Zigmund Bluvband and Sergey Porotsky (A.L.D., Israel)

Multi-Label Text Categorization Procedure for Aerospace Anomaly Discovery

Abstract
Article describes the results of the development and using of text mining algorithms to discover aerospace anomalies according analysis of free-text aviation safety reports. The goal of automatic text-categorization system is to assign new (or old, but not categorized) reports to several of predefined categories on the basis of their textual content. Examples of anomalies, extracted from ASRS data base, are "Ground Encounter: Animal", "In-Flight Encounter: Bird", "In-Flight Encounter: Weather", etc.
Optimal categorization functions can be constructed from labeled training examples (i.e., after expert categorization) by means of supervised learning algorithm and cross-validation. To support high values of output criteria (e.g., Recall and Precision have to be more than 90%) it is proposed mixed, partially automated approach – to select most of anomalies automatically, by means of text categorization algorithm, but sometimes to use human expertise. Some numerical results are considered.

BACKGROUND

Text classification (categorization) is a fundamental task in text mining. Since a text document often belongs to multiple categories, text categorization is generally defined as assigning one or more pre-defined category labels to each data sample. To solve this problem usually used approach, based on "supervised learning". It uses mathematical model "to learn" the relationship between a set of data and some known category field. We assume, that exact category of data may be assigned only manually, by means of the human expertise; examples of these situations are Aviation Safety Reports categorization, Handwritten Documents categorization, Medical Reports classification, Article abstracts classification, etc. and etc.

Prediction accuracy for Text Mining tasks with unbalanced dataset is usually calculated as follows:

$$recall = \frac{TP}{TP + FN} \qquad precision = \frac{TP}{TP + FP}$$

where:
FP is the number of negative examples incorrectly classified as positive (False Positives),
FN is the number of positive examples incorrectly classified as negative (False Negatives)
TP is the number of positive examples correctly classified (True Positives).

Several techniques may be used for data categorization – Logistic Regression, Neural Networks, Find Similar, SVM (Support Vector Machines), etc. Nevertheless even these different and numerous tools sometimes don't allow to receive high values simultaneously of Recall and Precision. According different articles, typical values for high-imbalanced data sets (amount of positive samples is less than 3%) are not more than Recall = Precision = 0.5…0.6. Even for low-imbalanced data sets (amount of positive samples is 5…20%) typical values are not more than Recall = Precision = 0.7…0.8. But in many situations it is necessary to provide high values of both Recall and Precision, e.g., more than 0.9…0.95. So, major limitation of these prior art approaches is as following: their inability to support specific for a given application and given category high values of both Recall and Precision.

APPROACH DESCRIPTION

We consider data points, received after current stage of regular Expert Categorization performing, of the form: {(**X**[1], **Y**[1]), (**X**[2], **Y**[2]) ,…, (**X**[n], **Y**[n]) where the **Y**[i] is vector (y_1[i],... ,y_k[i],... ,y_K[i]), and y_k[i] either 1 or −1 -- this label denotes the category k to which the point **X**[i] belongs. Each of **X**[i] is a *m* dimensional vector of the binary values [0 ; 1] or TF.IDF values . Label 1 means, that document i belong for category k, label -1 means, that document don't belong for category k. Index i = 1…n, where n is full

amount of documents on Training Set, used for current text categorization. For using of binary coding the component j of m dimensional vector **X**[i] equals for 1, if j-th word from vocabulary is concluded on the document number i, otherwise this component equals for 0. For coding of document according word frequency the component j of n dimensional vector **X**[i] equals for TF.IDF of this j-th word from vocabulary in the document i. Index j = 1…m, where m is full amount of words on the current vocabulary for category k (k = 1…K). Our method is based on SVM (Support Vector Machines) binary classification approach, i.e. for multi-label categorization we really K times solve binary categorization of type One-Versus-Rest.
For classification according category k we view set {**X**[i]**,** y_k[i]**}** as *training data*, which denotes the correct classification which we would like the SVM to eventually distinguish, by means of the dividing hyperplane, which takes the form

$y(\mathbf{X}) = \sum_{i=1}^{n} a[i]y\_k[i]KERN(\mathbf{X}, \mathbf{X}[i]) + b$, where KERN(**X, X**[i]) is kernel function and b is bias

The training is really followed for Quadratic Programming Task solving: to find values a[1],…,a[n]

to minimize $\sum_{i=1}^{n} \sum_{p=1}^{n} a[i]a[p]y\_k[i]y\_k[p]KERN(\mathbf{X}[i], \mathbf{X}[p]) - 2\sum_{i=1}^{n} a[i]$,

s.t. 0 <= a[i] <= C[i], $\sum_{i=1}^{n} a[i]y\_k[i] = 0.$

Kernel parameters (type, degree for polynomial, delta for RBF, etc.), penalty parameters C[i] and proposed parameters F_Low and F_High are meta-parameters and they are defined by means of tuning performing (cross-validation using) for current category k. Usually C[i] is same for all points i = 1..n. We have to use different values due to the following reason - training set for multi-label and multi-class text categorization tasks, usually is highly imbalanced. For example, for some category it may consist on 20000 documents, marked as "negative" and only 200 documents, marked as "positive". According this, penalty parameters C[i] may get following values:
C_pos, if current point **X**[i] belongs to the positive-marked Category k
C_neg, if current point **X**[i] belongs to the negative-marked Category k

To support required high values of Recall and Precision the following additional procedure is proposed:

1. Customer selects desired RECALL value and PRECISION value per each category (e.g., 0.85 for Recall and 0.95 for Precision).
2. Specific tuning procedure is applied to select optimal values of standard control meta-parameters (Kernel, Penalty, Gamma, etc) and optimal values of proposed meta-parameters (F_Low and F_High), to support required RECALL and PRECISION levels, and to minimize amount of reports, which have to be verified by expert manually after automatic data categorization .
   This "Tuning" process includes setting of control parameters (Kernel: linear, polynomial, RBF; kernel parameters: gamma, degree; penalty; …, F_Low, F_High), cross validation and minimization of amount of categories to be verified by expert under control parameters values.

After obtaining of values of parameters a[i] and meta-parameters the algorithm uses these parameters for recognizing Test set (new documents), i.e. not-marked documents. It is very simple and fast procedure, so it can be quickly applied for very large amount of documents (e.g., hundred thousand).

For each non-marked document **X** it is calculated the its value $y(\mathbf{X}) = \sum_{i=1}^{n} a[i]y\_k[i]KERN(\mathbf{X}, \mathbf{X}[i]) + b$:

If $y(\mathbf{X}) >= F\_High$, the non-marked document **X** is recognized as "category k" and expert should not verify this solution ;

If y(**X**) <= F_Low, the non-marked document **X** isn't recognized as "category k" and expert should not verify this solution ;
If F_Low < y(**X**) < F_High, the expert should verify this solution .

Partial expert evaluation (Last of the 3 above possibilities) should not improve Recall of category k recognition, but should essentially increase Precision of category k recognition.

Numerical Example

To evaluate our proposed method empirically, we used test collection from ASRS On-Line Data Base (http://asrs.arc.nasa.gov/index.html). The ASRS (Aviation Safety Reporting System) is a well-known textual data set for aviation safety. This data set is a collection of ~ 1,000,000 reports categorized into 58 different anomalies (categories). For each single report it is assigned between zero and 10 categories. Each report is represented as a vector of words. The entries in the vector are simpler binary feature values (a word either occurs or does not occur in a report) or word frequency. Text collections containing thousands of unique terms are quite common. Thus feature selection is widely used when applying machine-learning methods to text categorization. To reduce number of features (vocabulary size) we have used following approaches:

- Stemming and lemmatization. Stemming is a well known technique of word reduction by which common suffix and prefix are stripped from the original word form. Lemmatization is process by which words are reduced to their canonical form (e.g., verbs – to their infinitive)
- Using of "Exclusion List". They are non-significant words as "and", 'be", "about", etc
- Eliminating features (terms) that appear in only one report.
- Further elimination from vocabulary terms (features), that appear in only two, three,… reports. In this step we perform feature reduction by selecting the most informative terms independently for every category.

For text categorization tasks, we employed word-frequency vectors of documents as feature vectors input into classifiers, using the independent word-based representation, known as the Bag-of-Words (BOW) representation. We normalized the word-frequency vectors to do negligible the effect of vector size on computation. We employ word weighting methods such as Term Frequency and Inverse Document Frequency (TF. IDF). After building of vocabulary and coding of each document the following action is performed - data calibration and reduction of large values of some “extreme” features.

We extracted 16,000 and 10,000 reports as training and test samples, respectively. We removed vocabulary words included either in the stop list or in only one report. After this we have performed vocabulary reduction independently for each category up 500 vocabulary words. Results for two categories are summarized on the following table.

Anomaly (Category)	**Inflight Encounter – VFR in IMC**	**Aircraft Equipment Problem – Critical**
Full Amount of reports	10,000	10,000
Amount of reports with this anomaly	100	1600
Required value of Recall	>= 0.9	>= 0.85
Required value of Precision	>= 0.95	>= 0.9
Optimal control parameter values (after cross-validation, tuning and optimization) :		
Kernel Type	RBF	Linear
Gamma	0.001	-----
C_ neg	3	0.001
C_pos	60	0.003
F_Low	-1.0	-0.6
F_High	1.9	0.9
Results after Automatic Categorization		
Amount of "pseudo-positive" documents	930	2360
Amount of Loss (positive) documents	10	240
Amount of Garbage (negative) documents	840	1000
Document amount should be checked by expert	910	1500
Document amount should not be checked by expert	20	860
Final Results (after Automatic Categorization and Partial Human Expertise)		
Amount of documents, predictive as positive	95 (20 non-checked + 75 from 910 checked)	1520 (860 non-checked + 660 from 1500 checked)
Amount of Loss documents (FN)	10	240
Amount of Garbage documents (FP)	5 (from 20 non-checked)	160 (from 860 non-checked)
Recall	0.9	0.85
Precision	0.95	0.9
Acceleration of Expert Work (Reduction of document amount, manually checked by expert)	11 times (= 10,000/910)	7 times (= 10,000/1500)