Chapter 7
Generating Explanations from Support Vector Machines for Psychological Classifications

Insu Song and Joachim Diederich

7.1 Introduction

In recent years, machine learning techniques, such as support vector machines (SVMs), have shown significant potential as aids to the practice of medicine and to psychiatric classification [1]. The application of machine learning techniques in psychiatric diagnosis has significant merit, because of the lack of standardized biological diagnostic tests. Conventionally, expert psychiatrists, consciously and unconsciously analyze the language of their patients for assessment purposes using diagnostic classification systems, such as DSM IV [9] and ICD-10 [12]. To provide a more objective clinical diagnosis, SVMs have been applied to conversations of patients and clinicians [1].

However, an explanation capability is crucial in security-sensitive domains, such as medical applications. Although support vector machines (SVMs) have shown superior performance in a range of classification and regression tasks, SVMs, like artificial neural networks (ANNs), lack an explanatory capability. There is a significant literature on obtaining human-comprehensible rules from SVMs and ANNs in order to explain how a decision was made or why a certain result was achieved [8].This chapter proposes a novel approach for SVM classifiers.

The experiments reported below describe a first attempt at generating textual and visual summaries for classification results. Learned model parameters are analyzed to select informative features, and filtering is applied to generate explanation terms by selecting subsets of more relevant and reliable features for

I. Song (✉)
School of Business and IT, James Cook University Australia, Singapore Campus, Singapore 574421, Singapore
e-mail: insu.song@jcu.edu.au

J. Diederich
School of Information Technology and Electrical Engineering, The University of Queensland, Brisbane, QLD 4072, Australia
e-mail: j.diederich@uq.edu.au

M. Lech et al. (eds.), *Mental Health Informatics*, Studies in Computational Intelligence 491, DOI: 10.1007/978-3-642-38550-6_7,

each case. We show that this approach is applicable to both linear and non-linear SVM classifiers.

To generate textual explanations (a set of sentences), a natural language parser is used to convert each text sample into a set of basic concept-constructs called basic-sentences (verb-subject-object tuples) that make up the sentences. For example, a sentence "I have a dog that is 9 years old" can be decomposed into two basic-sentences: "I have a dog" and "The dog is 9 years old" In some literatures, such basic-sentences are also referred to as grounded predicates. Generated explanation terms are used to rank relevant basic-sentences using a similarity measure function, which is based on a common sense database called ConceptNet. The ranked basic-sentences are used to generate textual explanations of SVM classifications. Unlike previous text summarization approaches, the generated text summaries explain why the particular sample is classified as positive or negative.

The generated explanations (informative features) are consistent in the sense that an explanation term does not appear in two separate explanations which are used to explain inconsistent samples. We define the accuracy of the explanation terms and show that the accuracy of an SVM model is bounded by the accuracy of explanation terms. That is, the accuracy of an explanation term is always greater or equal to the accuracy of an SVM model.

7.2 Background

The following section provides a brief overview of the core techniques, focusing on support vector machines (SVMs), the significance of generating human-comprehensible explanations from SVMs, and what it means to explain the decision-making process of a machine learning system to a human user who may not be a domain expert or familiar with methods in information technology.

7.2.1 Support Vector Machines

Cortes and Vapnik [7] introduce Support Vector Machines (SVMs) which are a novel approach to machine learning. SVMs are based on the structural risk minimization principle in order to overcome the overfitting problems. SVMs generate the hypotheses out of the hypothesis space H of a learning system which approximately minimizes the bound on the actual error by controlling the empirical error using training samples and the complexity of the model using the VC-dimension of H. SVMs are universal learning systems [13]. In their basic form, SVMs learn maximal margin hyperplanes (linear threshold functions). A hyperplane can be defined by a weight vector w and a bias b:

$$\mathbf{w} \cdot \mathbf{x} + b = 0 \tag{7.1}$$

The corresponding threshold function for an input vector x is then given by:

$$f(\mathbf{x}) = sign(\mathbf{w} \cdot \mathbf{x} + b) \tag{7.2}$$

However, it is possible to learn polynomial classifiers, radial basis function (RBF) networks, and three or more layered neural networks by mapping input data $\mathbf{x}$ to some other (possibly infinite dimensional) feature space $\phi(\mathbf{x})$ and using kernel functions $K(\mathbf{x}_i, \mathbf{x}_j)$ to obtain dot products, $\phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$, of feature vectors.

7.2.2 Explanations: The Foundation

To illustrate why it is important to add an explanation capability to SVMs, let us consider the case where medical doctors tell patients a diagnosis by use of test results or descriptions of symptoms. It is essential that doctors also use comprehensible explanations. The explanations may be via deductive arguments which include a list of patients' observed symptoms, a list of possible causes, and modus ponens (the rule of inference) for deriving the conclusion.

Thagard and Litt [19] illustrate several major approaches to generating explanations. The classical view is that explanations are deductive arguments that include background knowledge and inference rules, such as modus ponens. The inference rules allow the sequential application of if–then-else statements in order to justify explanatory targets. Whenever no precise knowledge is available, explanatory schemas or probabilistic rules can be used.

Cawsey [6] used a very simple definition of explanation: In general, explanations make knowledge clear to the hearers. Explanations is complete when the hearers are satisfied with the reply and understand the piece of knowledge. Hence, explanation is based on an information need.

7.2.3 Generating Explanations from SVMs

Much of the work that aims at providing an explanation capability to SVMs has focused on rule extraction techniques [8], following in the footsteps of efforts to obtain human-comprehensible rules from artificial neural networks (ANNs). One approach to classifying rule extraction methods is the translucency dimension which includes decompositional and pedagogical (or learning based) techniques as extremes [3].

The decompositional approach relies on the degree to which the internal representation of the ANN is accessible to the rule extraction technique. The basic strategy of decompositional techniques is to extract rules at the level of each individual hidden and output unit within the trained ANN. In general, decompositional rule extraction techniques incorporate some form of analysis of the weight vector and associated bias (threshold) of each unit in the trained ANN. Then, by treating each unit in the ANN as an isolated entity, decompositional techniques

initially generate rules in which the antecedents and consequents are expressed in terms which are local to the unit from which they are derived.

In contrast to the decompositional approaches, the strategy of the pedagogical approaches is to view the trained ANN at the minimum possible level of granularity, i.e., as a single entity or alternatively as a black box. The focus is on finding rules that map the ANN inputs (i.e., the attribute-value pairs from the problem domain) directly to outputs [22]. In addition to these two main categories, Andrews et al. [3] also proposed a third category which they labeled as eclectic to accommodate those rule extraction techniques which incorporate elements of both the decompositional and pedagogical approaches.

7.2.4 Translucency and Explanation Quality Applied to Explanation Extraction from SVMs

It is very easy to illustrate the limitations of current studies on rule extraction from SVMs by use of an example: text classification. SVMs can achieve good performance with very simple text representation formats such as the "bag-of words" (BOW) technique. BOW methods use a document-term matrix such that rows are indexed by the documents and columns by the terms (e.g. words). SVMs allow the classification of texts of differing lengths; hence, document vectors may differ greatly in the number of elements.

A disadvantage of the BOW representation is that after successful classification, it may not be obvious what has been learned. For instance, an author or speaker may have a preference for certain topics and, as a result, an SVM trained on an authorship identification problem may, in reality, perform topic detection. In the case of author or speaker verification, this problem has led to various techniques to eliminate content from the BOW input, for instance, by replacing content words with lexical tags (categories).

Given the fact that it is not at all obvious what contributes to classification in the case of a BOW input representation, rule extraction from support vector machines is presented with a special opportunity. However, the number of features in input or support vectors can be very large given, the number of words that exist in a given natural language. While a combination of words constitutes meaning in a natural language, a BOW representation is based on words in isolation. This is a significant problem for rule quality: The antecedents in a rule include individual words completely out of context. As the set of antecedents includes completely unrelated words, human or semantic comprehensibility is low.

7.2.5 Evaluation of the Quality of Extracted Explanations

Rule extraction from neural networks adopted criteria for the quality of the extracted rules. The set of criteria for evaluating rule quality includes [3]:

1. accuracy
2. fidelity
3. consistency, and
4. comprehensibility of the extracted rules.

A rule set is considered to be accurate if it can correctly classify a set of previously unseen examples from the problem domain [22]. Similarly, a rule set is considered to display a high level of fidelity if it can mimic the behavior of the neural network from which it was extracted by capturing all of the information represented in the ANN. An extracted rule set is deemed to be consistent if, under differing training sessions, the neural network generates rule sets which produce the same classification of unseen examples. Finally, the comprehensibility of a rule set is determined by measuring the size of the rule set (in terms of the number of rules) and the number of antecedents per rule [22].

7.2.6 Overview

The reminder of this chapter summarizes experiments and their results: classification of text and image data, explanation generation for classification results, and technical details of methods with statistical analysis on the model parameters that are generated for depression poems. Then, in Sect. 7.5, we show how explanation terms can be used to generate textual summaries of the classification results.

7.3 Experimental Evaluation

Figure 7.1b shows how our method can be used to provide explanation assisted Fig. 7.1a illustrates an overview of our approach of generating explanations for psychological assessments using Support Vector Machines (SVMs). Explanation terms are extracted from assessment documents using both SVM models and classification results. Figassessment of autism and other mental health issues. For example, the explanations can highlight the main issues that were used to differentiate autism cases from normal cases.

7.3.1 Methodology: Explanation Term Generation

A preliminary study was undertaken on generating explanations of classification results of depression poems, online text messages, autism descriptions, facial expressions, and facial palsy. Poems were obtained from the Internet (Poetry-America.com) and comprise a total of 76 poems: 56 depression poems and 20

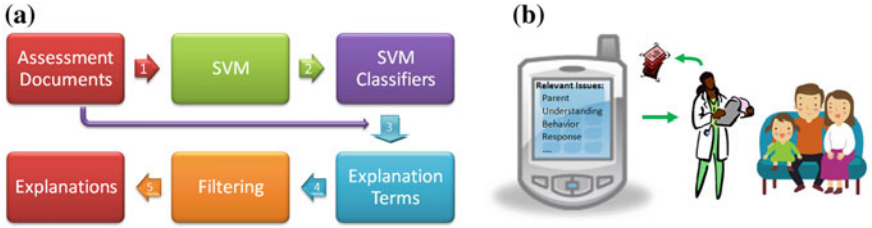

Fig. 7.1 **a** Illustrates the overall process of generating textual explanations to classification results. **b** Shows an example use of the explanation method, where a clinician make use of textual explanations to previous or current mental assessments, such as Autism. Mobile devices, such as smart phones, can be used to record interview questions and provide on the spot classification and explanations, resulting in more objective mental health assessments

funny poems. The online text messages were obtained from Usenet news groups and comprise a total of 350 sentences: 297 open questions and 53 closed questions. The autism descriptions were obtained from autism forums (http://www.autism-pdd.net) and ADHD (Attention Deficit Hyperactivity Disorder) forums (http://www.addforums.com) where parents discuss problems of their children. The autism data comprise a total of 200 descriptions: 100 autism descriptions and 100 ADHD descriptions.

For the text data sets, the resulting text documents are represented as attribute-value vectors (bag of words representation) where each distinct word corresponds to a feature whose value is the frequency of the word in the text sample: a text document is represented as a feature vector $\mathbf{x} = (x_{1,.}, x_{j,} \ldots, x_L)$ where x_j is the jth feature. Values were transformed with regard to the length of the sample. For the poem and autism data sets, functional words were removed, and each word was converted into its lemma form (its base form without inflections). In addition, words that were not present in ConceptNet [16][1] were removed. For the question data set, all words were used. In summary, input vectors for machine learning consist of attributes (the words used in the sample) and values (the transformed frequency of the words). Outputs are depression versus funny, open question versus closed question, and autism versus ADHD, that is, binary decision tasks were learned. Clearly, the expressive power of the resulting explanations is limited by this bag-of-words representation.

For LOO (leave-one-out) cross validation, 76, 350, and 200 SVM models were generated using the linear kernel for the poem, online message, and autism text data sets, respectively. Thus, each model was used to classify one document. An SVM model is defined by support vectors x_i and associated parameters. The decision value of a text sample (represented as a feature vector x) is then obtained as follows:

[1] Used ConceptNet v2.1 from the Common Sense Computing Initiative at the MIT Media Lab (http://csc.media.mit.edu).

$$d(x) = w \cdot \phi(x) + b = \sum_{i \in SV} \alpha_i y_i K(x_i, x) + b \tag{7.3}$$

where $\mathbf{x}_i$ are support vectors and $\mathbf{x}$ is the feature vector, α_i are Lagrangian multipliers, y_i are the labels (+1, −1) of the support vectors, and b is the offset. The support vectors and the Lagrangian multipliers can be found by solving a quadratic programming problem. A popular setup of an SVM quadratic programming problem that allows some classification errors in the solution [7] is shown below:

$$\begin{gathered} \min\left(\tfrac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i\right) \\ \text{subject to: } y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \end{gathered} \tag{7.4}$$

where C is the penalty for errors and ξ_i are slack variables for allowing errors. This particular formula isn't important here. The important thing is that the antecedent of the rule of inference is $d(x) \geq 0$. That is, if $d(x) \geq 0$, then the feature vector x is positive or else negative. Unlike previous rule extraction approaches, for each sample x, we formulate textual summaries to explain why $d(x) \geq 0$ or $d(x) < 0$. We start with generating an explanation (a set of explanation terms) for each classification result. An explanation term is a selected feature that is considered to be informative in explaining why $d(x) \geq 0$ or $d(x) < 0$. For text documents, an explanation term can be a word. For image data, an explanation term can be a region in the image. In Sect. 7.5, we extend this method to generate textual summaries. Features are filtered according to their sensitivity and contributions to the decision value $d(x)$ with respect to a reference point $\mathbf{C} = (c_1, c_2, \ldots, c_\mathrm{L})$ in the input space. We define three types of explanations for a classification result of a feature vector $\mathbf{x}$, where each explanations comprise a subset of features x_j of the feature vector $\mathbf{x} = (x_1, x_2, \ldots, x_\mathrm{L})$:

1. Explanation A comprising all the features x_j contributing to the decision value $d(\mathbf{x})$ with its feature value x_j greater than a reference point c_j:
   - For $d(\mathbf{x}) > 0$, this includes all the features x_j with positive contribution values $d(\mathbf{x})_j > 0$ and a feature value x_j greater than a reference point c_j.
   - For $d(\mathbf{x}) > 0$, this includes all the features x_j with negative contribution values $d(\mathbf{x})_j > 0$ and afeature value x_j greater than a reference point c_j.
2. Explanation B comprising top-N contributing features that are sufficient to classify the feature vector:
   - For $d(\mathbf{x}) > 0$, the sum of contributions of the features included in B is greater than the absolute value of the sum of all the negative contributions from the other features of the feature vector $\mathbf{x}$:

$$\sum_{x_i \in B} d(\mathbf{x})_i + \sum_{d(x)_j < 0} d(\mathbf{x})_j > 0 \tag{7.5}$$

where $d(\mathbf{x})_i > 0$ are the positive contributions of the ith features that are included in B and $d(\mathbf{x})_j < 0$ are the negative contributions.

3. Explanation C comprising top-N contributing features that also have their sensitivity values, $|\partial d(\mathrm{x})/\partial x_j|$, greater than a threshold value c.

Technical details on generating each explanation types are described in Sect. 7.4. This approach is clearly decompositional in nature: analysis of the model parameters to select informative features and selecting subsets of more relevant features. Figure 7.2 summaries the significance of each type of explanation. It plots sensitivity, contribution, and word rank of all features of the depression poem and autism-ADHD text data sets. It shows that sample features having higher ranking order (more frequent words in the text corpus) and higher sensitivity values tend to have larger absolute contribution values. This suggests that features having higher sensitivity values and higher ranking orders provide greater information in decision making than other features. It also shows that most of large contributions are made by more frequent words (high rank words).

In Sect. 7.4, we show that the accuracy of explanation terms is positively correlated with the accuracy of the SVM model: the accuracy of explanation terms increases as the accuracy of the SVM model increases.

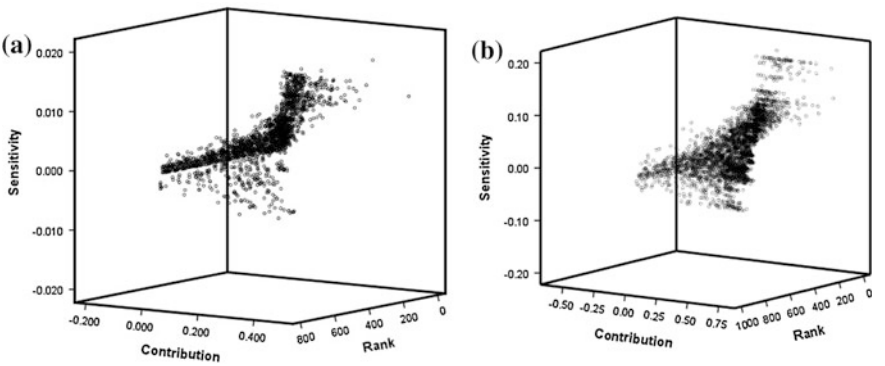

Fig. 7.2 Relationship between sensitivity, contribution (*deviation*), and word ranks. Each point is a feature that contributes to the decision of a feature vector. If a feature vector is a positive (*negative*) case, only the features having positive (*negative*) contributions are plotted. Rank 1 represents the most frequent feature (*vocabulary term*)

7.3.2 Results of Explanation Generation

7.3.2.1 Explanations of Text Classifications

Support vector machines trained on the poem and online text message data sets achieved accuracies of 94 and 98 %, respectively. The sensitivity values were adjusted manually to obtain reasonable numbers of explanation terms for Explanation type C. Sample explanations of a depressive poem are provided below:

1. Explanation A: dont (56 53), call (23 44), tear (21 50), know (17 18), fall (16 37), leave (16 35), cut (16 27), dark (14 24), sad (10 17), cold (10 16), face (10 20), smile (8 13), belong (5 7), letter (4 7), star (4 7), say (4 9), grave (2 3), shell (2 3), mold (1 2), useless (1 1).
2. Explanation B: dont (56 53), call (23 44), tear (21 50), **know** (17 18), fall (16 37), leave (16 35), cut (16 27), **dark** (14 24), **sad** (10 17), **cold** (10 16), **face** (10 20).
3. Explanation C: dont (56 53), call (23 44), tear (21 50), fall (16 37), leave (16 35), cut (16 27).

The numbers (d, q) in the brackets indicate relative contribution values d to the decision value d(x) and sensitivity values q, respectively. For positive cases, if the sensitivity value of a feature is positive, an increase in the frequency of the feature contributes to the decision value. Sensitivity filtering (Explanation C) eliminates some of less sensitive terms (bold-faced terms) from Explanation B. Sample explanations of a funny poem are provided below:

1. Explanation A: always (−45 −25), food (−34 −38), guy (−34 −38), come (−23 −32), name (−22 −24), good (−18 −21), im (−18 −25), same (−11 −14), best (−10 −11), go (−9 −13), mouth (−7 −8), true (−5 −5), happy (−2 −2), bad (−2 −2)
2. Explanation B: always (−45 −25), food (−34 −38), guy (−34 −38), come (−23 −32), name (−22 −24)
3. Explanation C: always (−45 −25), food (−34 −38), guy (−34 −38), come (−23 −32)

Explanation terms for negative cases have negative contribution values to derive $d(\mathbf{x})$ below zero. For negative cases, the sensitivity values of features must be negative. That is, the increase in the frequency of a feature contributes to deriving a decision value $d(\mathbf{x})$ below zero.

The explanations of the question data sets are much shorter (please note that this is a binary decision problem, that is, the task is to decide whether this is an open or closed question). Explanations for open questions (e.g., What is the dolphin species seen in most of Oceania?) are provided below:

1. Explanation A: what (539 166), species (34 8), most (15 3), in (7 2).
2. Explanation B: what (539 166)
3. Explanation C: what (539 166)

The corresponding explanations of a closed question (Do dolphins live shorter lives in captivity?) are provided here:

1. Explanation A: dolphins (−156 −35), live (−73 −15)
2. Explanation B: dolphins (−156 −35)
3. Explanation C: {empty}

As expected, questions are explained by the presence or absence of question words, such as what, why, or how.

Support vector machines trained on the autism description data set achieved an accuracy of 93 %. Sample explanations of the autism data are shown below with the sample sentences from the descriptions:

1. Explanation A for Autism description: speech (97 23), boy (91 45), begin (90 42), month (64 33), old (61 45), therapy (51 24), issue (50 23), train (48 22), school (39 25), year (38 26), soon (23 11), good (23 11), cream (10 4), improve (8 4), receive (4 1)
   - "…receives **speech therapy** in…. In recent **months**, his **speech** has **improved** greatly…. we will begin the… process very soon."
2. Explanation A for ADHD description: entire (−74 −24), sit (−60 −20), still (−52 −19), wall (−33 −11), pen (−26 −8), crayon (−21 −6), use (−11 −4), hour (−8 −2)
   - "…**uses** all the handwash to wash his hands, has drawn over his **entire wall** with **pen** and **crayon**…… and can **sit still** for **hours**."

The explanation terms are highlighted on the descriptions to provide contexts. This can assist clinicians to better understand their assessments more quickly during consultations as illustrated in Fig. 7.1b.

7.3.2.2 MPEG-7 Annotations for Explaining Facial Expressions and Facial Palsy

Figure 7.3 shows preliminary results of generating MPEG-7 annotations for explaining why a face image in a video frame is smiling or classified as facial palsy. The experiments shown here are to highlight that our method can be used to understand and verify learned classifier models. For example, in one attempt, we observed from the visual explanations that our classifier achieved good accuracy simply by learning different lightning conditions in the forehead facial regions. After equalizing image histograms, the SVM classifier learned more relevant features as shown in Fig. 7.3a, b, where it now highlights facial expression regions. For the expression classification task, we utilized facial expressions of one of the authors. The facial palsy images were obtained from Mater Misericordiae Health Services in Brisbane, which was previously used in [1]. Support vector

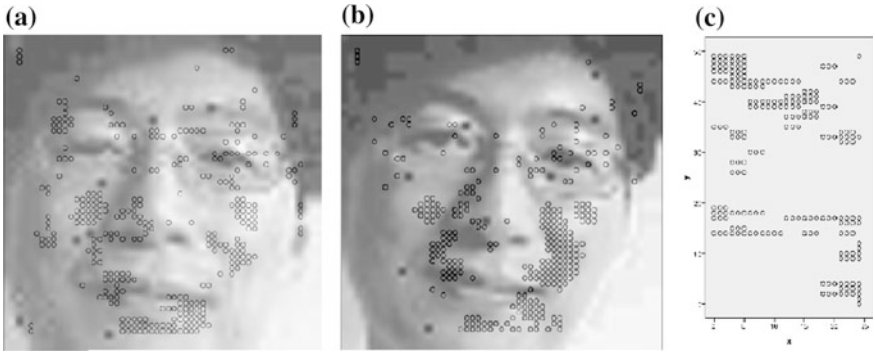

Fig. 7.3 Explaining image classification: (**a**) is a normal expression and (**b**) is a smiling expression. The dots represent explanation points. The last picture (c) shows explanation points of a facial palsy patient, highlighting the parts of face with deformities (the patient's face image removed for privacy reasons)

machines trained on the facial palsy images achieved an accuracy of 78 % and AUC of 0.84. For the facial expression classification task, the images were down sampled to 30 by 30 gray scale image features. For the facial palsy classification task, we used hamming distances between the right and left halves of each face image. Similarly to the method of generating textual explanations, learned SVM model parameters were analyzed to select informative features (pixels for facial expressions and hamming distances for facial palsy) and filtering was applied to select subsets of more relevant and reliable features. Further, the selected features were clustered to form explanation regions, which were then used to explain the classification of a region of interest in a video frame as shown in Fig. 7.3.

7.4 Generating Explanations from SVM Models

In order to calculate the contribution values of each feature of a feature vector **x**, we use the centroid **C** of the population as the reference point, which is estimated using the centroid $\mathbf{C}_{sv}$ of the support vectors:

$$C_{sv} = \frac{1}{Nsv} \sum_{i \in SV} \phi(x_i) \tag{7.6}$$

where N_{sv} is the number of support vectors. We use the estimated centroid as the neutral point where no clear decisions can be made. When the classifier is a non-linear classifier, we can identify K nearest support vectors in the input space and use them to form a centroid and a linear SVM model as shown in Fig. 7.9. Using the centroid, we can calculate the deviation of a feature vector x from the estimated population-centroid:

$$D(x) = \phi(x) - C_{sv} \tag{7.7}$$

The deviation vector $\mathbf{D}(\mathbf{x})$ represents how much the feature vector deviates from the center of the population. We use this deviation to calculate how each feature deviates from the centroid to contribute to the decision value $d(\mathbf{x})$. The contributions are obtained by projecting the deviation to the normal vector $\mathbf{w}$ of the hyperplane.

Corollary 1 *Let* $\mathbf{D}(\mathbf{x}) = \phi(\mathbf{x}) - \mathbf{C}$ *be the deviation of a feature vector from a centroid C of the population which is on an SVM hyperplane*: $\mathbf{w} \cdot \phi(\mathrm{x}) + b = 0$. *Then, the decision value of a feature vector is proportional to the projection of the deviation to a normal vector of the hyperplane*:

$$d(x) = w \cdot D(x) = a \frac{w}{||w||} \cdot D(x) \tag{7.8}$$

where a is a positive constant.

Proof Since $\mathbf{C}$ is on the hyperplane, we have $\mathbf{w} \cdot \mathbf{C} = -b$. Then, we can obtain the decision value $d(\mathbf{x})$ using the deviation $\mathbf{D}(\mathbf{x})$ as follows:

$$\begin{aligned} d(x) &= w \cdot (D(x) + C) + b = w \cdot D(x) \\ &= a \frac{w}{||w||} \cdot D(x) \end{aligned}$$

where a $= \| \mathbf{w} \|$.

For linear SVMs, we can obtain the contributions of the j th feature as shown below:

Corollary 2 *Let* $\mathbf{D}(\mathbf{x}) = \mathbf{x} - \mathbf{C}$ be *the deviation of a feature vector from a centroid* $\mathbf{C}$ *of the population which is on a linear SVM hyperplane:* $\mathbf{w} \cdot \mathbf{x} + b = 0$. Then, the deviation $\mathbf{D}(\mathbf{x})_j$ of the jth feature x_j of the feature vector $\mathbf{x}$ is proportional to the contribution $d(\mathbf{x})_j$ of the jth feature to the decision value $d(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$:

$$d(x)_j = w_j D(x)_j \tag{7.9}$$

where w_j is the jth component of the weight vector $\mathbf{w}$.

Proof According to Corollary 1, the decision value of the feature vector is proportional to the projection of the deviation to a normal vector of the hyperplane:

$$d(x) = w \cdot D(x)$$

If K is the linear kernel, we can estimate the contribution of each jth feature x_{j} as follows:

$$\begin{aligned} d(x)_j &= \sum_{i \in SV} \alpha_i y_i x_{i,j}(x_j - C_j) \\ &= (x_j - C_j) \sum_{i \in SV} \alpha_i y_i x_{i,j} \\ &= w_j D(x)_j \end{aligned} \tag{7.10}$$

where $w_j = \sum_{i \in SV} \alpha_i y_i x_{i,j}$ is the jth component of the weight vector $\mathbf{w}$.

For linear SVMs, we can directly use Corollary 2 to generate explanations. For non-linear classifiers with a convex hull, such as the elliptical decision boundary shown in Fig. 7.9, we can identify a subset of support vectors as reference points, which form both a centroid $\mathbf{C}'$ and a linear hyperplane with weight vector $\mathbf{w}'$ in the input space. The weight vector $\mathbf{w}'$ of the reference hyperplane can then be used with Corollary 2 to generate explanations for non-linear classifiers. The explanations will then be with respect to the reference point $\mathbf{C}'$. Figure 7.9 illustrates this procedure.

7.4.1 Consistency of Explanations

By the definition of Explanation A, if a sample is classified as positive (negative), a feature x_j is included in Explanation A as an explanation term only if $d(x)_j > 0$ ($d(x)_j < 0$), respectively. If a feature x_j is included in an explanation with $D(x)_j > 0$ for a sample, it means that the feature is included as an explanation term because the feature appears more frequently in the sample than the centroid C_j of the corresponding feature. Naturally then, for our explanations to be consistent, the same feature should not be included in an explanation to explain an opposite class. We now show that Explanations A, B, and C are consistent. Consistency is one of the criteria for evaluating rule quality [3].

Theorem 1 *Explanations A, B, and C are consistent: Given a feature vector x and its classification, let* $\mathrm{A}(\mathbf{x})$ *be its Explanation* A. *For Explanations* A, B, and C, *the following holds*:

1. If a feature vector $\mathbf{x}$ is classified as positive and $x_j \in \mathrm{A}(\mathbf{x})$, then $x_j \notin \mathrm{A}(\mathbf{x}')$ for all feature vectors $\mathbf{x}'$ that are classified as negative.
2. If a feature vector $\mathbf{x}$ is classified as negative and $x_j \in \mathrm{A}(\mathbf{x})$, then $x_j \notin \mathrm{A}(\mathbf{x}')$ for all feature vectors $\mathbf{x}'$ that are classified as positive.

Proof We first show that condition (1) holds. Suppose a sample x is classified as positive and x_j is included in Explanation A. By the definition of Explanation A, $d(\mathbf{x})_j = w_j\, \mathbf{D}(\mathbf{x})_j > 0$ and $\mathbf{D}(\mathbf{x})_j > 0$. Thus, $w_j > 0$. Now, we also suppose that the same feature x_j is included in Explanation A for a negative sample $\mathbf{x}'$, then by the definition of Explanation A $d(\mathbf{x}')_j = w_j\, \mathbf{D}(\mathbf{x}')_j < 0$ and $\mathbf{D}(\mathbf{x}')_j > 0$. This means that $w_j < 0$. Contradiction! Therefore, condition (1) must hold. Condition (2) can be proved similarly. B and C are subsets of A, and thus B and C are consistent as well.

The above theorem is applicable to certain non-linear SVM models by defining a reference point and a new linear hyperplane in the input space as shown in Fig. 7.9.

7.4.2 Accuracy of Explanation Terms

Another important criterion for evaluating rule quality is accuracy [3]. Conventionally, the accuracy of a binary classifier is defined as follows:

$$A_M = \frac{TP + TN}{N} \tag{7.11}$$

where N is the total number of samples, TP is the number of true-positive classification results, and TN is the number of true-negative classification results. Unfortunately, it is not that straightforward to define accuracy of explanation terms. However, we find that the following definition is the most natural way of defining the concept of accuracy of explanation terms. We start by defining the error rate of an explanation term.

Definition 1 The error rate of an explanation term x_j is the number of times that the term x_j is used incorrectly in an explanation divided by the number of explanations generated.

For Explanations A, B, and C, the number of times that x_j is used incorrectly in an explanation is the sum of the number of times that x_j is used for explaining negative samples with $d(x)_j > 0$ and the number of times that x_j is used for explaining positive samples with $d(x)_j > 0$.

Definition 2 Let M be a linear SVM model. Then, the empirical error rate $E_{j,M}$ of an explanation term x_j for Explanation A of the SVM model is

$$E_{j,M} = \frac{FP_{d(x)_j > 0} + FN_{d(x)_j < 0}}{N} \tag{7.12}$$

where N is the total number of samples, $FP_{d(x)j>0}$ is the number of explanations containing x_j for false-positive classification results, and $FN_{d(x)j<0}$ is the number of explanations containing x_j for false-negative classification results. The accuracy of an explanation term x_j is $A_{j,M} = 1 - E_{j,M}$.

With these definitions, we can now show that the accuracy of an SVM model is bounded by the accuracy of the explanation terms.

Theorem 2 *Let M* be *a linear SVM model. Then, the accuracy A_M of the SVM model M is bounded by the accuracy $A_{j,M}$ of explanation terms x_j:*

$$A_M \leq A_{j,M} \tag{7.13}$$

Proof The accuracy of an explanation term x_j is defined as follows:

$$\begin{aligned} A_{j,M} &= 1 - E_{j,M} \\ &= 1 - \frac{FP_{d(x)_j > 0} + FN_{d(x)_j < 0}}{N} \end{aligned}$$

By definition, the accuracy of the SVM model M is:

$$\begin{aligned} A_M &= \frac{TP + TN}{N} \\ &= 1 - \frac{FP + FN}{N} \end{aligned}$$

where FP is the number of false-positive classification results and FN is the number of false-negative results. By definition, the set of elements included in $FP_{d(x)j>0}$ ($FN_{d(x)j<0}$) is a subset of the set of elements included in FP (FN), respectively. Thus, $FP_{d(x)j>0} \leq FP$ and $FN_{d(x)j<0} \leq FN$. Furthermore, the following holds for any explanation term x_j:

$$\frac{FP_{d(x)_j > 0} + FN_{d(x)_j < 0}}{N} \leq \frac{FP + FN}{N}$$

Therefore, $A_M \leq A_{j,M}$ for any explanation term x_j.

7.4.2.1 Experimental Results of Accuracy of Explanation Terms

Figure 7.4 shows the distribution of the explanation term accuracy with the poem text data. As shown in Theorem 2, the minimum accuracy value is 0.94, which is the accuracy of the SVM model for the poem text data.

7.4.3 Fidelity of Explanations

In order to test the explanation capability of the explanation terms, we used all entries in Explanation A to generate a new feature set for the poem text data. By using explanation terms only for generating the new feature set, the vocabulary size was reduced from 1410 to 554. The ROC (Receiver Operating Curve) of the classification is shown in Fig. 7.5. Support vector machines trained on the new feature set achieved accuracy of 87 % and AUC (Area Under the Curve) of 0.89. The corresponding ROC (Receiver Operating Curve) is shown in Fig. 7.5.

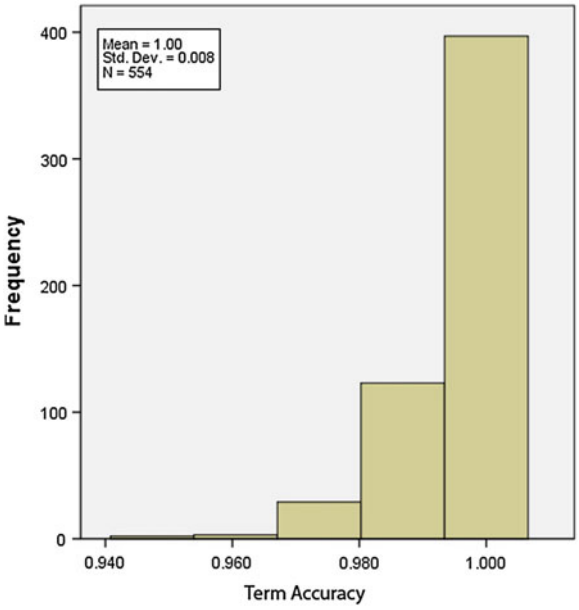

Fig. 7.4 Accuracy distribution of the explanation terms of the poem text data. The minimum accuracy value of the explanation terms is 0.94, which is the accuracy of the corresponding SVM model

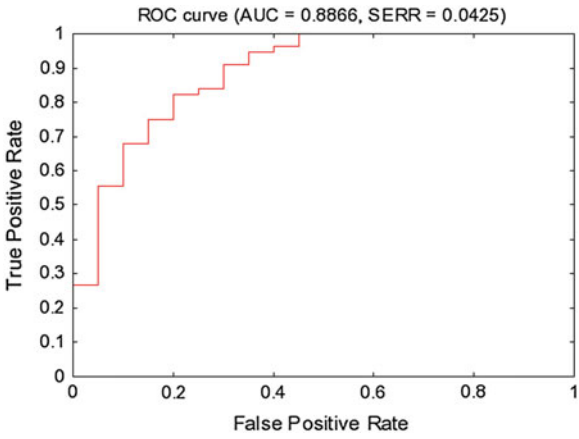

Fig. 7.5 True-positive rate versus false-positive rate of a linear support vector machine using an explanation-term vocabulary

The ROC curve for the SVM model using the full vocabulary is shown in Fig. 7.6. This clearly shows that the explanation terms are representative. That is, the explanations display a high level of fidelity because the explanation term set can mimic the behavior of the SVM model from which the explanation terms are extracted. According to Theorem 2, the explanation terms are also consistent. That is, if an explanation term is used to explain a positive case, then the same explanation term is not used to explain a negative case.

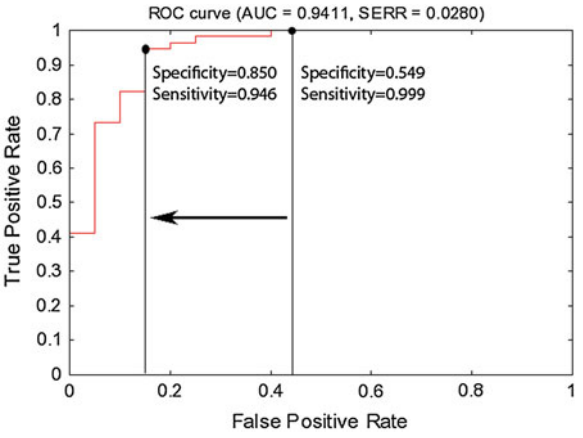

Fig. 7.6 ROC curve of the SVM model of the depression poem data. This illustrates that performance indicators can be adjusted by moving the threshold of the decision value. Specificity (recall) rate is increased from 0.549 to 0.85 by moving the decision threshold from 0 to 0.162. This effectively moves the estimated centroid of the population to produce more accurate explanations for imbalanced data

7.4.4 Optimization for Imbalanced Data

SVMs have been successfully applied to many text classification tasks, for example, to determine mental health problems using transcribed speech samples [1]. However, very few data sets are balanced: often the numbers of positive and negative samples are very different. This is particularly true for medical data, where few positive samples may be available, because positive cases are rare or there are a few negative examples only. For example, Autism assessment records contain very few negative cases, because most of the patients have been referred to Autism specialists by medical practitioners who have provided a first assessment. This imbalance can have a significant impact on the performance of machine learning algorithms. Furthermore, this imbalance in data can affect the accuracy of the estimated population-centroid. Thus explanations can become unreliable.

Various adjustments to SVMs have been proposed to improve the performance of SVMs with imbalanced data [14, 24]. Most of these approaches are based on the idea that the locations of the SVM hyperplanes can be adjusted to account for imbalanced data. One approach is to use separate cost factor measures C_+ and C_- for positive and negative samples, respectively [14]. Another approach is to adjust the bias term b [24] after n-fold cross validation to find optimal performance indicators.

In our experiments, we used the approach of adjusting the bias term b after n-fold cross validation. We used receiver-operating-curve (ROC) and balanced accuracy (BAC) as the heuristics for finding the optimal adjustment amount of the bias term.

$$BAC = \frac{specificity + sensitivity}{2}$$
$$specificity = \text{true\_negative\_rate} = P(O_-|L_-)$$
$$sensitivity = \text{true\_positive\_rate} = P(O_+|L_+)$$

Figure 7.6 shows the ROC curve of the LOO (leave-one-out) cross validation results for the depression poem data. Using the default bias, the model had a specificity value of 0.59. By adjusting the bias to b −0.17628 (i.e., a sample is classified as positive if $d(\mathbf{x}) > 0.17628$), we obtained specificity and sensitivity values of 0.8 and 0.96, respectively.

Adjusting the bias term moves the hyperplane along the weight vector $\mathbf{w}$. This movement has to be considered in calculating the deviation.

$$D'(x) = \phi(x) - (C_{sv} + \Gamma) \tag{7.14}$$

where $\mathbf{\Gamma}$ is the adjustment to the centroid. If the adjustment value to the bias term is δ (i.e., a sample is classified as positive if $d(\mathbf{x}) > \delta$), $\mathbf{\Gamma}$ is defined as follows:

$$\Gamma = \delta \frac{w}{||w||^2} = \delta_n w \tag{7.15}$$

where $\delta_n = \delta/\|\mathbf{w}\|^2$ is the normalized adjustment of the hyperplane. For a hyperplane in the input space $\mathbf{w} \cdot \mathbf{x} + b = \delta$, we can estimate the contribution of each jth feature x_j as follows:

$$d'(x)_j = \sum_{i \in SV} \alpha_i y_i x_{i,j} (x_j - C_{sv,j} - \delta_n x_{i,j}) \tag{7.16}$$

7.4.5 Filtering Explanations with Sensitivity

Training a support vector machine for a data set of interest generates a hyperplane, which can be used to obtain the distance of a feature vector to the hyperplane. The distance is normal to the hyperplane. Thus, the importance of a feature can be measured as the rate of change of the distance with respect to the feature. This can be easily obtained for linear classifiers as follows:

$$\frac{\partial d(x)}{\partial x_j} = \sum_{i \in SV} \alpha_i y_i x_{i,j} = w_j \tag{7.17}$$

where $d(\mathbf{x})$ is the distance of a feature vector $\mathbf{x}$ to the hyperplane, x_j is the jth feature, x_i is the jth feature value of a support vector $\mathbf{x}_i$, and w_j is the jth component of the weight vector $\mathbf{w}$.

Figure 7.7 shows a histogram of sensitivity and contribution values of features for the poem text data set. Greater population is centered at sensitivity value 0 and

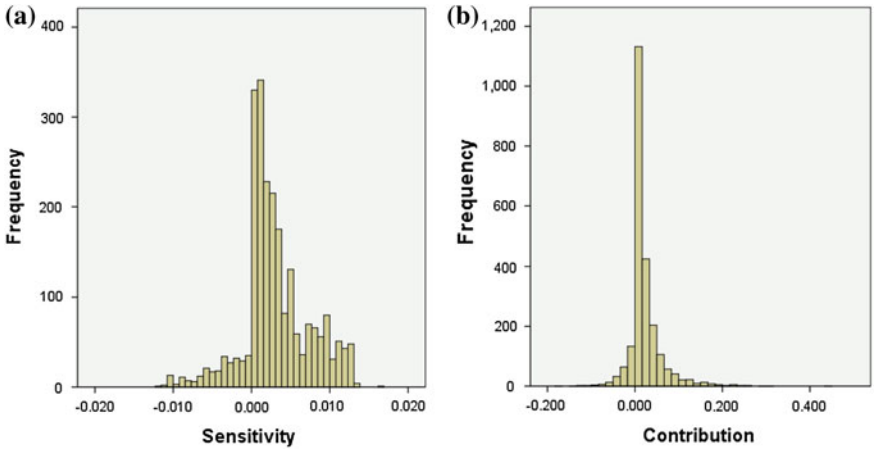

Fig. 7.7 Distribution of sensitivity and contribution values of the poem text data set

contribution value 0. This suggests that features having higher sensitivity and contribution values will provide more information on the decisions. It is also suggested in Fig. 7.2 that most of the contributions are made by terms with higher sensitivity values.

Figure 7.8 shows the relationship between word ranks and sensitivity values of sample features. Those with higher sensitivity values tend to have higher ranking order. The relationship between the word rank and sensitivity for the poem text data set can be summarized as follows:

$$rank \leq \alpha \frac{1}{|sensitivity|} \tag{7.18}$$

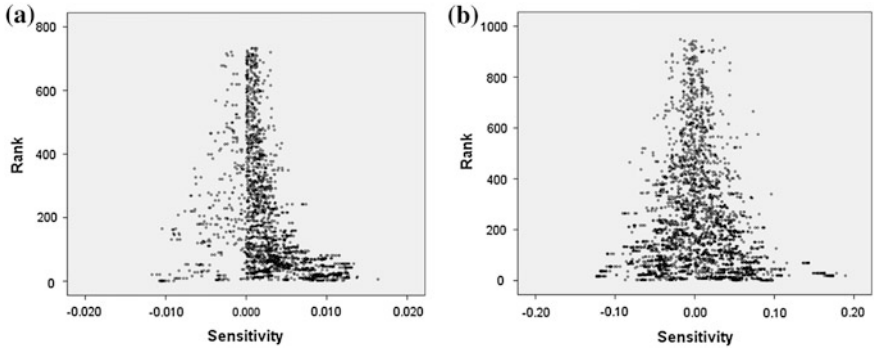

Fig. 7.8 Feature components with higher sensitivity values tend to have higher ranking orders (rank 1 is the highest rank and the most frequent term in the text corpus)

for some positive constant $\alpha > 0$. A similar relationship is also observed between the word rank and contribution across different text data sets ($\beta > 0$):

$$rank \leq \beta \frac{1}{|contribution|} \tag{7.19}$$

Whereas the contribution and sensitivity are proportional

$$|contribution| \leq \gamma |sensitivity| \tag{7.20}$$

which strongly suggests the two-step filtering (Explanation type B and Explanation type C).

7.4.6 Non-Linear SVMs

For non-linear cases, we have to obtain partial derivatives of kernels to obtain contribution and sensitivity values. As an example, let us consider the polynomial kernel: $K(\mathrm{x}_i \cdot \mathrm{x})_{\gamma,\mathrm{d}} = (\gamma \mathrm{x}_\mathrm{i} \cdot \mathrm{x} + r)^d$. The sensitivity of a feature x_j can be obtained as follows:

$$\begin{aligned} \frac{\partial d(x)}{\partial x_j} &= \sum_{i \in SV} \alpha_i y_i d\gamma x_{i,j} (\gamma x_i \cdot x + r)^{d-1} \\ &= d\gamma \sum_{i \in SV} \alpha_i y_i x_{i,j} K(x_i \cdot x)_{\gamma, d-1} \end{aligned} \tag{7.21}$$

This shows that the importance of the jth input feature for the hyperplane is a combination of other input features weighted by support vector elements. To avoid this, we can form a new hyperplane in the input space by identifying a subset of support vectors as shown in Fig. 7.9. The straight line in the figure is a newly formed hyperplane in the input space that separates the positive cases on the upper right side from the negative cases at the centre. The generated explanations are then, with respect to the new reference point, the centroid of the identified support vectors. This is similar to a decision tree algorithm in the sense that it divides the input space into subspaces, but using support vectors of an SVM model.

7.5 Contextual Text Summarization: Application of Explanation Generation

The explanations generated provide the relevance of each feature for a particular case. We can use this information to measure the relevance of each part of the text data to generate a textual summary with regard to a classification result. Unlike previous text summarization approaches, the textual summaries generated by the

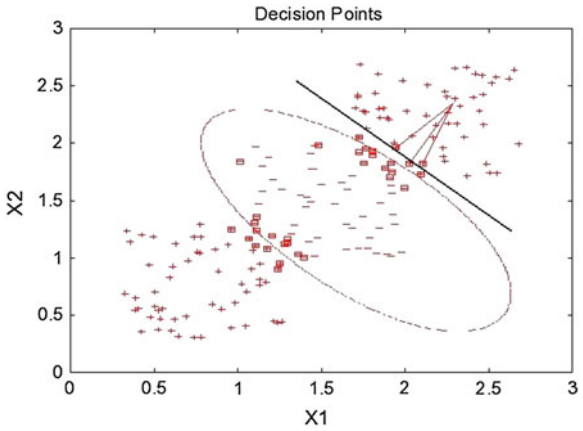

Fig. 7.9 K-NN (*nearest neighbour*) method of identifying a subset of support vectors (*square points*) to generate explanations for a convex non-linear decision boundary. In the figure, a positive data point at the upper right hand side is associated with the three nearest support vectors, and its distance to the centroid of the support vectors is used as an explanation. The support vectors form both a centroid and a new hyperplane in the input space

approach explained in this chapter aims at explaining why the particular sample is classified as positive or negative.

We start with a simple approach to generating a textual summary. The first method is scoring each sentence in a sample, using the explanation terms generated for the sample. In this approach, each sentence is given a score by determining contributions made by parts of the sentences. The score for the kth sentence in a sample text is defined as follows:

$$s_k = \frac{1}{|S_k||E|} \sum_{j \in X} \sum_{t \in S_k \cap E} u_j(t) d(x)_j q(x)_j \tag{7.22}$$

where j is a term included in explanation X, S_k is a set of terms in the kth sentence, E is the set of all explanation terms for the model, $u_j(t)$ is the utility function that measures how close the term t in $S_k \cap E$ is to term j, $d(x)$ is the amount of contribution of the feature j, and $q(x)_j$ is the sensitivity of the feature j. The text summary is then generated by selecting a subset of sentences from the text using the scores as relevance measures. The ConceptNet analogy space [16] is used as the utility function. The following example shows similarity values for $j =$ 'frustration':

- u_j (end) = 0.269967456035
- u_j (miss) = 0.75678954875
- u_j (cry) = 0.599278222108
- u_j (tear) = 0.168901775853

The second method we developed uses more basic elements of text. Each sentence is parsed into a set of basic-sentences (verb-subject-object tuples). Antecedents of pronouns are identified by using ConceptNet and the pronouns are replaced with their corresponding antecedents to improve readability of basic-sentences. For example, sentence I have a dog that is 9 years old" is decomposed into two basic sentences: p_1 = "I have a dog" and p_2 = "That is 9 years old". The pronoun 'that' is then replaced with its antecedent 'dog'. We then calculate scores of the basic-sentences. The score of the kth basic-sentence in a sample is defined as follows:

$$p_k = \frac{1}{|P_k||E|} \sum_{j \in X} \sum_{t \in P_k \cap E} u_j(t) d(x)_j q(x)_j \tag{7.23}$$

where $u_j(t)$ is the utility function that measures how close the term t in the basic-sentence ρ_k is to the feature j in an explanation X. Similarly, with the sentence-based summarization, the text summary is generate0d by selecting a subset of the basic-sentences using the scores as relevance measures.

7.5.1 Result of Text Summarization

The following is an example sentence-based text summary that is generated for a depression poem by selecting the top-5 most relevant sentences out of a total of 28 sentences:

> Time is the only one who can really tell us.
> Then soon enough it will be the end I cry almost every minute.
> So much pain so much hurt.
> You may ask and look concerned wanting to know why I cry.

The following is a sample basic-sentence-based text summary generated for the same depression poem by selecting the top-5 most relevant basic-sentences:

> It seem death. Death be me. You see. Who tell us

To evaluate the effectiveness of our approach, we measure similarities between each explanations and sentences of the poem text files. An explanation selects the top- K sentences that are most similar to the explanation. The error rate of this evaluation method is defined as follows:

$$error = \frac{\#\_of\_Incorrect\_Sentences\_selected\_in\_K}{K} \tag{7.24}$$

For an explanation of a depression (funny) poem, a sentence is incorrectly selected if the sentence is from a funny (depression) poem, respectively. This

measures the degree to which an explanation of a depressing (funny) poem prefers sentences of depressing (funny) poems, respectively. This is an extrinsic method of measuring text summaries based on relevance prediction [5], which is shown to be less sensitive than and positively correlated with ROUGE scores [15]. This is also a multi-document summarization task, but significantly different from existing approaches, such as clustering based approaches [2]. Our approach automatically extracts key words that are relevant to a given classification task and uses the set of key words to measure similarities.

Figure 7.10 shows the average error rates of 18 explanations when the top-K most similar sentences were selected out of 90 sentences (45 sentences from depression poems and 45 sentences from funny poems). The left figure (the unit of K is 5) shows that the average error rate was below 35 % when fewer than the top 15 most similar sentences were selected using top 3 most contributing explanation terms. Figure 7.11 shows that within-class similarities (the average similarity between explanations and sentences of the same classes) are consistently better than between-class similarities (the average similarity between explanations and sentences of the different classes).

Similar performance is achieved by selecting the top-K documents that are most similar to an explanation: a below 35 % error rate when selecting fewer than 35 % of the total documents. This method can be used to recommend patients with similar assessments to specific doctors or social networking communities.

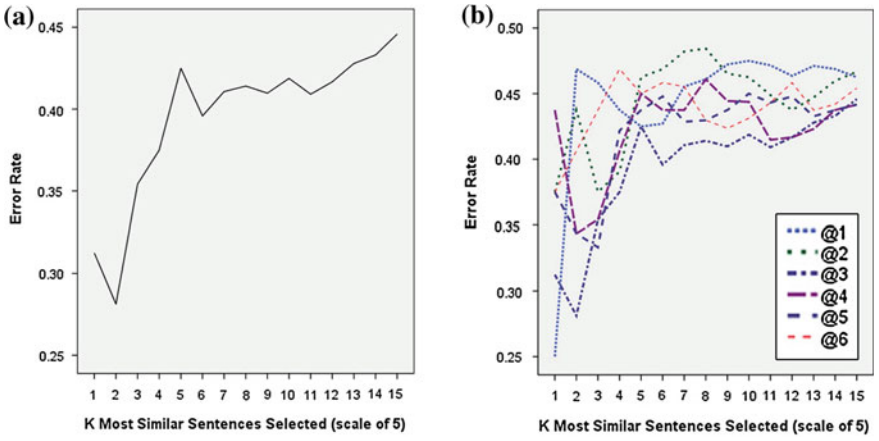

Fig. 7.10 Average error rates of text summaries are plotted against the K most similar sentences that were selected for each explanation. The figure on the left shows the average error rate when the top-3 most contributing terms are used to measure similarities. The figure on the right shows the average error rates for each of the top-N most contributing terms used ranging from 1 to 6 terms

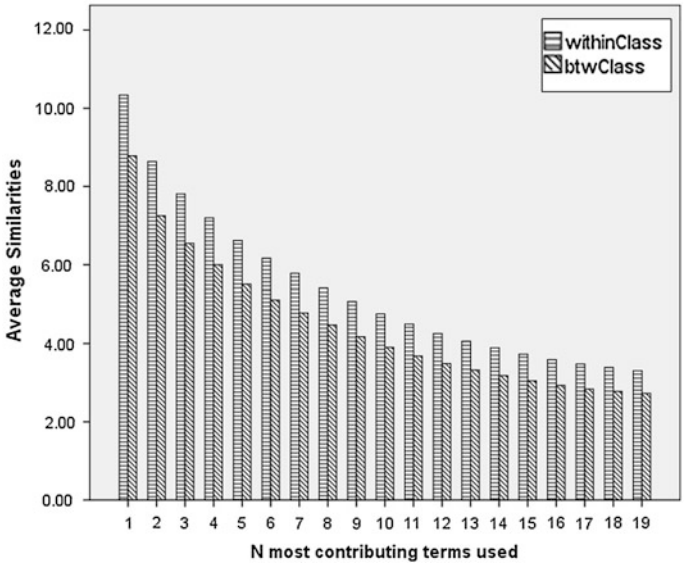

Fig. 7.11 Average similarity measures between explanations and poem sentences when the top-N most contributing terms are used to measure the similarities

7.6 Discussion and Future Work

Although feature selection and sensitivity analysis methods have been explored extensively in previous studies as to their ability to improve performance of machine learning algorithms [4, 11, 23, 25], the problem of generating explanations for each case has received much less attention. Decision tree classifiers [20] and other rule extraction methods [10, 17, 18, 21] can generate an explanation for each case, similar to our method, but do not have the explanation-term consistency or the explanation-term accuracy properties of our method. Similar rule extraction methods also are lacking in that no feature relevance information (e.g., contributions of each explanation terms) is provided.

This is the first report of a novel approach to generating consistent informative features for each separate case directly from SVM models and input data: consistent top-*N* features for each case as an explanation. This is also the first report of a novel approach to generating high-quality textual explanation for psychological classification: the selected features are used to generate textual explanations (sets of basic sentences) using semantic similarity measures. Other novel features include: (1) dynamic centroid identification in the context of SVM to identify reference points; and (2) selecting informative features for each case by combining both contributions and sensitivity.

We have shown that the explanations are consistent, accurate, display a high level of fidelity, and can generate text summaries with error rates below 35 %.

Furthermore, we have shown that the approach can be applied to imbalanced data by adjusting SVM hyperplanes and centroids using ROC curves. This approach can be easily extended to non-linear classifiers, for example, by combining with K-NN to identify support vectors that can be used as explanation reference points.

To improve the comprehensibility of explanations, the input text is parsed into basic-sentences and scored using a common sense database called ConceptNet. We believe that this approach overcomes the subjective nature of measuring comprehensibility. Considerable further research is required. The approach of extracting pieces of knowledge using machine learning in explaining psychiatric assessments has the potential to improve the usability of machine learning techniques in the medical and security domains. This approach can be further expanded by using alternative feature representations of text data sets, such as concept terms or semantic terms. There is massive potential for incorporating these sophisticated information extraction technologies within psychiatry and in medicine more generally.

References

1. Afifi, N., Diederich, J., Shanableh, T.: Computational methods for the detection of facial palsy. J. Telemedicine Telecare **12**(SUPPL. 3), 3–7 (2006)
2. Aliguliyev, R.M.: Clustering techniques and discrete particle swarm optimization algorithm for multi-document summarization. Comput. Intell. **26**(4), 420–448 (2010)
3. Andrews, R., Diederich, J., Tickle, A.B.: Survey and critique of techniques for extracting rules from trained artificial neural networks. Knowl.-Based Syst. **8**(6), 373–389 (1995)
4. Blum, A.L., Langley, P.: Selection of relevant features and examples in machine learning. Artif. Intell. **97**(1–2), 245–271 (1997). doi:10.1016/s0004-3702(97)00063-5
5. Dorr, B.J., Monz, C., President, S., Schwartz, R., Zajic, D.: A methodology for extrinsic evaluation of text summarization: Does rouge correlate?. In: Proceedings of the ACL 2005. pp. 1–8
6. Cawsey, A.: Explanation and interaction: the computer generation of explanatory dialogues. MIT Press, USA (1993)
7. Cortes, C., Vladimir, V.: Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995)
8. Diederich, J.: Rule extraction from support vector machines: An introduction. Stud. Comput. Intell. **80** (2008)
9. DSM-IV: Diagnostic and statistical manual of mental disorders. American Psychiatric Association (1994)
10. Féraud, R., Clérot, F.: A methodology to explain neural network classification. Neural Networks **15**(2), 237–246 (2002)
11. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J Mach Learn Res **3**, 1157–1182 (2003)
12. ICD10: International Statistical Classification of Disease and Related Health. World Health Organization, Geneva (1992)
13. Joachims, T.: Making Large-Scale SVM Learning Practical. Advances in Kernel Methods—Support Vector Learning. MIT-Press, NY (1999)
14. Morik, K., Brockhausen, P., Joachims, T.: Combining statistical learning with a knowledge-based approach—A case study in intensive care monitoring. In: Proceedings of the 16th International Conference on Machine Learning, Morgan Kaufmann Publishers, USA (1999)

15. Lin, C.-Y.: Rouge: A package for automatic evaluation of summaries. In: Proceedings of ACL workshop on Text Summarization Branches Out (2004)
16. Liu, H., Push, S.: Conceptnet: A practical commonsense reasoning toolkit. BT Technol. J. **22**, 211–226 (2004)
17. Menkovski, V., Christou, I.T., Efremidis, S.: Oblique decision trees using embedded support vector machines in classifier ensembles (2008)
18. Mitra, S., Hayashi, Y.: Neuro-fuzzy rule generation: survey in soft computing framework. IEEE Trans. Neural Networks **11**(3), 748–768 (2000)
19. Thagard, P., Little, A.: Models of Scientific Explanation. Cambridge University Press, Cambridge (2007)
20. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (1986)
21. Setiono, R., Liu, H.: A connectionist approach to generating oblique decision trees. IEEE Trans. Syst. Man Cybern. B Cybern. **29**(3), 440–444 (1999)
22. Tickle, A.B., Andrews, R., Golea, M., Diederich, J.: The truth will come to light: directions and challenges in extracting the knowledge embedded within trained artificial neural networks. IEEE Trans. Neural Networks **9**(6), 1057–1068 (1998)
23. Yan, J., Liu, N., Yan, S., Yang, Q., Fan, W., Wei, W., Chen, Z.: Trace-oriented feature analysis for large-scale text data dimension reduction. IEEE Trans. Knowl. Data Eng. **23**(7), 1103–1117 (2011). doi:10.1109/tkde.2010.34
24. Yang, Y: A study on thresholding strategies for text categorization. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New Orleans, LA, 9–13 September 2001, pp. 137–145 (2001)
25. Yu, L., Liu, H.: Efficient feature selection via analysis of relevance and redundancy. J Mach Learn Res **5**, 1205–1224 (2004)