A System to Filter Unwanted Messages from OSN User Walls

Marco Vanetti, Elisabetta Binaghi, Elena Ferrari, Barbara Carminati, Moreno Carullo
Department of Computer Science and Communication
University of Insubria
21100 Varese, Italy

E-mail: {marco.vanetti, elisabetta.binaghi, elena.ferrari, barbara.carminati, moreno.carullo}@uninsubria.it

***Abstract*—One fundamental issue in today On-line Social Networks (OSNs) is to give users the ability to control the messages posted on their own private space to avoid that unwanted content is displayed. Up to now OSNs provide little support to this requirement. To fill the gap, in this paper, we propose a system allowing OSN users to have a direct control on the messages posted on their walls. This is achieved through a flexible rule-based system, that allows users to customize the filtering criteria to be applied to their walls, and a Machine Learning based soft classifier automatically labeling messages in support of content-based filtering.**

***Index Terms*—On-line Social Networks, Information Filtering, Short Text Classification, Policy-based Personalization.**

I. Introduction

On-line Social Networks (OSNs) are today one of the most popular interactive medium to communicate, share and disseminate a considerable amount of human life information. Daily and continuous communications imply the exchange of several types of content, including free text, image, audio and video data. According to Facebook statistics[1] average user creates 90 pieces of content each month, whereas more than 30 billion pieces of content (web links, news stories, blog posts, notes, photo albums, etc.) are shared each month. The huge and dynamic character of these data creates the premise for the employment of web content mining strategies aimed to automatically discover useful information dormant within the data. They are instrumental to provide an active support in complex and sophisticated tasks involved in OSN management, such as for instance access control or information filtering. Information filtering has been greatly explored for what concerns textual documents and, more recently, web content (e.g., [1], [2], [3]). However, the aim of the majority of these proposals is mainly to provide users a classification mechanism to avoid they are overwhelmed by useless data. In OSNs, information filtering can also be used for a different, more sensitive, purpose. This is due to the fact that in OSNs there is the possibility of posting or commenting other posts on particular public/private areas, called in general *walls*. Information filtering can therefore be used to give users the ability to automatically control the messages written on their own walls, by filtering out unwanted messages. We believe that this is a key OSN service that has not been provided so far. Indeed, today OSNs provide very little support to prevent unwanted messages on user walls. For example, Facebook allows users to state who is allowed to insert messages in their walls (i.e., friends, friends of friends, or defined groups of friends). However, no content-based preferences are supported and therefore it is not possible to prevent undesired messages, such as political or vulgar ones, no matter of the user who posts them. Providing this service is not only a matter of using previously defined web content mining techniques for a different application, rather it requires to design ad-hoc classification strategies. This is because wall messages are constituted by short text for which traditional classification methods have serious limitations since short texts do not provide sufficient word occurrences.

The aim of the present work is therefore to propose and experimentally evaluate an automated system, called *Filtered Wall* (FW), able to filter unwanted messages from OSN user walls. We exploit *Machine Learning* (ML) text categorization techniques [4] to automatically assign with each short text message a set of categories based on its content.

The major efforts in building a robust short text classifier are concentrated in the extraction and selection of a set of characterizing and discriminant features. The solutions investigated in this paper are an extension of those adopted in a previous work by us [5] from which we inherit the learning model and the elicitation procedure for generating pre-classified data. The original set of features, derived from endogenous properties of short texts, is enlarged here including exogenous knowledge related to the context from which the messages originate. As far as the learning model is concerned, we confirm in the current paper the use of neural learning which is today recognized as one of the most efficient solutions in text classification [4]. In particular, we base the overall short text classification strategy on *Radial Basis Function Networks* (RBFN) for their proven capabilities in acting as soft classifiers, in managing noisy data and intrinsically vague classes. Moreover, the speed

[1] http://www.facebook.com/press/info.php?statistics

Digital Object Indentifier 10.1109/TKDE.2011.230

in performing the learning phase creates the premise for an adequate use in OSN domains, as well as facilitates the experimental evaluation tasks.

We insert the neural model within a hierarchical two level classification strategy. In the first level, the RBFN categorizes short messages as *Neutral* and *Non-Neutral*; in the second stage, *Non-Neutral* messages are classified producing gradual estimates of appropriateness to each of the considered category.

Besides classification facilities, the system provides a powerful rule layer exploiting a flexible language to specify *Filtering Rules* (FRs), by which users can state what contents should not be displayed on their walls. FRs can support a variety of different filtering criteria that can be combined and customized according to the user needs. More precisely, FRs exploit user profiles, user relationships as well as the output of the ML categorization process to state the filtering criteria to be enforced. In addition, the system provides the support for user-defined *BlackLists* (BLs), that is, lists of users that are temporarily prevented to post any kind of messages on a user wall.

The experiments we have carried out show the effectiveness of the developed filtering techniques. In particular, the overall strategy was experimentally evaluated numerically assessing the performances of the ML short classification stage and subsequently proving the effectiveness of the system in applying FRs. Finally, we have provided a prototype implementation of our system having Facebook as target OSN, even if our system can be easily applied to other OSNs as well.

To the best of our knowledge this is the first proposal of a system to automatically filter unwanted messages from OSN user walls on the basis of both message content and the message creator relationships and characteristics. The current paper substantially extends [5] for what concerns both the rule layer and the classification module. Major differences include, a different semantics for filtering rules to better fit the considered domain, an online setup assistant to help users in FR specification, the extension of the set of features considered in the classification process, a more deep performance evaluation study and an update of the prototype implementation to reflect the changes made to the classification techniques.

The remainder of this paper is organized as follows. Section II surveys related work, whereas Section III introduces the conceptual architecture of the proposed system. Section IV describes the ML-based text classification method used to categorize text contents, whereas Section V illustrates FRs and BLs. Section VI illustrates the performance evaluation of the proposed system, whereas the prototype application is described in Section VII. Finally, Section VIII concludes the paper.

II. Related Work

The main contribution of this paper is the design of a system providing customizable content-based message filtering for OSNs, based on ML techniques. As we have pointed out in the introduction, to the best of our knowledge we are the first proposing such kind of application for OSNs. However, our work has relationships both with the state of the art in content-based filtering, as well as with the field of policy-based personalization for OSNs and, more in general, web contents. Therefore, in what follows, we survey the literature in both these fields.

A. Content-based filtering

Information filtering systems are designed to classify a stream of dynamically generated information dispatched asynchronously by an information producer and present to the user those information that are likely to satisfy his/her requirements [6].

In content-based filtering each user is assumed to operate independently. As a result, a content-based filtering system selects information items based on the correlation between the content of the items and the user preferences as opposed to a collaborative filtering system that chooses items based on the correlation between people with similar preferences [7], [8]. While electronic mail was the original domain of early work on information filtering, subsequent papers have addressed diversified domains including newswire articles, Internet "news" articles, and broader network resources [9], [10], [11]. Documents processed in content-based filtering are mostly textual in nature and this makes content-based filtering close to text classification. The activity of filtering can be modeled, in fact, as a case of single label, binary classification, partitioning incoming documents into relevant and non relevant categories [12]. More complex filtering systems include multi-label text categorization automatically labeling messages into partial thematic categories.

Content-based filtering is mainly based on the use of the ML paradigm according to which a classifier is automatically induced by learning from a set of pre-classified examples. A remarkable variety of related work has recently appeared, which differ for the adopted feature extraction methods, model learning, and collection of samples [13], [1], [14], [3], [15]. The feature extraction procedure maps text into a compact representation of its content and is uniformly applied to training and generalization phases. Several experiments prove that *Bag of Words* (BoW) approaches yield good performance and prevail in general over more sophisticated text representation that may have superior semantics but lower statistical quality [16], [17], [18]. As far as the learning model is concerned, there is a number of major approaches in content-based filtering and text classification in general showing mutual advantages and disadvantages in function of application dependent issues. In [4] a detailed comparison analysis has been conducted confirming superiority of Boosting-based classifiers [19], Neural Networks [20], [21] and Support Vector Machines [22] over other popular methods, such as Rocchio [23] and Naïve Bayesian [24]. However, it is worth to note that most of the work related to text filtering by ML has been applied for long-form text and

the assessed performance of the text classification methods strictly depends on the nature of textual documents.

The application of content-based filtering on messages posted on OSN user walls poses additional challenges given the short length of these messages other than the wide range of topics that can be discussed. Short text classification has received up to now few attention in the scientific community. Recent work highlights difficulties in defining robust features, essentially due to the fact that the description of the short text is concise, with many misspellings, non standard terms and noise. Zelikovitz and Hirsh [25] attempt to improve the classification of short text strings developing a semi supervised learning strategy based on a combination of labeled training data plus a secondary corpus of unlabeled but related longer documents. This solution is inapplicable in our domain in which short messages are not summary or part of longer semantically related documents. A different approach is proposed by Bobicev and Sokolova [26] that circumvent the problem of error-prone feature construction by adopting a statistical learning method that can perform reasonably well without feature engineering. However, this method, named Prediction by Partial Mapping, produces a language model that is used in probabilistic text classifiers which are hard classifiers in nature and do not easily integrate soft, multi-membership paradigms. In our scenario, we consider gradual membership to classes a key feature for defining flexible policy-based personalization strategies.

B. Policy-based personalization of OSN contents

Recently, there have been some proposals exploiting classification mechanisms for personalizing access in OSNs. For instance, in [27] a classification method has been proposed to categorize short text messages in order to avoid overwhelming users of microblogging services by raw data. The system described in [27] focuses on Twitter[2] and associates a set of categories with each tweet describing its content. The user can then view only certain types of tweets based on his/her interests. In contrast, Golbeck and Kuter [28] propose an application, called FilmTrust, that exploits OSN trust relationships and provenance information to personalize access to the website. However, such systems do not provide a filtering policy layer by which the user can exploit the result of the classification process to decide how and to which extent filtering out unwanted information. In contrast, our filtering policy language allows the setting of FRs according to a variety of criteria, that do not consider only the results of the classification process but also the relationships of the wall owner with other OSN users as well as information on the user profile. Moreover, our system is complemented by a flexible mechanism for BL management that provides a further opportunity of customization to the filtering procedure.

The only social networking service we are aware of providing filtering abilities to its users is MyWOT,[3] a social networking service which gives its subscribers the ability to: 1) rate resources with respect to four criteria: trustworthiness, vendor reliability, privacy, and child safety; 2) specify preferences determining whether the browser should block access to a given resource, or should simply return a warning message on the basis of the specified rating. Despite the existence of some similarities, the approach adopted by MyWOT is quite different from ours. In particular, it supports filtering criteria which are far less flexible than the ones of Filtered Wall since they are only based on the four above-mentioned criteria. Moreover, no automatic classification mechanism is provided to the end user.

Our work is also inspired by the many access control models and related policy languages and enforcement mechanisms that have been proposed so far for OSNs (see [29] for a survey), since filtering shares several similarities with access control. Actually, content filtering can be considered as an extension of access control, since it can be used both to protect objects from unauthorized subjects, and subjects from inappropriate objects. In the field of OSNs, the majority of access control models proposed so far enforce *topology-based access control*, according to which access control requirements are expressed in terms of relationships that the requester should have with the resource owner. We use a similar idea to identify the users to which a FR applies. However, our filtering policy language extends the languages proposed for access control policy specification in OSNs to cope with the extended requirements of the filtering domain. Indeed, since we are dealing with filtering of unwanted contents rather than with access control, one of the key ingredients of our system is the availability of a description for the message contents to be exploited by the filtering mechanism. In contrast, no one of the access control models previously cited exploit the content of the resources to enforce access control. Moreover, the notion of BLs and their management are not considered by any of the above-mentioned access control models.

Finally, our policy language has some relationships with the policy frameworks that have been so far proposed to support the specification and enforcement of policies expressed in terms of constraints on the machine understandable resource descriptions provided by Semantic web languages. Examples of such frameworks are KAoS [30] and REI [31], focusing mainly on access control, Protune [32], which provides support also to trust negotiation and privacy policies, and WIQA [33], which gives end users the ability of using filtering policies in order to denote given ”quality” requirements that web resources must satisfy to be displayed to the users. However, although such frameworks are very powerful and general enough to be customized and/or extended for different application scenarios they have not been specifically conceived to address information filtering in OSNs and therefore to consider the user social graph in the policy specification process. Therefore, we prefer to define our own abstract and more compact policy language, rather than extending one of the above-mentioned

[2] http://www.twitter.com
[3] http://www.mywot.com

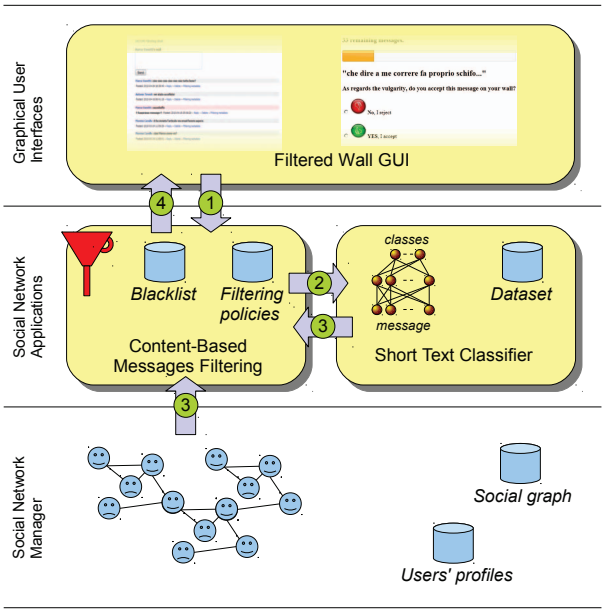

Fig. 1. Filtered Wall Conceptual Architecture and the flow messages follow, from writing to publication

ones.

III. Filtered Wall Architecture

The architecture in support of OSN services is a three-tier structure (Figure 1). The first layer, called *Social Network Manager* (SNM), commonly aims to provide the basic OSN functionalities (i.e., profile and relationship management), whereas the second layer provides the support for external *Social Network Applications* (SNAs).[4] The supported SNAs may in turn require an additional layer for their needed *Graphical User Interfaces* (GUIs). According to this reference architecture, the proposed system is placed in the second and third layers. In particular, users interact with the system by means of a GUI to set up and manage their FRs/BLs. Moreover, the GUI provides users with a FW, that is, a wall where only messages that are authorized according to their FRs/BLs are published.

The core components of the proposed system are the *Content-Based Messages Filtering* (CBMF) and the *Short Text Classifier* (STC) modules. The latter component aims to classify messages according to a set of categories. The strategy underlying this module is described in Section IV. In contrast, the first component exploits the message categorization provided by the STC module to enforce the FRs specified by the user. BLs can also be used to enhance the filtering process (see Section V for more details). As graphically depicted in Figure 1, the path followed by a message, from its writing to the possible final publication can be summarized as follows:

1) After entering the private wall of one of his/her contacts, the user tries to post a message, which is intercepted by FW.
2) A ML-based text classifier extracts metadata from the content of the message.
3) FW uses metadata provided by the classifier, together with data extracted from the social graph and users' profiles, to enforce the filtering and BL rules.
4) Depending on the result of the previous step, the message will be published or filtered by FW.

In what follows, we explain in more details some of the above-mentioned steps.

[4]See for example the Facebook Developers documentation, available at `http://developers.facebook.com/docs/`

IV. Short Text Classifier

Established techniques used for text classification work well on datasets with large documents such as newswires corpora [34], but suffer when the documents in the corpus are short. In this context, critical aspects are the definition of a set of characterizing and discriminant features allowing the representation of underlying concepts and the collection of a complete and consistent set of supervised examples.

Our study is aimed at designing and evaluating various representation techniques in combination with a neural learning strategy to sematically categorize short texts. From a ML point of view, we approach the task by defining a hierarchical two level strategy assuming that it is better to identify and eliminate "neutral" sentences, then classify "non neutral" sentences by the class of interest instead of doing everything in one step. This choice is motivated by related work showing advantages in classifying text and/or short texts using a hierarchical strategy [1]. The first level task is conceived as a hard classification in which short texts are labeled with crisp *Neutral* and *Non-Neutral* labels. The second level soft classifier acts on the crisp set of non-neutral short texts and, for each of them, it "simply" produces estimated appropriateness or "gradual membership" for each of the conceived classes, without taking any "hard" decision on any of them. Such a list of grades is then used by the subsequent phases of the filtering process.

A. Text Representation

The extraction of an appropriate set of features by which representing the text of a given document is a crucial task strongly affecting the performance of the overall classification strategy. Different sets of features for text categorization have been proposed in the literature [4], however the most appropriate feature set and feature representation for short text messages have not yet been sufficiently investigated. Proceeding from these considerations and on the basis of our experience [5], [35], [36], we consider three types of features, BoW, *Document properties* (Dp) and *Contextual Features* (CF). The first two types of features, already used in [5], are endogenous, that is, they are entirely derived from the information contained within the text of the message. Text representation using endogenous knowledge has a good general applicability, however in operational settings it is legitimate to use also exogenous

knowledge, i.e., any source of information outside the message body but directly or indirectly related to the message itself. We introduce CF modeling information that characterize the environment where the user is posting. These features play a key role in deterministically understanding the semantics of the messages [4]. All proposed features have been analyzed in the experimental evaluation phase in order to determine the combination that is most appropriate for short message classification (see Section VI).

The underlying model for text representation is the *Vector Space Model* (VSM) [37] according to which a text document d_j is represented as a vector of binary or real weights $d_j = w_{1j}, \ldots, w_{|\mathcal{T}|j}$, where $\mathcal{T}$ is the set of terms (sometimes also called features) that occur at least once in at least one document of the collection $\mathcal{T}r$, and $w_{kj} \in [0; 1]$ represents how much term t_k contributes to the semantics of document d_j. In the BoW representation, terms are identified with words. In the case of non-binary weighting, the weight w_{kj} of term t_k in document d_j is computed according to the standard *term frequency - inverse document frequency* (tf-idf) weighting function [38], defined as

$$tf - idf(t_k, d_j) = \#(t_k, d_j) \cdot \log \frac{|\mathcal{T}_r|}{\#\mathcal{T}_r(t_k)} \qquad (1)$$

where $\#(t_k, d_j)$ denotes the number of times t_k occurs in d_j, and $\#\mathcal{T}_r(t_k)$ denotes the document frequency of term t_k, i.e., the number of documents in $\mathcal{T}r$ in which t_k occurs. Domain specific criteria are adopted in choosing an additional set of features, Dp, concerning orthography, known words and statistical properties of messages. Dp features are heuristically assessed; their definition stems from intuitive considerations, domain specific criteria and in some cases required trial and error procedures. In more details:

- *Correct words:* it expresses the amount of terms $t_k \in \mathcal{T} \cap \mathcal{K}$, where t_k is a term of the considered document d_j and $\mathcal{K}$ is a set of known words for the domain language. This value is normalized by $\sum_{k=1}^{|\mathcal{T}|} \#(t_k, d_j)$.
- *Bad words:* they are computed similarly to the *Correct words* feature, where the set $\mathcal{K}$ is a collection of "dirty words" for the domain language.
- *Capital words:* it expresses the amount of words mostly written with capital letters, calculated as the percentage of words within the message, having more than half of the characters in capital case. The rational behind this choice lies in the fact that with this definition we intend to characterize the willingness of the author's message to use capital letters excluding accidental use or the use of correct grammar rules. For example, the value of this feature for the document "To be OR NOt to BE" is 0.5 since the words "OR" "NOt" and "BE" are considered as capitalized ("To" is not uppercase since the number of capital characters should be strictly greater than the characters count).
- *Punctuations characters:* it is calculated as the percentage of the punctuation characters over the total number of characters in the message. For example, the value of the feature for the document "Hello!!! How're u doing?" is $5/24$.
- *Exclamation marks:* it is calculated as the percentage of exclamation marks over the total number of punctuation characters in the message. Referring to the aforementioned document, the value is $3/5$.
- *Question marks:* it is calculated as the percentage of question marks over the total number of punctuations characters in the message. Referring to the aforementioned document, the value is $1/5$.

Regarding features based on the exogenous knowledge, CF, instead of being calculated on the body of the message, they are conceived as the VSM representation of the text that characterizes the environment where messages are posted (topics of the discussion, name of the group or any other relevant text surrounding the messages). CFs are not very dissimilar from BoW features describing the nature of data. Therefore, all the formal definitions introduced for the BoW features also apply to CFs.

B. Machine Learning-based Classification

We address short text categorization as a hierarchical two-level classification process. The first-level classifier performs a binary hard categorization that labels messages as *Neutral* and *Non-Neutral*. The first-level filtering task facilitates the subsequent second-level task in which a finer-grained classification is performed. The second-level classifier performs a soft-partition of *Non-neutral* messages assigning a given message a gradual membership to each of the non neutral classes. Among the variety of multi-class ML models well-suited for text classification, we choose the RBFN model [39] for the experimented competitive behavior with respect to other state of the art classifiers.

RFBNs have a single hidden layer of processing units with local, restricted activation domain: a Gaussian function is commonly used, but any other locally tunable function can be used. They were introduced as a neural network evolution of exact interpolation [40], and are demonstrated to have the universal approximation property [41], [42]. As outlined in [43], RBFN main advantages are that classification function is non-linear, the model may produce confidence values and it may be robust to outliers; drawbacks are the potential sensitivity to input parameters, and potential overtraining sensitivity. The first level classifier is then structured as a regular RBFN. In the second level of the classification stage we introduce a modification of the standard use of RBFN. Its regular use in classification includes a hard decision on the output values: according to the winner-take-all rule, a given input pattern is assigned with the class corresponding to the winner output neuron which has the highest value. In our approach, we consider all values of the output neurons as a result of the classification task and we interpret them as gradual estimation of multi-membership to classes.

The collection of pre-classified messages presents some critical aspects greatly affecting the performance of the overall classification strategy. To work well, a ML-based

classifier needs to be trained with a set of sufficiently complete and consistent pre-classified data. The difficulty of satisfying this constraint is essentially related to the subjective character of the interpretation process with which an expert decides whether to classify a document under a given category. In order to limit the effects of this phenomenon, known in literature under the name of inter-indexer inconsistency [44], our strategy contemplates the organization of "tuning sessions" aimed at establishing a consensus among experts through discussion of the most controversial interpretation of messages. A quantitative evaluation of the agreement among experts is then developed to make transparent the level of inconsistency under which the classification process has taken place (see Section VI-B2).

We now formally describe the overall classification strategy. Let Ω be the set of classes to which each message can belong to. Each element of the supervised collected set of messages $D = \{(m_i, \vec{y}_i), \ldots, (m_{|D|}, \vec{y}_{|D|})\}$ is composed of the text m_i and the supervised label $\vec{y}_i \in \{0,1\}^{|\Omega|}$ describing the belongingness to each of the defined classes. The set D is then split into two partitions, namely the training set TrS_D and the test set TeS_D.

Let M_1 and M_2 be the first and second level classifier, respectively, and $\vec{y}_1$ be the belongingness to the *Neutral* class. The learning and generalization phase works as follows:

1) from each message m_i we extract the vector of features $\vec{x}_i$. The two sets TrS_D and TeS_D are then transformed into $TrS = \{(\vec{x}_i, \vec{y}_i), \ldots, (\vec{x}_{|TrS_D|}, \vec{y}_{|TrS_D|})\}$ and $TeS = \{(\vec{x}_i, \vec{y}_i), \ldots, (\vec{x}_{|TeS_D|}, \vec{y}_{|TeS_D|})\}$, respectively.
2) a binary training set $TrS_1 = \{(\vec{x}_j, \vec{y}_j) \in TrS \mid (\vec{x}_j, y_j), y_j = \vec{y}_{j_1}\}$ is created for M_1.
3) a multi-class training set $TrS_2 = \{(\vec{x}_j, \vec{y}_j) \in TrS \mid (\vec{x}_j, \vec{y}'_j), \vec{y}'_{j_k} = \vec{y}_{j_{k+1}},\ k = 2, \ldots, |\Omega|\}$ is created for M_2.
4) M_1 is trained with TrS_1 with the aim to recognize whether or not a message is non-neutral. The performance of the model M_1 is then evaluated using the test set TeS_1.
5) M_2 is trained with the non-neutral TrS_2 messages with the aim of computing gradual membership to the non-neutral classes. The performance of the model M_2 is then evaluated using the test set TeS_2.

To summarize, the hierarchical system is composed of M_1 and M_2, where the overall computed function $f : R^n \rightarrow R^{|\Omega|}$ is able to map the feature space to the class space, that is, to recognize the belongingness of a message to each of the $|\Omega|$ classes. The membership values for each class of a given message computed by f are then exploited by the FRs, described in the following section.

V. Filtering Rules and Blacklist management

In this section, we introduce the rule layer adopted for filtering unwanted messages. We start by describing FRs, then we illustrate the use of BLs.

In what follows, we model a social network as a directed graph, where each node corresponds to a network user and edges denote relationships between two different users. In particular each edge is labeled by the *type* of the established relationship (e.g., friend of, colleague of, parent of) and, possibly, the corresponding *trust* level, which represents how much a given user considers trustworthy with respect to that specific kind of relationship the user with whom he/she is establishing the relationship. Without loss of generality, we suppose that trust levels are rational numbers in the range $[0, 1]$. Therefore, there exists a direct relationship of a given type RT and trust value X between two users, if there is an edge connecting them having the labels RT and X. Moreover, two users are in an indirect relationship of a given type RT if there is a path of more than one edge connecting them, such that all the edges in the path have label RT. In this paper, we do not address the problem of trust computation for indirect relationships, since many algorithms have been proposed in the literature that can be used in our scenario as well. Such algorithms mainly differ on the criteria to select the paths on which trust computation should be based, when many paths of the same type exist between two users (see [45] for a survey).

A. Filtering rules

In defining the language for FRs specification, we consider three main issues that, in our opinion, should affect a message filtering decision. First of all, in OSNs like in everyday life, the same message may have different meanings and relevance based on who writes it. As a consequence, FRs should allow users to state *constraints on message creators*. Creators on which a FR applies can be selected on the basis of several different criteria, one of the most relevant is by imposing conditions on their profile's attributes. In such a way it is, for instance, possible to define rules applying only to young creators or to creators with a given religious/political view. Given the social network scenario, creators may also be identified by *exploiting information on their social graph*. This implies to state conditions on type, depth and trust values of the relationship(s) creators should be involved in order to apply them the specified rules. All these options are formalized by the notion of creator specification, defined as follows.

Definition 1. (Creator specification). *A creator specification creatorSpec implicitly denotes a set of OSN users. It can have one of the following forms, possibly combined:*

1) *a set of attribute constraints of the form* $an\ OP\ av$, *where* an *is a user profile attribute name,* av *and* OP *are, respectively, a profile attribute value and a comparison operator, compatible with* an*'s domain.*
2) *a set of relationship constraints of the form* $(m,\ rt,\ minDepth,\ maxTrust)$, *denoting all the OSN users participating with user* m *in a relationship of type* rt, *having a depth greater than or equal to* $minDepth$, *and a trust value less than or equal to* $maxTrust$.

Example 1. *The creator specification* $CS_1 = \{Age < 16, Sex = male\}$ *denotes all the males whose age is less than 16 years, whereas the creator specification* $CS_2 = \{Helen, colleague, 2, 0.4\}$ *denotes all the users who are colleagues of Helen and whose trust level is less than or equal to 0.4. Finally, the creator specification* $CS_3 = \{(Helen, colleague, 2, 0.4), (Sex = male)\}$ *selects only the male users from those identified by* CS_2.

A further requirement for our FRs is that they should be able to support the specification of *content-based filtering criteria*. To this purpose, we make use of the two-level text classification introduced in Section IV. Thanks to this, it is for example possible to identify messages that, with high probability, are neutral or non-neutral, (i.e., messages with which the *Neutral/Non-Neutral* first level class is associated with membership level greater than a given threshold); as well as, in a similar way, messages dealing with a particular second level class. However, average OSN users may have difficulties in defining the correct threshold for the membership level to be stated in a FR. To make the user more comfortable in specifying the membership level threshold, we have devised an automated procedure, described in the following section, who helps the users in defining the correct threshold.

The last component of a FR is the *action* that the system has to perform on the messages that satisfy the rule. The possible actions we are considering are "block" and "notify", with the obvious semantics of blocking the message, or notifying the wall owner and wait him/her decision.

A FR is therefore formally defined as follows.

Definition 2. (Filtering rule). *A filtering rule* FR *is a tuple* $(author, creatorSpec, contentSpec, action)$*, where:*

- $author$ *is the user who specifies the rule;*
- $creatorSpec$ *is a creator specification, specified according to Definition 1;*
- $contentSpec$ *is a Boolean expression defined on* content constraints *of the form* (C, ml)*, where* C *is a class of the first or second level and* ml *is the minimum membership level threshold required for class* C *to make the constraint satisfied;*
- $action \in \{block, notify\}$ *denotes the action to be performed by the system on the messages matching* $contentSpec$ *and created by users identified by* $creatorSpec$.

In general, more than a filtering rule can apply to the same user. A message is therefore published only if it is not blocked by *any* of the filtering rules that apply to the message creator. Note moreover, that it may happen that a user profile does not contain a value for the attribute(s) referred by a FR (e.g, the profile does not specify a value for the attribute Hometown whereas the FR blocks all the messages authored by users coming from a specific city). In that case, the system is not able to evaluate whether the user profile matches the FR. Since how to deal with such messages depend on the considered scenario and on the wall owner attitudes, we ask the wall owner to decide whether to block or notify messages originating from a user whose profile does not match against the wall owner FRs because of missing attributes.

B. Online setup assistant for FRs thresholds

As mentioned in the previous section, we address the problem of setting thresholds to filter rules, by conceiving and implementing within FW, an Online Setup Assistant (OSA) procedure. OSA presents the user with a set of messages selected from the dataset discussed in Section VI-A. For each message, the user tells the system the decision to accept or reject the message. The collection and processing of user decisions on an adequate set of messages distributed over all the classes allows to compute customized thresholds representing the user attitude in accepting or rejecting certain contents.

Such messages are selected according to the following process. A certain amount of *non neutral* messages taken from a fraction of the dataset and not belonging to the training/test sets, are classified by the ML in order to have, for each message, the second level class membership values. Class membership values are then quantized into a number of q_C discrete sets and, for each discrete set, we select a number n_C of messages, obtaining sets M_C of messages with $|M_C| = n_C q_C$, where $C \in \Omega - \{Neutral\}$ is a second level class. For instance, for the second level class $Vulgar$, we select 5 messages belonging to 8 degrees of vulgarity, for a total of 40 messages. For each second level class C, messages belonging to M_C are shown. For each displayed message m, the user is asked to express the decision $m_a \in \{Filter, Pass\}$. This decision expresses the willingness of the user to filter or not filter the message. Together with the decision m_a the user is asked to express the degree of certainty $m_b \in \{0, 1, 2, 3, 4, 5\}$ with which the decision is taken, where $m_b = 5$ indicates the highest certainty, whereas $m_b = 0$ indicates the lowest certainty.

The above described procedure can be interpreted as a membership function elicitation procedure within the fuzzy set framework [46]. For each non-neutral class C, the fuzzy set is computed as $F_C = \sum_{M_C} \phi(m_a, m_b)$, where

$$\phi(m_a, m_b) = \frac{1}{2} + \begin{cases} m_b/10 & \text{if } m_a = Filter \\ -m_b/10 & \text{if } m_a = Pass \end{cases}$$

The membership value for the non-neutral class C is determined by applying the defuzzyfication procedure described in [47] to F_C, this value is then chosen as a threshold in defining the filtering policy.

Example 2. *Suppose that Bob is an OSN user and he wants to always block messages having an high degree of vulgar content. Through the session with OSA, the threshold representing the user attitude for the Vulgar class is set to* 0.8*. Now suppose that Bob wants to filter only messages coming from indirect friends, whereas for direct friends such messages should be blocked only for those users whose*

trust value is below 0.5*. This filtering criteria can be easily specified through the following FRs:*[5]

- *((Bob,* $friendOf$*,* 2*,* 1*), (*$Vulgar$*,* 0.80*), block)*
- *((Bob,* $friendOf$*,* 1*,* 0.5*), (*$Vulgar$*,* 0.80*), block)*

Eve, a friend of Bob with a trust value of 0.6*, wants to publish the message "G*d d*mn f*ck*ng s*n of a b*tch!" on Bob's FW. After posting the message, receives it in input producing the grade of membership* 0.85 *for the class Vulgar. Therefore the message, having a too high degree of vulgarity, will be filtered from the system and will not appear on the FW.*

C. Blacklists

A further component of our system is a BL mechanism to avoid messages from undesired creators, independent from their contents. BLs are directly managed by the system, which should be able to determine who are the users to be inserted in the BL and decide when users retention in the BL is finished. To enhance flexibility, such information are given to the system through a set of rules, hereafter called *BL rules*. Such rules are not defined by the SNM, therefore they are not meant as general high level directives to be applied to the whole community. Rather, we decide to let the users themselves, i.e., the wall's owners to specify BL rules regulating who has to be banned from their walls and for how long. Therefore, a user might be banned from a wall, by, at the same time, being able to post in other walls.

Similar to FRs, our BL rules make the wall owner able to identify users to be blocked according to their profiles as well as their relationships in the OSN. Therefore, by means of a BL rule, wall owners are for example able to ban from their walls users they do not directly know (i.e., with which they have only indirect relationships), or users that are friend of a given person as they may have a bad opinion of this person. This banning can be adopted for an undetermined time period or for a specific time window. Moreover, banning criteria may also take into account users' behavior in the OSN. More precisely, among possible information denoting users' bad behavior we have focused on two main measures. The first is related to the principle that if within a given time interval a user has been inserted into a BL for several times, say greater than a given threshold, he/she might deserve to stay in the BL for another while, as his/her behavior is not improved. This principle works for those users that have been already inserted in the considered BL at least one time. In contrast, to catch new bad behaviors, we use the *Relative Frequency* (RF) that let the system be able to detect those users whose messages continue to fail the FRs. The two measures can be computed either locally, that is, by considering only the messages and/or the BL of the user specifying the BL rule or globally, that is, by considering all OSN users walls and/or BLs.

A BL rule is therefore formally defined as follows.

[5]For simplicity, we omit the author component of the rules.

Definition 3. (BL rule). *A BL rule is a tuple* $(author, creatorSpec, creatorBehavior, T)$*, where:*

- $author$ *is the OSN user who specifies the rule, i.e., the wall owner;*
- $creatorSpec$ *is a creator specification, specified according to Definition 1;*
- $creatorBehavior$ *consists of two components* $RFBlocked$ *and* $minBanned$*.* $RFBlocked$ = $(RF, mode, window)$ *is defined such that:*
  - $RF = \frac{\#bMessages}{\#tMessages}$*, where* $\#tMessages$ *is the total number of messages that each OSN user identified by* $creatorSpec$ *has tried to publish in the* $author$ *wall* $(mode = myWall)$ *or in all the OSN walls* $(mode = SN)$*; whereas* $\#bMessages$ *is the number of messages among those in* $\#tMessages$ *that have been blocked;*
  - $window$ *is the time interval of creation of those messages that have to be considered for* RF *computation;*

  $minBanned$ = $(min, mode, window)$*, where* min *is the minimum number of times in the time interval specified in* $window$ *that OSN users identified by* $creatorSpec$ *have to be inserted into the BL due to BL rules specified by* $author$ *wall* $(mode = myWall)$ *or all OSN users* $(mode = SN)$ *in order to satisfy the constraint.*
- T *denotes the time period the users identified by* $creatorSpec$ *and* $creatorBehavior$ *have to be banned from* $author$ *wall.*

Example 3. *The BL rule:*

$$(Alice,\ (Age\ <\ 16),\ (0.5,\ myWall,\ 1\ week),\ 3\ days)$$

inserts into the BL associated with Alice's wall those young users (i.e., with age less than 16*) that in the last week have a relative frequency of blocked messages on Alice's wall greater than or equal to* 0.5*.*

Moreover, the rule specifies that these banned users have to stay in the BL for three days. If Alice adds the following component (3,SN, 1 week) to the BL rule, she enlarges the set of banned users by inserting also the users that in the last week have been inserted at least three times into any OSN BL.

VI. Evaluation

In this section, we illustrate the performance evaluation study we have carried out the classification and filtering modules. We start by describing the dataset.

A. Problem and Dataset Description

The analysis of related work has highlighted the lack of a publicly available benchmark for comparing different approaches to content based classification of OSN short texts. To cope with this lack, we have built and made available a dataset D of messages taken from Facebook[6].

[6]The dataset, called *WmSnSec 2*, is available online at `www.dicom.uninsubria.it/~marco.vanetti/wmsnsec/`

1266 messages from publicly accessible Italian groups have been selected and extracted by means of an automated procedure that removes undesired spam messages and, for each message, stores the message body and the name of the group from which it originates. The messages come from the group's web page section, where any registered user can post a new message or reply to messages already posted by other users. e. The role of the group's name within the text representation features was explained in Section IV-A.

The set of classes considered in our experiments is $\Omega = \{Neutral, Violence, Vulgar, Offensive, Hate, Sex\}$, where $\Omega - \{Neutral\}$ are the second level classes. The percentage of elements in D that belongs to the *Neutral* class is 31%.

In order to deal with intrinsic ambiguity in assigning messages to classes, we conceive that a given message belongs to more than one class. Each message has been labeled by a group of five experts and the class membership values $\vec{y}_j \in \{0,1\}^{|\Omega|}$ for a given message m_j were computed by a majority voting procedure. After the ground truth collection phase, the messages have been selected to balance as much as possible second-level class occurrences.

The group of experts has been chosen in an attempt to ensure high heterogeneity concerning sex, age, employment, education and religion. In order to create a consensus concerning the meaning of the *Neutral* class and general criteria in assigning multi-class membership we invited experts to participate to a dedicated tuning session.

Issues regarding the consistency between the opinions of experts and the impact of the dataset size in ML classification tasks will be discussed and evaluated in Section VI-B.

We are aware of the fact that the extreme diversity of OSNs content and the continuing evolution of communication styles create the need of using several datasets as a reference benchmark. We hope that our dataset will pave the way for a quantitative and more precise analysis of OSN short text classification methods.

B. Short Text Classifier Evaluation

1) Evaluation Metrics: Two different types of measures will be used to evaluate the effectiveness of first level and second level classifications. In the first level, the short text classification procedure is evaluated on the basis of the contingency table approach. In particular, the derived well known Overall Accuracy (OA) index capturing the simple percent agreement between truth and classification results, is complemented with the Cohen's KAPPA (K) coefficient thought to be a more robust measure taking into account the agreement occurring by chance [48]

At second level, we adopt measures widely accepted in the Information Retrieval and Document Analysis field, that is, Precision (P), that permits to evaluate the number of false positives, Recall (R), that permits to evaluate the number of false negatives, and the overall metric F-Measure (F_β), defined as the harmonic mean between the above two indexes [49]. Precision and Recall are computed by first calculating P and R for each class and then taking the average of these, according to the macro-averaging method [4], in order to compensate unbalanced class cardinalities. The F-Measure is commonly defined in terms of a coefficient β that defines how much to favor Recall over Precision. We chose to set $\beta = 1$.

2) Numerical Results: By trial and error we found a quite good parameter configuration for the RBFN learning model. The best value for the M parameter, that determines the number of Basis Function, is heuristically addressed to $N/2$, where N is the number of input patterns from the dataset. The value used for the spread σ, which usually depends on the data, is $\sigma = 32$ for both networks M_1 and M_2. As mentioned in Section IV-A, the text has been represented with the BoW feature model together with a set of additional features Dp and contextual features CF. To calculate *Correct words* and *Bad words* Dp features we used two specific Italian word-lists, one of these is the CoLFIS corpus [50]. The cardinalities of TrS_D and TeS_D, subsets of D with $TrS_D \cap TeS_D = \emptyset$, were chosen so that TrS_D is twice larger than TeS_D.

Network M_1 has been evaluated using the OA and the K value. Precision, Recall and F-Measure were used for the M_2 network because, in this particular case, each pattern can be assigned to one or more classes.

Table I shows the results obtained varying the set of features used in representing messages. In order to isolate the contribution of the individual types of features, different text representation have been experimented, obtained by partial combination of BoW, Dp and CF sets. The best result is obtained considering the overall set of features and using BoW with term weighting measure. In this configuration we obtain good results with an OA and K equal to 80.0% and 48.1% for the M_1 classifier and $P = 76\%$, $R = 59\%$ and $F_1 = 66\%$ for the second level, M_2 classifier. However, in all the considered combinations, the BoW representation with tf-idf weighting prevails over BoW with binary weighting.

Considered alone, the BoW representation does not allow sufficient results. The addition of Dp features leads to a slight improvement which is more significant in the first level of classification. These results, confirmed also by the poor performance obtained when using Dp features alone, may be interpreted in the light of the fact that Dp features are too general to significantly contribute in the second stage classification, where there are more than two classes, all of non-neutral type, and it is required a greater effort in order to understand the message semantics. The contribution of CFs is more significant, and this proves that exogenous knowledge, when available, can help to reduce ambiguity in short message classification.

Table II presents detailed results for the best classifier (BoW+Dp with tf-idf term weighting for the first stage and BoW with tf-idf term weighting for the second stage). The *Features* column indicates the partial combination of features considered in the experiments. The *BoW TW* column indicates the type of term weighting measure adopted. Precision, Recall and F-Measure values, related to each class, show that the most problematic cases are the *Hate*

TABLE I
RESULTS FOR THE TWO STAGES OF THE PROPOSED HIERARCHICAL CLASSIFIER

Text Representation		First Level Classification		Second Level Classification		
Features	BoW TW	OA	K	P	R	F_1
Dp	-	69.9%	21.6%	37%	29%	33%
BoW	binary	72.9%	28.8%	69%	36%	48%
BoW	tf-idf	73.8%	30.0%	75%	38%	50%
BoW+Dp	binary	73.8%	30.0%	73%	38%	50%
BoW+Dp	tf-idf	75.7%	35.0%	74%	37%	49%
BoW+CF	binary	78.7%	46.5%	74%	58%	65%
BoW+CF	tf-idf	79.4%	46.4%	71%	54%	61%
BoW+CF+Dp	binary	79.1%	48.3%	74%	57%	64%
BoW+CF+Dp	tf-idf	80.0%	48.1%	76%	59%	66%

TABLE II
RESULTS OF THE PROPOSED MODEL IN TERM OF PRECISION (P), RECALL (R) AND F-MEASURE (F_1) VALUES FOR EACH CLASS

	First level		Second Level				
Metric	Neutral	Non-Neutral	Violence	Vulgar	Offensive	Hate	Sex
P	81%	77%	82%	62%	82%	65%	88%
R	93%	50%	46%	49%	67%	39%	91%
F_1	87%	61%	59%	55%	74%	49%	89%

TABLE III
AGREEMENT BETWEEN FIVE EXPERTS ON MESSAGE NEUTRALITY

	Classification		Neutral			Non-Neutral		
Expert	OA	K	P	R	F_1	P	R	F_1
Expert 1	93%	84%	97%	93%	95%	97%	93%	95%
Expert 2	92%	80%	91%	98%	94%	95%	78%	85%
Expert 3	95%	90%	99%	94%	97%	88%	99%	93%
Expert 4	90%	76%	89%	98%	93%	94%	73%	82%
Expert 5	94%	84%	94%	97%	95%	93%	85%	89%

TABLE IV
AGREEMENT BETWEEN FIVE EXPERTS ON NON-NEUTRAL CLASSES IDENTIFICATION

	Violence			Vulgar			Offensive			Hate			Sexual		
Expert	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Expert 1	89%	99%	94%	89%	97%	93%	80%	90%	85%	78%	98%	87%	82%	98%	89%
Expert 2	77%	83%	80%	92%	67%	78%	71%	60%	65%	71%	69%	70%	85%	67%	75%
Expert 3	81%	84%	83%	76%	96%	85%	67%	79%	72%	53%	89%	66%	84%	76%	80%
Expert 4	96%	41%	58%	92%	78%	84%	70%	60%	65%	79%	42%	54%	97%	64%	77%
Expert 5	84%	90%	87%	92%	77%	84%	77%	73%	75%	78%	84%	81%	85%	77%	82%

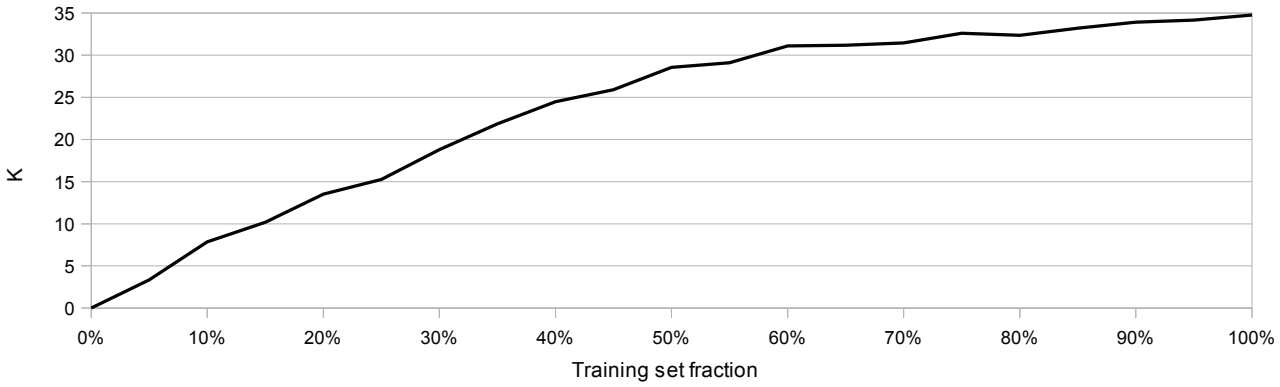

Fig. 2. K value obtained training the model with different fractions of the original training set

and *Offensive* classes. This can be attributed to the fact that messages with hate and offensive contents often hold quite complex concepts that hardly may be understood using a term based approach.

In Tables III and IV we report the results of a consistency analysis conducted comparing for each message used in training, the individual expert judgment with the attributed judgment. The attributed judgment results from the majority voting mechanism applied on the judgments collected by the the five considered experts. In most cases the experts reached a sufficient level of consistency reflecting however the inherent difficulty in providing consistent judgments. The lowest consistency values are in *Hate* and *Offensive* classes that are confirmed to be problematic.

We then performed an analysis aimed to evaluate the completeness of the training set used in the experiments to see to what extent the size of the dataset substantially contributes to the quality of classification. The analysis was conducted considering different training set configurations obtained with incremental fractions of the overall training set. For each fraction, we have performed 50 different distributions of messages between training set and test set, in order to reduce the statistical variability of each evaluation. The results, shown in Fig. 2, was obtained for each dataset fraction by averaging the K evaluation metric over 50 independent trials. Improvement in the classification has a logarithmic growth in function of the size of the dataset. This suggests that any further efforts focused in the enlargement of the dataset will probably lead to small improvements in terms of classification quality.

3) Comparison analysis: The lack of benchmarks for OSN short text classification makes problematic the development of a reliable comparative analysis. However, an indirect comparison of our method can be done with work that show similarities or complementary aspects with our solution. A study that responds to these characteristics is proposed in [27], where a classification of incoming tweets into five categories is described. Similarly to our approach, messages are very short and represented in the learning framework with both internal, content-based and contextual properties. In particular, the features considered in [27] are BoW, Author Name, plus 8 document properties features.

Qualitatively speaking, the results of the analysis conducted in [27] on the representative power of the three type of features tallied in general with our conclusions: contextual features are found to be very discriminative and BoW considered alone does not reach a satisfactory performance. Best numerical results obtained in our work are comparable with those obtained in [27]. Limiting to accuracy index, which is the only metric used in [27], our results are slightly inferior, but this result must be interpreted considering the following aspects. First of all, we use a much smaller set of pre-classified data (1266 vs 5407), and this is an advantage over the tweets classification considering the efforts in manually pre-classifying messages with an acceptable level of consistency. Secondly, the classes we considered have a higher degree of vagueness, since their semantics is closely linked to subjective interpretation. A second work [26] provides weak conditions for a comparative evaluation. The authors deal with short text classification using a statistical model, named Prediction by Partial Matching (PPM), without feature engineering. However, their study is oriented to text containing complex terminology and prove the classifier on medical texts from Newsgroups, clinical texts and Reuters-21578.[7] These differences may lower the level of reliability in comparison. In addition, we observe that the performance reported in [26] are strongly affected by the data set used in the evaluation. If we consider results in [26] obtained on clinical texts our classifier with the best results of Prec. 0.76, Recall 0.59, is considerably higher than PPM classifier (Prec. 0.36, Recall 0.42). It has a comparable behavior, if we consider the averaged performance on three Reuters subsets (Prec. 0.74, Recall 0.63) and slightly inferior when considering the newsgroups data set (Prec. 0.96, Recall 0.84).

C. Overall Performance and Discussion

In order to provide an overall assessment of how effectively the system applies a FR, we look again at Table II. This table allows us to estimate the Precision and Recall of our FRs, since values reported in Table II have been computed for FRs with content specification component set to $(C, 0.5)$, where $C \in \Omega$. Let us suppose that the system applies a given rule on a certain message. As such, Precision reported in Table II is the probability that the decision taken on the considered message (that is, blocking it or not) is actually the correct one. In contrast, Recall has to be interpreted as the probability that, given a rule that must be applied over a certain message, the rule is really enforced. Let us now discuss, with some examples, the results presented in Table II, which reports Precision and Recall values. The second column of Table II represents the Precision and the Recall value computed for FRs with $(Neutral, 0.5)$ content constraint. In contrast, the fifth column stores the Precision and the Recall value computed for FRs with $(Vulgar, 0.5)$ constraint.

Results achieved by the content-based specification component, on the first level classification, can be considered good enough and reasonably aligned with those obtained by well-known information filtering techniques [51]. Results obtained for the content-based specification component on the second level are slightly less brilliant than those obtained for the first, but we should interpret this in view of the intrinsic difficulties in assigning to a messages a semantically most specific category (see the discussion in Section VI-B2). However, the analysis of the features reported in Table I shows that the introduction of contextual information (CF) significantly improves the ability of the classifier to correctly distinguish between non-neutral classes. This result makes more reliable all policies exploiting non-neutral classes, which are the majority in real-world scenarios.

[7] Available online at `http://www.daviddlewis.com/resources/testcollections/reuters21578/`

Fig. 3. DicomFW: a message filtered by the wall's owner FRs (messages in the screenshot have been translated to make them understandable)

VII. DicomFw

DicomFW is a prototype Facebook application[8] that emulates a personal wall where the user can apply a simple combination of the proposed FRs. Throughout the development of the prototype we have focused our attention only on the FRs, leaving BL implementation as a future improvement. However, the implemented functionality is critical, since it permits the STC and CBMF components to interact.

Since this application is conceived as a wall and not as a group, the contextual information (from which CF are extracted) linked to the name of the group are not directly accessible. Contextual information that is currently used in the prototype is relative to the group name where the user that writes the message is most active. As a future extension, we want to integrate contextual information related to the name of all the groups in which the user participates, appropriately weighted by the participation level. It is important to stress that this type of contextual information is related to the environment preferred by the user who wants to post the message, thus the experience that you can try using DicomFW is consistent with what described and evaluated in Section VI-C.

To summarize, our application permits to:

1) view the list of users' FWs;
2) view messages and post a new one on a FW;
3) define FRs using the OSA tool.
   When a user tries to post a message on a wall, he/she receives an alerting message (see Figure 3) if it is blocked by FW.

VIII. Conclusions

In this paper, we have presented a system to filter undesired messages from OSN walls. The system exploits a ML soft classifier to enforce customizable content-dependent FRs. Moreover, the flexibility of the system in terms of filtering options is enhanced through the management of BLs.

[8] http://apps.facebook.com/dicompostfw/

This work is the first step of a wider project. The early encouraging results we have obtained on the classification procedure prompt us to continue with other work that will aim to improve the quality of classification. In particular, future plans contemplate a deeper investigation on two interdependent tasks. The first concerns the extraction and/or selection of contextual features that have been shown to have a high discriminative power. The second task involves the learning phase. Since the underlying domain is dynamically changing, the collection of pre-classified data may not be representative in the longer term. The present batch learning strategy, based on the preliminary collection of the entire set of labeled data from experts, allowed an accurate experimental evaluation but needs to be evolved to include new operational requirements. In future work, we plan to address this problem by investigating the use of on-line learning paradigms able to include label feedbacks from users. Additionally, we plan to enhance our system with a more sophisticated approach to decide when a user should be inserted into a BL.

The development of a GUI and a set of related tools to make easier BL and FR specification is also a direction we plan to investigate, since usability is a key requirement for such kind of applications. In particular, we aim at investigating a tool able to automatically recommend trust values for those contacts user does not personally known. We do believe that such a tool should suggest trust value based on users actions, behaviors and reputation in OSN, which might imply to enhance OSN with audit mechanisms. However, the design of these audit-based tools is complicated by several issues, like the implications an audit system might have on users privacy and/or the limitations on what it is possible to audit in current OSNs. A preliminary work in this direction has been done in the context of trust values used for OSN access control purposes [52]. However, we would like to remark that the system proposed in this paper represents just the core set of functionalities needed to provide a sophisticated tool for OSN message filtering. Even if we have complemented our system with an online assistant to set FR thresholds,

the development of a complete system easily usable by average OSN users is a wide topic which is out of the scope of the current paper. As such, the developed Facebook application is to be meant as a proof-of-concepts of the system core functionalities, rather than a fully developed system. Moreover, we are aware that a usable GUI could not be enough, representing only the first step. Indeed, the proposed system may suffer of problems similar to those encountered in the specification of OSN privacy settings. In this context, many empirical studies [53] have shown that average OSN users have difficulties in understanding also the simple privacy settings provided by today OSNs. To overcome this problem, a promising trend is to exploit data mining techniques to infer the best privacy preferences to suggest to OSN users, on the basis of the available social network data [54]. As future work, we intend to exploit similar techniques to infer BL rules and FRs.

Additionally, we plan to study strategies and techniques limiting the inferences that a user can do on the enforced filtering rules with the aim of bypassing the filtering system, such as for instance randomly notifying a message that should instead be blocked, or detecting modifications to profile attributes that have been made for the only purpose of defeating the filtering system.

REFERENCES

[1] A. Adomavicius, G.and Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Transaction on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.

[2] M. Chau and H. Chen, "A machine learning approach to web page filtering using content and structure analysis," *Decision Support Systems*, vol. 44, no. 2, pp. 482–494, 2008.

[3] R. J. Mooney and L. Roy, "Content-based book recommending using learning for text categorization," in *Proceedings of the Fifth ACM Conference on Digital Libraries*. New York: ACM Press, 2000, pp. 195–204.

[4] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.

[5] M. Vanetti, E. Binaghi, B. Carminati, M. Carullo, and E. Ferrari, "Content-based filtering in on-line social networks," in *Proceedings of ECML/PKDD Workshop on Privacy and Security issues in Data Mining and Machine Learning (PSDML 2010)*, 2010.

[6] N. J. Belkin and W. B. Croft, "Information filtering and information retrieval: Two sides of the same coin?" *Communications of the ACM*, vol. 35, no. 12, pp. 29–38, 1992.

[7] P. J. Denning, "Electronic junk," *Communications of the ACM*, vol. 25, no. 3, pp. 163–165, 1982.

[8] P. W. Foltz and S. T. Dumais, "Personalized information delivery: An analysis of information filtering methods," *Communications of the ACM*, vol. 35, no. 12, pp. 51–60, 1992.

[9] P. S. Jacobs and L. F. Rau, "Scisor: Extracting information from on-line news," *Communications of the ACM*, vol. 33, no. 11, pp. 88–97, 1990.

[10] S. Pollock, "A rule-based message filtering system," *ACM Transactions on Office Information Systems*, vol. 6, no. 3, pp. 232–254, 1988.

[11] P. E. Baclace, "Competitive agents for information filtering," *Communications of the ACM*, vol. 35, no. 12, p. 50, 1992.

[12] P. J. Hayes, P. M. Andersen, I. B. Nirenburg, and L. M. Schmandt, "Tcs: a shell for content-based text categorization," in *Proceedings of 6th IEEE Conference on Artificial Intelligence Applications (CAIA-90)*. IEEE Computer Society Press, Los Alamitos, US, 1990, pp. 320–326.

[13] G. Amati and F. Crestani, "Probabilistic learning for selective dissemination of information," *Information Processing and Management*, vol. 35, no. 5, pp. 633–654, 1999.

[14] M. J. Pazzani and D. Billsus, "Learning and revising user profiles: The identification of interesting web sites," *Machine Learning*, vol. 27, no. 3, pp. 313–331, 1997.

[15] Y. Zhang and J. Callan, "Maximum likelihood estimation for filtering thresholds," in *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001, pp. 294–302.

[16] C. Apte, F. Damerau, S. M. Weiss, D. Sholom, and M. Weiss, "Automated learning of decision rules for text categorization," *Transactions on Information Systems*, vol. 12, no. 3, pp. 233–251, 1994.

[17] S. Dumais, J. Platt, D. Heckerman, and M. Sahami, "Inductive learning algorithms and representations for text categorization," in *Proceedings of Seventh International Conference on Information and Knowledge Management (CIKM98)*, 1998, pp. 148–155.

[18] D. D. Lewis, "An evaluation of phrasal and clustered representations on a text categorization task," in *Proceedings of 15th ACM International Conference on Research and Development in Information Retrieval (SIGIR-92)*, N. J. Belkin, P. Ingwersen, and A. M. Pejtersen, Eds. ACM Press, New York, US, 1992, pp. 37–50.

[19] R. E. Schapire and Y. Singer, "Boostexter: a boosting-based system for text categorization," *Machine Learning*, vol. 39, no. 2/3, pp. 135–168, 2000.

[20] H. Schütze, D. A. Hull, and J. O. Pedersen, "A comparison of classifiers and document representations for the routing problem," in *Proceedings of the 18th Annual ACM/SIGIR Conference on Resea*. Springer Verlag, 1995, pp. 229–237.

[21] E. D. Wiener, J. O. Pedersen, and A. S. Weigend, "A neural network approach to topic spotting," in *Proceedings of 4th Annual Symposium on Document Analysis and Information Retrieval (SDAIR-95)*, Las Vegas, US, 1995, pp. 317–332.

[22] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proceedings of the European Conference on Machine Learning*. Springer, 1998, pp. 137–142.

[23] ——, "A probabilistic analysis of the rocchio algorithm with tfidf for text categorization," in *Proceedings of International Conference on Machine Learning*, 1997, pp. 143–151.

[24] S. E. Robertson and K. S. Jones, "Relevance weighting of search terms," *Journal of the American Society for Information Science*, vol. 27, no. 3, pp. 129–146, 1976.

[25] S. Zelikovitz and H. Hirsh, "Improving short text classification using unlabeled background knowledge," in *Proceedings of 17th International Conference on Machine Learning (ICML-00)*, P. Langley, Ed. Stanford, US: Morgan Kaufmann Publishers, San Francisco, US, 2000, pp. 1183–1190.

[26] V. Bobicev and M. Sokolova, "An effective and robust method for short text classification," in *AAAI*, D. Fox and C. P. Gomes, Eds. AAAI Press, 2008, pp. 1444–1445.

[27] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas, "Short text classification in twitter to improve information filtering," in *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010*, 2010, pp. 841–842.

[28] J. Golbeck, "Combining provenance with trust in social networks for semantic web content filtering," in *Provenance and Annotation of Data*, ser. Lecture Notes in Computer Science, L. Moreau and I. Foster, Eds. Springer Berlin / Heidelberg, 2006, vol. 4145, pp. 101–108.

[29] F. Bonchi and E. Ferrari, *Privacy-aware Knowledge Discovery: Novel Applications and New Techniques*. Chapman and Hall/CRC Press, 2010.

[30] A. Uszok, J. M. Bradshaw, M. Johnson, R. Jeffers, A. Tate, J. Dalton, and S. Aitken, "Kaos policy management for semantic web services," *IEEE Intelligent Systems*, vol. 19, pp. 32–41, 2004.

[31] L. Kagal, M. Paolucci, N. Srinivasan, G. Denker, T. Finin, and K. Sycara, "Authorization and privacy for semantic web services," *IEEE Intelligent Systems*, vol. 19, pp. 50–56, July 2004.

[32] P. Bonatti and D. Olmedilla, "Driving and monitoring provisional trust negotiation with metapolicies," in *In 6th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY 2005)*. IEEE Computer Society, 2005, pp. 14–23.

[33] C. Bizer and R. Cyganiak, "Quality-driven information filtering using the wiqa policy framework," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 7, pp. 1–10, January 2009.

[34] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li, "Rcv1: A new benchmark collection for text categorization research," *Journal of Machine Learning Research*, 2004.

[35] M. Carullo, E. Binaghi, and I. Gallo, "An online document clustering technique for short web contents," *Pattern Recognition Letters*, vol. 30, pp. 870–876, July 2009.
[36] M. Carullo, E. Binaghi, I. Gallo, and N. Lamberti, "Clustering of short commercial documents for the web," in *Proceedings of 19th International Conference on Pattern Recognition (ICPR 2008)*, 2008.
[37] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, UK: Cambridge University Press, 2008.
[38] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.
[39] J. Moody and C. Darken, "Fast learning in networks of locally-tuned processing units," *Neural Computation*, vol. 1, p. 281, 1989.
[40] M. J. D. Powell, "Radial basis functions for multivariable interpolation: a review," pp. 143–167, 1987.
[41] E. J. Hartman, J. D. Keeler, and J. M. Kowalski, "Layered neural networks with gaussian hidden units as universal approximations," *Neural Computation*, vol. 2, pp. 210–215, 1990.
[42] J. Park and I. W. Sandberg, "Approximation and radial-basis-function networks," *Neural Computation*, vol. 5, pp. 305–316, 1993.
[43] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 4–37, 2000.
[44] C. Cleverdon, "Optimizing convenient online access to bibliographic databases," *Information Services and Use*, vol. 4, no. 1, pp. 37–47, 1984.
[45] J. A. Golbeck, "Computing and applying trust in web-based social networks," Ph.D. dissertation, PhD thesis, Graduate School of the University of Maryland, College Park, 2005.
[46] J. L. Chameau and J. C. Santamarina, "Membership functions i: Comparing methods of measurement," *International Journal of Approximate Reasoning*, vol. 1, pp. 287–301, 1987.
[47] V. Leekwijck and W. Kerre, "Defuzzification: criteria and classification," *Fuzzy Sets and Systems*, vol. 108, pp. 159–178, 1999.
[48] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, March 1977.
[49] W. B. Frakes and R. A. Baeza-Yates, Eds., *Information Retrieval: Data Structures & Algorithms*. Prentice-Hall, 1992.
[50] A. Laudanna, A. M. Thornton, G. Brown, C. Burani, and L. Marconi, "Un corpus dell'italiano scritto contemporaneo dalla parte del ricevente," *III Giornate internazionali di Analisi Statistica dei Dati Testuali*, vol. 1, pp. 103–109, 1995.
[51] U. Hanani, B. Shapira, and P. Shoval, "Information filtering: Overview of issues, research and systems," *User Modeling and User-Adapted Interaction*, vol. 11, pp. 203–259, 2001.
[52] J. Nin, B. Carminati, E. Ferrari, and V. Torra, "Computing reputation for collaborative private networks," in *Proceedings of the 2009 33rd Annual IEEE International Computer Software and Applications Conference - Volume 01*. Washington, DC, USA: IEEE Computer Society, 2009, pp. 246–253.
[53] K. Strater and H. Richter, "Examining privacy and disclosure in a social networking community," in *Proceedings of the 3rd symposium on Usable privacy and security (SOUPS 2007)*. New York, NY, USA: ACM, 2007, pp. 157–158.
[54] L. Fang and K. LeFevre, "Privacy wizards for social networking sites," in *Proceedings of the 19th international conference on World wide web (WWW 2010)*. New York, NY, USA: ACM, 2010, pp. 351–360.

Marco Vanetti was born in Varese, Italy, in 1984. He received the B.Eng. in electronic engineering from Polytechnic University of Milan in 2006, and in 2009 the M.Sc. in Computer Science from University of Insubria. In 2010 he entered as Ph.D. student in Computer Science the ArTeLab research laboratory at the Department of Computer Science and Communications of University of Insubria. His research interests focus mainly on Computer Vision and Web Content Mining.

Elisabetta Binaghi received the degree in Physics from the University of Milan, Italy, in 1982. From 1985 to 1993, she worked at the Institute of Cosmic Physics of the National Research Council of Milan within the group of Image Analysis. In 1994 she joined the Institute for Multimedia Information Technology of National Research Council of Milan developing research in the field of Pattern Recognition, Image Analysis and Soft Computing. She coordinated research activities of the Artificial Intelligence and the Soft Computing Laboratory of the Institute. Since March 2002, she has been Associate Professor of Image Processing at the University of Insubria of Varese, Italy. In 2004 she was named Director of the Center of Research in Image Analysis and Medical Informatics. Her research interests include Pattern Recognition, Computational Intelligence and Computer Vision.

Elena Ferrari is a full professor of Computer Science at the University of Insubria, since March 2001, where she is the head of the Database and Web Security Group. Her research activities are related to various aspects of data management systems, including Web security, access control and privacy, Web content rating and filtering, multimedia and temporal databases. On these topics, she has published more than 120 scientific publications in international journals and conference proceedings. In 2009, she has been selected as the recipient of an IEEE Computer Society Technical Achievement Award for pioneering contributions to Secure Data Management. Prof. Ferrari is working / has worked on national and international projects such as SPADA-WEB, ANONIMO, EUFORBIA (IAP-26505), DHX (IST-2001-33476), and QUATRO Plus (SIP 2006-211001) and she recently received a Google Research Award.

Barbara Carminati is an assistant professor of Computer Science at the University of Insubria, Italy. Her main research interests are related to security and privacy for innovative applications, like XML data sources, semantic web, data outsourcing, web service, data streams and social networks. On these topics she has published more that fifty publications in international journals and conference proceedings. Barbara Carminati has been involved in several national and international research projects, among which a project funded by European Office of Aerospace Research and Development (EOARD), where she is PI. She has been involved in several conference organization (e.g., program chair of 15th SACMAT, general chair of the 14th SACMAT, tutorial, workshop and panel co-chair for International Conference on CollaborateCOM). Barbara Carminati is the editor in chief of the Computer Standards & Interfaces journal, Elsevier press.

Moreno Carullo was born in Varese, Italy, in 1982. He received both the B.Sc. and M.Sc. in Computer Science from University of Insubria in 2005 and 2007. He obtained his Ph.D. in Computer Science on Jan, 2011 from the University of Insubria. His research interests are focused on applied Machine Learning, Web Mining and Information Retrieval. He is currently an eXtreme Programming Coach at 7Pixel, an Italian company focused on price comparison services.