77	72.50	81.41	49.00	46.95	63.75	69.20
4	**93.50**	**98.14**	73.75	74.52	67.25	81.26	46.50	52.39	53.00	65.33
5	**74.25**	82.11	68.75	79.26	71.25	83.39	60.00	67.90	64.75	**86.28**
6	72.25	80.71	63.25	64.49	**74.50**	**81.83**	62.25	69.89	68.00	81.78
7	**75.25**	72.42	62.25	56.43	59.00	**73.01**	60.00	60.89	54.25	64.11
8	74.50	86.78	58.50	65.27	**75.75**	**86.98**	40.75	37.87	65.50	70.53
9	78.75	79.62	61.50	58.97	**79.25**	**89.98**	36.00	33.46	62.75	68.28
10	**80.00**	75.20	58.25	52.56	67.25	**87.18**	36.00	32.80	46.75	56.57
11	**80.25**	**85.82**	62.75	70.71	70.00	83.26	57.75	65.65	65.75	77.29
12	**79.75**	**86.69**	69.25	74.99	67.25	81.13	60.25	67.75	64.75	77.94
13	**88.75**	**91.28**	70.25	68.70	67.50	80.70	55.00	54.13	66.25	80.19
14	64.75	83.12	58.50	69.62	**78.25**	**86.16**	59.75	66.58	71.75	82.19
15	**73.25**	75.49	65.75	62.64	62.00	**81.92**	57.50	60.42	54.25	67.93
Avg	**78.01**	82.29	63.41	64.22	69.81	**83.52**	49.80	51.56	59.70	70.47

Bold values signify maximum or best performance

Table 5 Global spam filtering results for ECUE-1 and ECUE-2 datasets

Dataset	DTWC		NB		ME		BW		SVM	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
ECUE-1	**92.20**	**96.88**	50.05	50.05	78.3	86.63	83.05	90.74	83.30	89.84
ECUE-2	**83.45**	**98.24**	50.00	50.00	79.5	84.54	77.50	83.95	76.95	85.62

Bold values signify maximum or best performance

Table 6 Global spam filtering results for PU1 and PU2 datasets

Dataset	DTWC		NB		ME		BW		SVM	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
PU1	**98.27**	**99.65**	96.55	97.70	96.89	99.48	97.24	99.27	96.21	99.51
PU2	**97.18**	**97.06**	87.32	71.24	94.36	96.65	90.61	89.34	88.26	93.60

Bold values signify maximum or best performance

e-mails received by a single user. For these datasets too, DTWC outperforms the others in both percent accuracy and AUC values.

The ECML-B dataset represents a more challenging spam filtering problem where (1) distribution shift exists between training and test sets and (2) the training and test sets are much smaller in size (100 and 400 e-mails, respectively). On this dataset, DTWC comes out on top in 16 out of 30 results. The next best algorithm is ME with 13 out of 30 winning results.

Table 7 Personalized spam filtering results for ECML-A dataset

User	PSSF1		PSSF2		NB-SSL		ME-SSL		BW-SSL		SVM-SSL	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
1	87.84	98.60	**96.68**	**98.96**	81.84	83.71	81.48	81.06	76.28	81.22	64.56	70.66
2	89.92	98.78	**97.24**	**99.60**	80.96	82.54	85.76	86.65	77.80	82.71	70.08	79.30
3	**97.00**	**99.43**	93.36	98.92	86.36	88.14	85.16	86.02	81.24	85.42	80.44	90.78
Avg	91.58	98.94	**95.80**	**99.16**	83.05	84.79	84.13	84.57	78.44	83.11	71.69	80.24

Bold values signify maximum or best performance

This spam filtering setting, however, is unlikely to occur in practice because large quantities of generic labeled e-mails are readily available for training purposes. When the size of the training set is larger, as for ECML-A dataset, DTWC outperforms the other algorithms by a wide margin.

It is worth noting that SVM performs poorly on datasets involving distribution shift. We study the impact of varying distribution shift on algorithm performance in a subsequent section.

These results demonstrate the robustness of our algorithm and its suitability as a global spam filter at the service-side. An extensive evaluation of DTWC for supervised text classification is given in Junejo and Karim [38], where it is shown that DTWC performs accurately on other text datasets as well.

6.2 Personalized spam filtering

In this section, we evaluate the performance of PSSF, NB-SSL, ME-SSL, BW-SSL, and SVM-SSL for personalized spam filtering. In this setting, the filter that is learned on the training data is adapted for each test set in a semi-supervised fashion. Performance results are reported for the personalized filters on their respective test sets.

Tables 7, 8, 9, and 10 show the personalized spam filtering results of the algorithms on ECML-A, ECML-B, ECUE (ECUE-1 and ECUE-2), and PU (PU1 and PU2) datasets, respectively. The results demonstrate the effectiveness of personalization for datasets with distribution shift (ECML-A, ECML-B, and ECUE). For ECML-A dataset, the performance of both PSSF1 and PSSF2 improves over that of the global spam filter (DTWC), with PSSF2 outperforming the others in 4 results and PSSF1 outperforming the others in the remaining two results. The performance of the other algorithms (except SVM-SSL) also improves from that of their supervised versions. For ECML-B dataset, PSSF1 has the highest average percent AUC value while BW-SSL tops the others in average percent accuracy. For ECUE-1 dataset, PSSF1 has the best results while ME-SSL outperforms the others on ECUE-2 dataset. ME-SSL and BW-SSL have the wining results on PU1 and PU2 datasets, respectively. In all, PSSF1/PSSF2 outperforms the others in 22 out of 44 results. In most cases, where our algorithm does not win, the difference in performance with the winner is minor, e.g. the average accuracy of PSSF1 on the ECML-B dataset is less 1.5% less than the winner, whereas for NB-SSL and SVM-SSL, they are almost 27 and 30% behind, respectively.

The most surprising results are those of BW-SSL on ECML-B dataset. The performance of BW jumped by about 40% (on both average accuracy and AUC value) after semi-supervised learning. Balanced winnow learns a hyperplane by updating its parameters whenever mistakes in classification are made. In theory, the hyperplane learned by BW from the training

Table 8 Personalized spam filtering results for ECML-B dataset

User	PSSF1		PSSF2		NB-SSL		ME-SSL		BW-SSL		SVM-SSL	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
1	**91.25**	**94.15**	76.75	77.79	57.00	56.28	77.75	85.94	83.75	90.88	49.75	53.42
2	**93.75**	96.50	78.50	79.54	53.00	54.31	82.75	91.53	93.50	**97.31**	44.25	50.83
3	93.50	96.29	86.00	93.73	64.00	64.22	92.00	95.83	**94.75**	**98.21**	63.75	70.59
4	**98.50**	**99.22**	98.25	98.91	69.50	70.23	80.25	87.27	90.75	93.74	53.00	60.37
5	**82.75**	**93.79**	74.75	93.12	67.75	68.85	79.25	83.66	75.75	71.11	64.75	81.29
6	76.00	78.65	71.25	78.39	63.50	64.54	**92.00**	**95.05**	87.75	86.30	68.00	77.81
7	80.25	**92.72**	76.25	70.79	63.50	63.74	83.50	91.46	**91.00**	88.78	54.25	60.66
8	91.50	95.46	80.50	91.62	59.00	60.23	**92.25**	**97.52**	87.50	91.31	65.50	72.16
9	92.00	**99.32**	80.00	92.74	60.75	61.48	89.75	94.29	**92.25**	94.37	62.75	71.15
10	82.75	**98.12**	80.00	83.55	56.75	57.39	77.25	86.42	**94.25**	97.84	46.75	49.20
11	90.75	94.08	80.50	88.91	62.50	64.31	88.00	93.80	**94.75**	**96.19**	65.75	74.80
12	85.75	91.26	79.25	86.82	66.75	66.89	82.25	86.65	**92.75**	**94.63**	64.75	73.65
13	**97.00**	**98.85**	90.50	94.98	68.50	68.54	89.50	96.12	92.25	96.55	66.25	79.00
14	75.50	88.21	65.00	84.46	57.75	58.79	**92.25**	**95.86**	86.50	86.06	71.75	82.56
15	**89.00**	**90.52**	77.00	79.31	66.50	67.96	74.50	82.36	84.75	78.53	54.25	58.17
Avg	88.01	**93.81**	79.63	86.31	62.45	63.18	84.88	90.91	**89.48**	90.78	59.70	67.71

Bold values signify maximum or best performance

Table 9 Personalized spam filtering results for ECUE-1 and ECUE-2 datasets

	PSSF1		PSSF2		NB-SSL		ME-SSL		BW-SSL		SVM-SSL	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
ECUE-1	**93.60**	**97.15**	93.35	97.12	50.00	50.00	89.25	88.52	81.00	85.80	83.30	89.36
ECUE-2	84.7	96.55	83.45	96.68	50.00	50.00	**95.20**	**98.61**	64.25	92.18	76.95	85.20

Bold values signify maximum or best performance

Table 10 Personalized spam filtering results for PU1 and PU2 datasets

	PSSF1		PSSF2		NB-SSL		ME-SSL		BW-SSL		SVM-SSL	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
PU1	98.62	99.28	98.27	99.28	97.93	97.87	**99.65**	**99.89**	97.58	97.37	96.21	98.66
PU2	96.24	97.43	97.18	96.46	82.62	60.48	99.06	99.63	**99.53**	**99.88**	88.26	93.00

Bold values signify maximum or best performance

data should not change during naive SSL. However, in practice, significant improvement is seen which can be attributed to the poor convergence characteristics (and lack of robustness) of the BW learning algorithm. This is supported by the observation that SVM and SVM-SSL perform almost identically.

We compare PSSF's performance with four published results on ECML-A and ECML-B datasets in Table 11. Two of these results [16,39] are winning performances of the 2006

Table 11 Comparison with other published personalized spam filtering results (all numbers are average percent AUC values)

Algorithm	ECML-A	ECML-B
PSSF1	98.94	93.81
PSSF2	**99.16**	86.31
Junejo et al. [39]	98.75	–
Kyriakopoulou and Kalamboukis [48]	97.31	**95.08**
Cormack [16]	93.00	94.90
Cheng and Li [13]	93.33	–

Bold values signify maximum or best performance

ECML-PKDD Discovery Challenge [6], while Kyriakopoulou and Kalamboukis [48] present the best performance on ECML-B dataset later. PSSF outperforms all algorithms on ECML-A dataset and lags the best performance by 1.27% on ECML-B dataset. Junejo et al. [39] has the previous best performance on ECML-A dataset (they do not report results for ECML-B dataset). PSSF1/PSSF2 improves on their algorithm using estimated probabilities rather than occurrence counts for defining the local model. Kyriakopoulou and Kalamboukis [48] preprocess the dataset by clustering the training data with each test set. The combined set is augmented with additional metafeatures derived from the clustering. This combined set is then learned using transductive SVM. This approach is computationally expensive and non-adaptive. Cormack [16] uses statistical compression models for predicting spam and non-spam e-mails. His approach is adaptive but the reported performances lag the best results. Cheng and Li [13] present a semi-supervised classifier ensemble approach for the personalized spam filtering problem. Their approach is also computationally expensive when compared to PSSF, and it lags in performance by more than 5% on ECML-A dataset (they do not report results for ECML-B dataset).

6.3 Statistical significance analysis

We present the results of statistical significance tests that compare the performance of DTWC/PSSF with the other classifiers on global and personalized spam filtering. Statistical analysis is necessary to ascertain the consistency of the observed performances and to reject the possibility that they are produced purely by chance. Although numerous statistical tests have been proposed, Demsar [22] recommends the nonparametric Wilcoxon signed-ranks test and the nonparametric Friedman's test (with post hoc analysis) for comparing two or more classifiers on multiple datasets. We apply these tests separately for performances measured with accuracy and AUC values. These tests are performed using the SPSS software version 19.

The Wilcoxon signed-ranks test compares the performance of two classifiers on multiple datasets by ranking their absolute difference in performance on the datasets. The null hypothesis that the observed differences in performance are insignificant is rejected when the smaller of the sum of ranks for positive and negative differences is less than a critical value for a specified confidence level. In our analysis, we perform two tests (one each for accuracy and AUC values) comparing the differences in performances of DTWC and PSSF with each of the other classifiers evaluated for global and personalized filtering, respectively (e.g. DTWC is compared with ME, PSSF is compared with ME-SSL). We find that DTWC/PSSF's performance

Table 12 Friedman's test with stepwise step-down post hoc analysis of filters' performances (accuracy)

Homogeneous Subsets						
		Subsets				
		1	2	3	4	5
Classifier[a]	BW	2.09				
	SVM	3.31	3.31			
	SVM-SSL		3.31			
	NB-SSL		4.04	4.04		
	NB		4.31	4.31		
	ME			5.04		
	DTWC				7.40	
	BW-SSL				8.00	8.00
	ME-SSL				8.36	8.36
	PSSF					9.09
Test Statistic		6.54	4.80	3.90	5.54	2.81
Sig. (2-sided test)		0.01	0.18	0.14	0.06	0.24
Adjusted Sig. (2-sided test)		0.05	0.40	0.39	0.19	0.60
[a] Each cell shows the classifier's average rank						

Table 13 Friedman's test with stepwise step-down post hoc analysis of filters' performances (AUC value)

Homogeneous Subsets							
		Subsets					
		1	2	3	4	5	6
Classifier[a]	BW	2.04					
	NB-SSL	3.02	3.02				
	NB	3.29	3.29				
	SVM-SSL		3.77	3.77			
	SVM			4.86			
	ME				6.40		
	DTWC				7.22	7.22	
	BW-SSL				7.31	7.31	7.31
	ME-SSL					7.90	7.90
	PSSF						9.13
Test Statistic		4.93	3.43	4.54	4.72	4.45	6.81
Sig. (2-sided test)		0.08	0.18	0.03	0.09	0.10	0.03
Adjusted Sig. (2-sided test)		0.25	0.48	0.15	0.28	0.31	0.10
[a] Each cell shows the classifier's average rank							

is significantly different (better) than the others on both accuracy and AUC values at the confidence level of 0.05 for accepting the null hypothesis. In fact, except for the significance level (or p value) of 0.025 obtained when DTWC/PSSF is compared with ME/ME-SSL on AUC values, all other p values are less than 0.001. This result provides evidence that the difference in performance observed between our algorithms and the others on both global and personalized filtering is statistically significant.

The Friedman's test evaluates multiple classifiers on multiple datasets by comparing the average ranks of the classifiers on the datasets (higher average ranks are considered better). The null hypothesis that the classifiers are equivalent is rejected when the Friedman statistic is greater than a critical value at a specified confidence level. In our analysis, we perform two Friedman's tests (one each for accuracy and AUC values) comparing 10 classifiers (DTWC, NB, ME, BW, SVM, PSSF, NB-SSL, ME-SSL, BW-SSL, SVM-SSL) on 22 datasets (3 ECML-A, 15 ECML-B, 2 PU, and 2 ECUE datasets). We use the Friedman's test with stepwise step-down post hoc analysis, which finds subsets of classifiers that are statistically equivalent (homogeneous subsets). Tables 12 and 13 show the results of this test for

the accuracy and AUC values, respectively. These results show that DTWC's performance in accuracy is statistically better than all other classifiers evaluated for global filtering as it is placed in a subset that contains none of the other global filters. When performance is measured using AUC values, DTWC is placed in the same subset with ME, although it still maintains a higher average rank. For personalized filtering, PSSF is grouped with BW-SSL and ME-SSL. Nonetheless, PSSF maintains a higher average rank and the significance level of this subset under AUC is not very high (0.10). Moreover, PSSF appears in only one subset, unlike the other two classifiers.

These statistical analyses provide evidence for the overall superiority of our algorithms on global and personalized spam filtering. They also validate the robustness of our algorithms across different spam filtering settings.

6.4 Varying distribution shift

In this section, we evaluate the performance of DTWC/PSSF, NB, ME, BW, and SVM under varying distribution shift between training and test data. This evaluation is performed on ECML-A dataset by swapping varying numbers of e-mails between training and test (user) data. By increasing the number of e-mails swapped, the distribution shift between training and test data is reduced as well. To illustrate the evaluation procedure, suppose 100 randomly selected e-mails from user 1 are moved to the training data and 100 randomly selected e-mails from the training data are moved to user 1 e-mails. The filters are then trained and tested using the modified training and test data. This procedure is repeated for each user and for different numbers of e-mails swapped.

To quantify the distribution shift between training and test data, we adapt the KL divergence (KLD) and total variation distance (TVD) as follows:

$$D_{KL}(L,U) = \frac{1}{2T}\left[\sum_j p_L(x_j|+)\log\frac{p_L(x_j|+)}{p_U(x_j|+)} + \sum_j p_L(x_j|-)\log\frac{p_L(x_j|-)}{p_U(x_j|-)}\right]$$

$$D_{TV}(L,U) = \frac{1}{2T}\left[\sum_j |p_L(x_j|+) - p_U(x_j|+)| + \sum_j |p_L(x_j|-) - p_U(x_j|-)|\right],$$

where $D_{KL}(\cdot,\cdot)$ and $D_{TV}(\cdot,\cdot)$ denote the adapted KL divergence and total variation distance, respectively, L and U identify training and test data, respectively, and T is the total number of distinct terms in the training and test data. The quantity D_{KL} (D_{TV}) is computed as the average of the KL divergence (total variation distance) for spam and non-spam conditional distributions normalized by T. The normalization ensures that these quantities range from 0 to 1 for all training–test data pairs irrespective of the numbers of terms in them. Table 14 shows the average performance for the three test sets in ECML-A dataset. It is seen from this table that as the number of e-mails swapped between training and test sets (given in the first column of Table 14) increases, the distribution shift between the sets decreases, as quantified by the values of D_{KL} and D_{TV}. More interestingly, it is observed that as the distribution shift decreases the performance gap between DTWC/PSSF and the other algorithms narrows down. The performance of all the algorithms improves with the decrease in distribution shift, especially for ME, BW, and SVM. For example, the average accuracy of ME jumps up by 28.21% from the case when no e-mails are swapped to the case when 1,500 e-mails are swapped. Our supervised spam filter, DTWC, comprehensively outperforms the other algorithms when distribution shift is large, while its performance compares well with the others at low distribution shift. Another observation from this evaluation

Table 14 Performance under varying distribution shift

#	D_{KL} ($\times 10^{-7}$)	D_{TV} ($\times 10^{-7}$)	DTWC		PSSF1		PSSF2		NB		ME		BW		SVM	
			Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
0	717	444	90.29	96.10	91.58	98.94	**95.80**	**99.16**	84.30	87.83	69.76	83.21	66.40	71.33	71.69	80.79
100	517	384	93.30	98.04	92.86	98.77	**96.01**	**99.11**	92.08	95.84	87.90	94.53	75.53	82.73	88.12	94.96
250	477	336	94.68	98.72	92.40	98.39	**96.44**	**99.02**	93.70	97.05	92.80	97.36	82.13	88.87	92.08	97.70
500	346	259	**96.60**	**99.38**	94.92	98.41	96.56	99.14	95.57	98.01	96.18	98.68	87.29	92.91	95.70	99.17
1500	157	127	97.63	99.55	96.41	98.56	96.94	98.97	96.39	97.82	**97.97**	99.01	95.08	98.11	97.30	**99.68**
Avg			94.5	98.35	93.63	98.61	**96.35**	**99.08**	92.40	95.31	88.92	94.55	81.28	86.79	88.97	94.46

Average percent accuracy and AUC values are given for ECML-A dataset
Bold values signify maximum or best performance

is that as the distribution shift decreases the benefit of semi-supervised learning diminishes.

6.5 Gray e-mails

Gray e-mails were introduced and defined in Sect. 2. They present a particulary challenging problem for spam filters because similar e-mails are labeled differently by different users. To evaluate the performance of DTWC/PSSF on gray e-mails, we devise an experiment based on e-mails for user 1 and user 2 in ECML-A dataset. Consider a graph consisting of two sets of vertices corresponding to the e-mails for user 1 and user 2. Edges in the graph exist between vertices in the two sets if the corresponding e-mails have a cosine similarity greater than a specified threshold, s. The cosine similarity between two e-mails $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined as

$$sim(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i^T \mathbf{x}_j}{\|\mathbf{x}_i\| \|\mathbf{x}_j\|},$$

where $\|\mathbf{x}\| = \sqrt{\mathbf{x}^T \mathbf{x}}$. Given this graphical representation and the true labels of e-mails, an e-mail belonging to user 1 (user 2) is a gray e-mail if an edge exists from its vertex to one or more vertices belonging to user 2 (user 1) *and* the labels of the two e-mails are different. This procedure identifies the set of gray e-mails for user 1 and the set of gray e-mails for user 2. We evaluate the performance of DTWC/PSSF on these sets of e-mails by reporting the number of e-mails that are incorrectly labeled by the algorithms.

The performance of DTWC and PSSF2 on gray e-mails is reported in Table 15. For each user, the table shows the total number of gray e-mails according to the specified similarity threshold s and the number of gray e-mails incorrectly classified by DTWC and PSSF2. The results show that personalization performed by PSSF2 results in a significant reduction in errors on gray e-mails. This is attributable to the adaptation of the local and global models on the user e-mails. The most significant improvement in classification on gray e-mails occurs when the global discriminative model is adapted (therefore results of PSSF1 are not shown in the table). This observation is consistent with the fact that gray e-mails result from concept shift, which is better tracked by adapting the discriminative model.

Table 15 Performance on gray e-mails identified from user 1 and user 2 e-mails in ECML-A dataset. For DTWC and PSSF2, the table gives the number of gray e-mails that are misclassified

s	User 1			User 2		
	No. of GE	DTWC	PSSF2	No. of GE	DTWC	PSSF2
0.5	1,471	130	65	1,443	132	46
0.6	1,016	79	39	1,109	97	40
0.7	622	46	34	588	45	22
0.8	174	12	10	216	17	6
0.9	34	3	1	71	6	1

s similarity threshold, *GE* gray e-mails

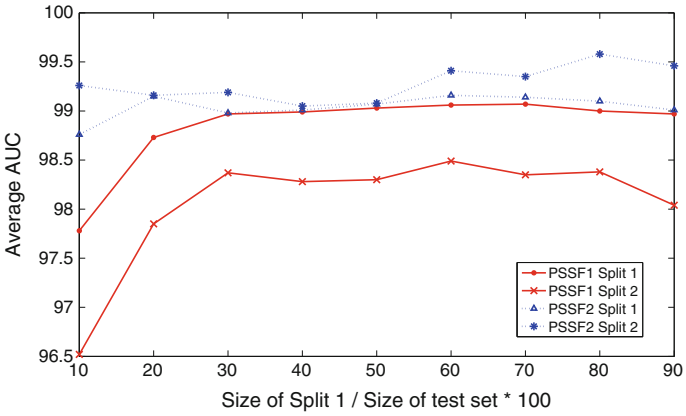

Fig. 4 Generalization performance on ECML-A dataset

6.6 Generalization to unseen data

In the previous subsections, we presented results of PSSF1 and PSSF2 on the test sets where the unlabeled e-mails in the test sets were utilized during learning. However, in practice, once a personalized spam filter is learned using labeled and unlabeled e-mails, it is applied to unseen e-mails for a while before it is adapted again. The performance over these unseen e-mails represents the generalization performance of the filter. We evaluate the generalization property of PSSF by splitting the test set into two: split 1 is used during semi-supervised learning and split 2 contains the unseen e-mails for testing. The generalization performance of PSSF1 and PSSF2 on ECML-A dataset is shown in Fig. 4. In general, the difference between the average percent AUC values for split 1 and split 2 is less than 1% for PSSF1 and less than 0.5% for PSSF2. Furthermore, the decrease in average AUC value with decrease in size of split 1 (increase in size of split 2) is graceful. PSSF2, in particular, exhibits excellent generalization performance. It is able to learn the personalized filter for each user from a small number of e-mails for the user. This characteristic of PSSF2 stems from the realignment of the decision line after considering the user's e-mails. These results also demonstrate the robustness of PSSF.

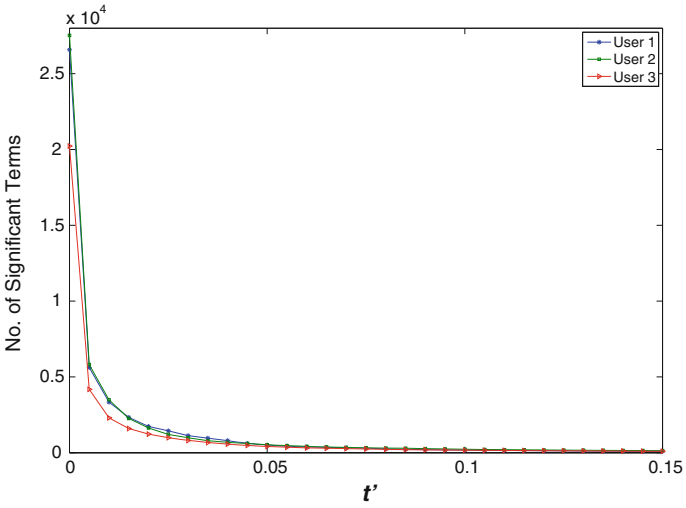

Fig. 5 Number of significant terms versus term selection parameter t' for PSSF1/PSSF2 on ECML-A dataset

Table 16 Scalability of PSSF: impact of filter size on performance. The results are averages for the three test sets in ECML-A dataset

t'	Terms	PSSF1		PSSF2	
		Acc	AUC	Acc	AUC
0.00	24777	91.58	98.94	**95.80**	**99.16**
0.05	484	**93.90**	98.80	93.88	**98.81**
0.12	146.66	94.00	98.75	**96.06**	**98.88**
0.20	66	92.17	97.77	**93.80**	**98.03**
0.30	26.33	88.88	96.00	**89.44**	**96.84**
0.35	14.66	**85.92**	93.75	84.33	**94.38**
0.40	9.33	**82.85**	90.96	81.42	**91.69**
0.41	8	**82.68**	87.45	81.11	**88.07**

Bold values signify maximum or best performance

6.7 Scalability

Robustness and scalability are essential for personalized service-side spam filtering [46]. To implement PSSF on the service-side for a given user, the local and global model parameters must be resident in memory. There are only two global model parameters; the slope and bias of the line. The local model parameters are the discriminative term weights for the significant spam and non-spam terms. The number of significant spam and non-spam terms, which defines the size of the filter, is controlled by the term selection parameter t'. Figure 5 shows that by increasing the value of t' slightly from zero, the number of significant terms drops sharply. This reduction in filter size does not degrade filtering performance significantly (and sometimes it even improves performance) as only less discriminating terms are removed from the model.

Table 16 shows the term selection parameter, average filter size (as average number of significant terms), and average percent accuracy and AUC values of PSSF1 and PSSF2 on

ECML-A dataset. It is seen that even when the average filter size is reduced by about a factor of 165 (from 24,777 to 146.66 terms), the average AUC value for PSSF1 decreases very slightly (less than 0.25%), and this value is still greater than that reported by Kyriakopoulou and Kalamboukis [48], Cormack [16], Cheng and Li [13] (see Table 11). Moreover, with an average filter size of only 66 terms, PSSF performs remarkably well with average AUC values greater than 97%. It is worth pointing out that even when $t' = 0$, the average number of significant terms is much less than the vocabulary size of 41,675 for this dataset. This is because terms that provide no discrimination information for either spam or non-spam classification are removed from the model.

The average filter size is directly related to the scalability of the filter—the smaller the size the greater the number of users that can be served with the same computing resources. For example, when the average filter size is 8 terms, PSSF1 can serve approximately 30,000 users with 1 MB of memory (assuming 4 bytes per discriminative term weight). Interestingly, the average performance of PSSF1 with this filter size is similar to that of a single global NB filter (see Table 3). However, the size of the NB filter will be over 83,350 ($41,675 \times 2$) when compared to only 24 (8×3) for PSSF1 (for three users).

It is worth emphasizing from a scalability perspective that DTWC/PSSF requires a single pass over the labeled data to learn the local model parameters and its global model parameters are obtained by solving a straightforward optimization problem. Also, the naive SSL approach that we use for personalization decouples the adaptation from the training data and as such can be performed on the client-side in addition to the service-side.

7 Conclusion

E-mail spam continues to be a menace for users and e-mail service providers (ESPs) with content-based e-mail spam filters providing a popular defense against spam. Spam filtering solutions deployed at the ESP's side can either be a single global filter for all users or multiple filters each personalized for a specific user. The motivation for having personalized filters is to cater for the differing spam and non-spam preferences of users. However, personalized spam filtering solutions need to be scalable before they are practically useful for large ESPs. Similarly, a global filtering solution needs to be robust to differing preferences of users for its practical usefulness.

We address the above dilemma by presenting a scalable and robust spam filter appropriate for global and personalized spam filtering. Based on local and global modeling of discrimination, our filter learns a discriminant function in a two-dimensional feature space in which spam and non-spam e-mails are well separated. The two dimensions of the feature space correspond to the linear opinion pool or ensemble average of discrimination information (opinions) provided by significant spam and non-spam terms. This feature space representation makes our filter robust to distribution shift. In addition to a supervised version, named DTWC, we also present two semi-supervised versions, named PSSF1 and PSSF2, that are suitable for personalized spam filtering. PSSF1/PSSF2 can be adapted to the distribution of e-mails received by individual users to improve filtering performance.

We evaluate DTWC/PSSF on six e-mail datasets and compare its performance with four classifiers. DTWC/PSSF outperforms the other algorithms in 51 out of 88 results (accuracy and AUC) in global and personalized spam filtering. Statistical tests show that DTWC/PSSF perform significantly better than the other algorithms. In particular, DTWC/PSSF performs remarkably well when distribution shift is significant between training and test data, which is common in e-mail systems. We also evaluate our algorithms under varying distribution shift,

on gray e-mails, on unseen e-mails, and under varying filter size. Our personalized spam filter, PSSF, is shown to scale well for personalized service-side spam filtering.

In this paper, we also discuss the nature of the spam filtering problem and the challenges to effective global and personalized spam filtering. We define key characteristics of e-mail classification such as distribution shift and gray e-mails and relate them to machine learning problem settings.

Our algorithms are based on an intuitive statistical understanding of the spam filtering problem. Although they have been shown to be reliable and accurate, they do have some limitations. Currently, we consider each term independently of the others in the local discrimination model, which disregards any correlation between terms. One way of addressing this is to discover combinations of terms (or termsets) with high discrimination power and use them as weighted features in the global model. We faced some convergence problems while learning the global model parameters for small datasets. This issue can be resolved using a regularized or large margin classifier for the global model. In this work, we use the naive SSL approach for personalization. Although this approach has some merits for our problem setting, as discussed in Sect. 4.4, it discards the local model learned from the training data after labeling the test data before building a new model from the test data alone. It might be worthwhile to investigate approaches that consider knowledge from both the training and the test data as a whole. Presently, our personalized spam filter cannot take advantage of some labeled e-mails of each user. In the future, in addition to addressing the above points, we would also like to study the classifier framework introduced in this paper by considering other discriminative term weighting and opinion pooling strategies.

We believe that service-side spam filtering presents significant learning and implementation challenges for researchers. Some research directions include characterization of users' spam/non-spam preferences, incorporation of users' characterizations in learning models, and development and evaluation of hybrid global/local/service-side/user-side solutions.

Acknowledgments The first author gratefully acknowledges support from Lahore University of Management Sciences (LUMS) and Higher Education Commission (HEC) of Pakistan for this work. We would also like to thank the anonymous reviewers for their helpful feedback.

References

1. Agirre E, de Lacalle OL (2008) On robustness and domain adaptation using SVD for word sense disambiguation. In: COLING-08: Proceedings of the 22nd international conference on computational Linguistics. Association for Computational Linguistics, pp 17–24
2. Alpaydin E (2004) Introduction to machine learning. MIT Press, Cambridge
3. Androutsopoulos I, Koutsias J, Chandrinos KV, Spyropoulos CD (2000) An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages. In: SIGIR-00: Proceedings of the 23rd conference on research and development in information retrieval. ACM, New York, pp 160–167
4. Atkins S (2003) Size and cost of the problem. In: IETF-03: in 56th meeting of the Internet engineering task force. San Francisco
5. Ben-David S, Blitzer J, Crammer K, Pereira F (2007) Analysis of representations for domain adaptation. In:NIPS-07: Advances in neural information processing systems. MIT Press, Cambridge, pp 137–144
6. Bickel S (2006) ECML-PKDD discovery challenge 2006 overview. In: Proceedings of ECML-PKDD discovery challenge workshop, pp 1–9
7. Bickel S, Scheffer T (2006) Dirichlet-enhanced spam filtering based on biased samples. In: NIPS-06: Advances in neural information processing systems. MIT Press, Cambridge, pp 161–168
8. Bickel S, Brückner M, Scheffer T (2009) Discriminative learning under covariate shift. J Mach Learn Res 10:2137–2155

9. Bigi B (2003) Using Kullback–Leibler distance for text categorization. In: ECIR:03 Proceedings of 25th European conference on information retrieval research. Springer, Berlin, pp 305–319
10. Blitzer J, McDonald R, Pereira F (2006) Domain adaptation with structural correspondence learning. In: EMNLP-06: Proceedings of 11th conference on empirical methods in natural language processing. Association for Computational Linguistics, pp 120–128
11. Bringmann B, Nijssen S, Zimmermann A (2009) Pattern-based classification: a unifying perspective. In: Proceedings of ECML-PKDD workshop on from local patterns to global models, pp 36–50
12. Chang M, Yih W, McCann R (2008) Personalized spam filtering for gray mail. In: CEAS-08: Proceedings of 5th conference on email and anti-spam
13. Cheng V, Li CH (2006) Personalized spam filtering with semi-supervised classifier ensemble. In: WI-06: Proceedings of the IEEE/WIC/ACM international conference on Web intelligence. IEEE Computer Society, pp 195–201
14. Chung YM, Lee JY (2001) A corpus-based approach to comparative evaluation of statistical term association measures. J Am Soc Inf Sci Technol 52(4):283–296
15. Cormack GV (2007) Email spam filtering: a systematic review. Found Trends Inf Retr 1(4):335–455
16. Cormack GV (2006) Harnessing unlabeled examples through application of dynamic markov modeling. In: Proceedings of ECML-PKDD discovery challenge workshop, pp 10–15
17. Cortes C, Mohri M (2004) AUC optimization vs. error rate minimization. In: NIPS-04: advances in neural information processing systems. MIT Press, Cambridge
18. Dagan I, Karov Y, Roth D (1997) Mistake driven learning in text categorization. In: EMNLP-97: Proceedings of 2nd conference on empirical methods in natural language processing. Association for Computational Linguistics, pp 55–63
19. Delany SJ, Cunningham P, Coyle L (2005a) An assessment of case-based reasoning for spam filtering. J Artif Intell Rev 24(3–4):359–378
20. Delany SJ, Cunningham P, Tsymbal A, Coyle L (2005b) A case-based technique for tracing concept drift in spam filtering. Knowl Based Syst 18:187–195
21. Delany SJ, Cunningham P, Tsymbal A (2006) A comparison of ensemble and case-base maintenance techniques for handling concept drift in spam filtering. In: FLAIRS-06: Proceedings of the 19th international Florida Artificial Intelligence Research Society Conference. AAAI Press, pp 340–345
22. Demsar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30
23. Druck G, Pal C, McCallum A, Zhu X (2007) Semi-supervised classification with hybrid generative/discriminative methods. In: KDD-07: Proceedings of 13th conference on knowledge discovery and data mining. ACM, New York, pp 280–289
24. Fawcett T (2003) "In vivo" spam filtering: a challenge problem for kdd. SIGKDD Explor Newsl 5(2): 140–148
25. Forman G (2003) An extensive empirical study of feature selection metrics for text classification. J Mach Learn Res 3:1289–1305
26. Goodman J, Gormack GV, Heckerman D (2007) Spam and the ongoing battle for the inbox. Commun ACM 50:24–33
27. Gray A, Haahr M (2004) Personalised collaborative spam filtering. In: CEAS-04: Proceedings of 1st conference on email and anti-spam
28. Hämäläïnen W (2010) StatApriori: an efficient algorithm for searching statistically significant association rules. Knowl Inf Syst 23:373–399
29. Hsieh DA, Manski CF, McFadden D (1985) Estimation of response probabilities from augmented retrospective observations. J Am Stat Assoc 80(391):651–662
30. Jaakkola TS, Haussler D (1998) Exploiting generative models in discriminative classifiers. In: NIPS 98: advances in neural information processing systems. MIT Press, Cambridge
31. Jacobs RA (1995) Methods for combining experts' probability assessments. Neural Comput 7:867–888
32. Jiang J (2007) A literature survey on domain adaptation of statistical classifiers. http://www.mysmu.edu/faculty/jingjiang/
33. Joachims T (1998) Text categorization with support vector machines: Learning with many relevant features. In: ECML-98: Proceedings on 10th European conference on machine learning. Springer, Berlin, pp 137–142
34. Joachims T (1999) Making large-scale support vector machine learning practical. MIT Press, Cambridge, pp 169–184. ISBN 0-262-19416-3
35. Joachims T (2001) A statistical learning model of text classification for support vector machines. In: SIGIR-01: Proceedings of the 24th conference on research and development in information retrieval. ACM, New York, pp 128–136

36. Juan A, Vilar D, Ney H (2007) Bridging the gap between naive Bayes and maximum entropy for text classification. In: PRIS-07: Proceedings of the 7th international workshop on pattern recognition in information systems. INSTICC Press, Setubal, pp 59–65
37. Junejo KN, Karim A (2007) PSSF: a novel statistical approach for personalized service-side spam filtering. In: WI-07: Proceedings of the IEEE/WIC/ACM international conference on web intelligence. IEEE Computer Society, pp 228–234
38. Junejo KN, Karim A (2008) A robust discriminative term weighting based linear discriminant method for text classification. In: ICDM-08: Proceedings of 8th international conference on data mining. IEEE Computer Society, pp 323–332
39. Junejo KN, Yousaf MM, Karim A (2006) A two-pass statistical approach for automatic personalized spam filtering. In: Proceedings of ECML-PKDD discovery challenge workshop, pp 16–27
40. Katakis I, Tsoumakas G, Vlahavas I (2010) Tracking recurring contexts using ensemble classifiers: an application to email filtering. Knowl Inf Syst 22(3):371–391
41. Kennedy JE, Quine MP (1989) The total variation distance between the binomial and poisson distributions. Ann Probab 17:396–400
42. Han KS, Rim HC, Myaeng SH (2006) Some techniques for naive Bayes text classification. IEEE Trans Knowl Data Eng 18(11):1457–1466
43. Knobbe A, Valkonet J (2009) Building classifiers from pattern teams. In: Proceedings of ECML-PKDD workshop on from local patterns to global models, pp 77–93
44. Knobbe A, Cremileux B, Furnkranz J, Scholz M (2008) From local patterns to global models: the lego approach to data mining. In: Proceedings of ECML-PKDD workshop on from local patterns to global models, pp 1–16
45. Kolcz A, Yih WT (2007) Raising the baseline for high-precision text classifiers. In: KDD-07: Proceedings of the 13th conference on knowledge discovery and data mining. ACM, New york, pp 400–409
46. Kolcz A, Bond M, Sargent J (2006) The challenges of service-side personalized spam filtering: scalability and beyond. In: InfoScale-06: Proceedings of the 1st international conference on scalable information systems, ACM, New york, p 21
47. Kullback S, Leibler RA (1951) On information and sufficiency. Ann Math Stat 22:79–86
48. Kyriakopoulou A, Kalamboukis T (2006) Text classification using clustering. In: Proceedings of ECML-PKDD discovery challenge workshop, pp 28–38
49. Leavitt N (2007) Vendors fight spam's sudden rise. Computer 40(3):16–19
50. LeBlanc M, Crowley J (1992) Relative risk trees for censored survival data. Biometrics 48(2):411–425
51. Lewis DD, Yang Y, Rose TG, Li F (2004) RCV1: a new benchmark collection for text categorization research. J Mach Learn Res 5:361–397
52. Li H, Li J, Wong L, Feng M, Tan YP (2005) Relative risk and odds ratio: a data mining perspective. In: PODS '05: Proceedings of the 24th ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems. ACM, New York, pp 368–377
53. Li J, Liu G, Wong L (2007) Mining statistically important equivalence classes and delta-discriminative emerging patterns. In: KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, New York, pp 430–439
54. Luenberger DG (1984) Linear and nonlinear programming. 2. Addison-Wesley, Reading
55. Malik H, Fradkin D, Moerchen F (2011) Single pass text classification by direct feature weighting. Knowl Inf Syst 28:79–98
56. Mannila H (2002) Local and global methods in data mining: basic techniques and open problems. In: Automata, languages, and programming, lecture notes in computer science, vol 2380. Springer, Berlin, pp 778–778
57. McCallum A (2002) Mallet: a machine learning for language toolkit. http://mallet.cs.umass.edu
58. Mccallum A, Pal C, Druck G, Wang X (2006) Multi-conditional learning: generative/discriminative training for clustering and classification. In: AAAI-06: Proceedings of the 21st national conference on artificial intelligence. AAAI Press, Menlo Park, pp 433–439
59. Nigam K, Lafferty J, McCallum A (1999) Using maximum entropy for text classification. In: IJCAI-99 workshop on machine learning for information filtering. pp 61–67
60. Pang B, Lee L, Vaithyanathan S (2002) Thumbs up?: sentiment classification using machine learning techniques. In: EMNLP-02: Proceedings of 7th conference on empirical methods in natural language processing. Association for Computational Linguistics, pp 79–86
61. Peng T, Zuo W, He F (2008) SVM based adaptive learning method for text classification from positive and unlabeled documents. Knowl Inf Syst 16(3):281–301
62. Raina R, Shen Y, Ng AY (2004) Classification with hybrid generative/discriminative models. In: NIPS 04: advances in neural information processing systems. MIT Press, Cambridge

63. Raina R, Battle A, Lee H, Packer B, Ng AY (2007) Self-taught learning: transfer learning from unlabeled data. In: ICML-07: Proceedings of the 24th international conference on machine learning. ACM, New York, pp 759–766
64. Sebastiani F (2002) Machine learning in automated text categorization. ACM Comput Surv 34:1–47
65. Seewald AK (2007) An evaluation of naive Bayes variants in content-based learning for spam filtering. Intell Data Anal 11(5):497–524
66. Segal R (2007) Combining global and personal anti-spam filtering. In: CEAS-07: Proceedings of 4th conference on email and anti-spam
67. Segal R, Crawford J, Kephart J, Leiba B (2004) Spamguru: an enterprise anti-spam filtering system. In: CEAS-04: Proceedings of 1st conference on email and anti-spam
68. Stern H (2008) A survey of modern spam tools. In: CEAS-08: Proceedings of 5th conference on email and anti-spam
69. Xing D, Dai W, Xue GR, Yu Y (2007) Bridged refinement for transfer learning. In: PKDD-07: Proceedings of the 11th European conference on principles and practice of knowledge discovery in databases, pp 324–335. Springer, Berlin
70. Xue JC, Weiss GM (2009) Quantification and semi-supervised classification methods for handling changes in class distribution. In: KDD-09: Proceedings of the 15th conference on knowledge discovery and data mining. ACM, New york, pp 897–906
71. Zhang L, Zhu J, Yao T (2004) An evaluation of statistical spam filtering techniques. ACM Trans Asian Lang Inf Process (TALIP) 3(4):243–269
72. Zhou ZH, Li M (2010) Semi-supervised learning by disagreement. Knowl Inf Syst 24:415–439
73. Zhu X (2008) Semi-supervised learning literature survey. http://pages.cs.wisc.edu/~jerryzhu/research/ssl/semireview.html

Author Biographies

Khurum Nazir Junejo is currently pursing a PhD in Computer Science from School of Science and Engineering at Lahore University of Management Sciences (LUMS) Lahore, Pakistan. He did his MS from LUMS and B.Sc. (Honors) from University of Karachi in 2007 and 2004, respectively. He also holds a teaching faculty position at Karachi campus of FAST National University of Computer and Emerging Sciences, Pakistan. His research interests include data mining and machine learning algorithms and applications with main focus on text and spam classification.

Asim Karim received his B.Sc. (Honors) Engineering degree from University of Engineering and Technology (UET) Lahore, Pakistan in 1994 and his MS and PhD degrees from The Ohio State University in 1996 and 2002, respectively. He is currently an associate professor of computer science at LUMS School of Science and Engineering, where he directs the Knowledge and Data Engineering research group. His research interests include data mining and machine learning algorithms and applications with recent focus on text analytics and Web applications. He is the author/co-author of two books and over forty research publications.