Anomalous User Comment Detection in Social News Websites

Jorge de-la-Peña-Sordo, Iker Pastor-López, Xabier Ugarte-Pedrero, Igor Santos, and Pablo G. Bringas

S[3]Lab, DeustoTech Computing, University of Deusto
Avenida de las Universidades 24, 48007, Bilbao, Spain
{jorge.delapenya, iker.pastor, xabier.ugarte, isantos, pablo.garcia.bringas}@deusto.es

Abstract. The Web has evolved over the years and, now, not only the administrators of a site generate content. Users of a website can express themselves showing their feelings or opinions. This fact has led to negative side effects: sometimes the content generated is inappropriate. Frequently, this content is authored by troll users who deliberately seek controversy. In this paper we propose a new method to detect trolling comments in social news websites. To this end, we extract a combination of statistical, syntactic and opinion features from the user comments. Since this troll phenomenon is quite common in the web, we propose a novel experimental setup for our anomaly detection method: considering troll comments as base model (normal behaviour: 'normality'). We evaluate our approach with data from 'Menéame', a popular Spanish social news site, showing that our method can obtain high rates whilst minimising the labelling task.

Keywords: Information Retrieval, Troll Detection, Web Categorisation, Content Filtering, Machine-Learning.

1 Introduction

World Wide Web is more sociable than never, evolving from the Web 2.0 paradigm to nearly a global social network [1]. Thanks to the development of web technologies towards this paradigm, the Internet Community became more sensitive about the primordial users' needs when surfing the net. Since then, the users' dynamic interaction and collaboration was drastically enhanced.

On this basis, users have an active participation in the Internet and, particularly, in social news websites. In consequence, content generation within social webs has evolved. Users can comment diverse stories or other users' comments. However, this fact has led to negative side effects like the apparition of troll users and the increasing participation in social websites and so on. This phenomenon has been studied by the academic community. There is overview of related work that adequately account for the wealth of prior art dedicated to analysing, detecting and countering cyberbullying [2–4], trolling [5, 6] and flamewars [7–9] in social media. Social news websites such as Digg[1] or 'Menéame'[2] are very popular among users. These sites work in a very simple and intuitive way: users submit their links to stories online, and other users of these systems rate them by voting. The most voted stories are promoted and shown, finally, at the front-page [10].

We focus on 'Menéame'. This social news website has already a method for automatic moderation of comments and stories to automatically filter them. However, it is based on the votes of other users and, therefore, it can be manipulated. To avoid this problem, we have selected a more linguistic and statistical representation of the comments. There are approaches to filter spam in reviews [11, 12], that can be applied to this particular domain.

In our previous work [13], we proposed an approach able to automatically categorise comments in these social news sites using supervised machine-learning algorithms. Nevertheless, supervised learning requires a high number of labelled data for each of the classes (i.e., trolling or normal comment). It is quite difficult to obtain valuable information from unlabelled data for a real-world problem such as web mining and troll filtering. To generate and label these datasets, a time-consuming process of manual analysis is required.

Considering this background, we present a novel method based on anomaly detection to categorise troll comments that reduces the necessity of previous labelling (troll and 'not troll') of comments, as it measures the deviation of comments respect the base model (only employs the representation of base model comments). Since the difference between the number of troll and 'not troll' comments in our dataset is high, we considerer troll comments as the base model (denominated 'normality'). The features employed for the representation of the comments are statistical, syntactic and opinion based. If the comment under inspection exceed a threshold, it presents a considerable deviation to what it is considered normal; therefore, it will be considered anomalous.

In summary, our main contributions are: (i) an adaptation of the anomaly detection approach to comment filtering and (ii) an empirical validation which shows that our method can maintain high rates, minimising the effort of labelling.

The remainder of this paper is structured as follows. Section 2 describes the features extracted from the comments. Section 3 describes the anomaly detection based method we applied to this particular task. Section 4 describes the experimental procedure and discusses the obtained results. Finally, Section 5 concludes and outlines the avenues of the future work.

2 Method Description

'Menéame' is a Spanish social news website, in which news and stories are promoted. It was developed in later 2005 by Ricardo Galli and Benjamín Villoslada and it is currently licensed as free software. We labelled its comments regarding the controversy level of the comment: *Not Troll*, which raises no controversy and

[1] http://digg.com/
[2] http://meneame.net/

Troll, a comment that, on purpose, seeks controversy with harmful intention performed by a troll user.

2.1 Extracted Features

In this sub-section, we describe the features we extract from the comments, dividing them into 3 different categories: statistical, syntactic and opinion.

Statistical Features The statistical category has several features:

- **Comment body:** To represent the the information contained in the comment body we have used an Information Retrieval (IR) model. It can be defined as a 4-tuple $[\mathcal{C}, F, \mathcal{Q}, R(q_i, c_j)]$ [14] where $\mathcal{C}$, is a set of representations of comments; F, is a framework for modelling comments, queries and their relationships; $\mathcal{Q}$, is a set of representations of user queries; and, finally, $R(q_i, c_j)$ is a ranking function that associates a real number with a query q_i ($q_i \in \mathcal{Q}$) and a comment representation c_j ($c_j \in \mathcal{C}$).
  As $\mathcal{C}$ is the set of comments c, $\{c : \{t_1, t_2, ...t_n\}\}$, each comprising n terms $t_1, t_2, \ldots, t_n$, we define the weight $w_{i,j}$ as the number of times the term t_i appears in the comment c_j, if t_i is not present in c, $w_{i,j} = 0$. Therefore, a comment c_j can be represented as the vector of weights $\boldsymbol{c_j} = (w_{1,j}, w_{2,j}, ...w_{n,j})$.
  On the basis of this formalisation, IR systems commonly use the Vector Space Model (VSM) [14], which represents comments algebraically as vectors in a multidimensional space. This space consists only of positive axis intercepts. Comments are represented by a term-by-comment matrix, where the $(i, j)^{th}$ element illustrates the association between the $(i, j)^{th}$ term and the j^{th} comment. This association reflects the occurrence of the i^{th} term in comment j. Terms can represent diverse textual units (e.g., words or n-grams) and can also be individually weighted, allowing the terms to become more or less important within a comment or the collection $\mathcal{C}$ as a whole.
  We used the *Term Frequency – Inverse Document Frequency* (TF–IDF) [15] weighting schema, where the weight of the i^{th} term in the j^{th} comment, denoted by $weight(i, j)$, is defined by: $weight(i, j) = tf_{i,j} \cdot idf_i$ where *term frequency* $tf_{i,j}$ is defined as: $tf_{i,j} = n_{i,j}/\sum_k n_{k,j}$ where $n_{i,j}$ is the number of times the term $t_{i,j}$ appears in a comment c, and $\sum_k n_{k,j}$ is the total number of terms in the comment c. The inverse term frequency idf_i is defined as: $idf_i = |\mathcal{C}|/|\mathcal{C} : t_i \in c|$ where $|\mathcal{C}|$ is the total number of comments and $|\mathcal{C} : t_i \in c|$ is the number of comments containing the term t_i.
  As the terming schema we have employed two different alternatives. First, we used the word as term. Second, we used a n-gram approach. N-gram is the overlapping subsequence of n words from a given comment. In order to compare with our previous supervised machine-learning approach [13], we employed the same feature set, removing all the VSM attributes (both words and n-grams) devoid of value for the classification.

- **Number of references to the comment (in-degree):** It indicates the number of times the comment has been referenced in other comments of the same news story. In 'Menéame' the reference is indicated by the symbol '#' followed by the comment number. This measure should be effective in capturing the importance of a comment in the whole discussion.
- **Number of references from the comment (out-degree):** It indicates the number of references of the comment to other comments of the same news story. We consider that this feature captures if the comment is talking about the news story or, instead, is a comment about other comment.
- **The number of the comment:** We also use the number of the comment which indicates the oldness of the comment. In 'Menéame', as happens also in other media, if a news story has a high number of comments, the main topic has usually derived to a discussion which may be controversial.
- **The similarity of the comment with the snippet of the news story:** We used the similarity of the comment VSM with the snippet model of the news story. In particular, we employ the cosine similarity [16]: $sim(\overrightarrow{v}, \overrightarrow{u}) = \cos(\theta) = \overrightarrow{v} \cdot \overrightarrow{u} / ||\overrightarrow{v}|| \cdot ||\overrightarrow{u}||$ where $\overrightarrow{v} \cdot \overrightarrow{u}$ is the inner product of $\overrightarrow{v}$ and $\overrightarrow{u}$ whereas $||\overrightarrow{v}|| \cdot ||\overrightarrow{u}||$ is the cross product of $\overrightarrow{v}$ and $\overrightarrow{u}$. This value ranges from 0 to 1, where 0 means that the two of them are completely different (i.e., the vectors are orthogonal between them) and 1 means that the comments are equivalent. We have used this feature on the assumption that it can indicate how much the comment relates to the news story.
- **Number of coincidences between comment words and news story tags:** We have counted the number of words that appear in the comment and that are tags of the news story. We have used this measure because it could be indicative of how related the comment is respect to the news story.
- **Number of URLs in the comment body:** We have counted the number of URLs within the comment body. This feature tries to indicate whether the comment uses external sources in order to support its asseveration.

Syntactic Features In this category we count the number of words in the different syntactic categories. To this end, we performed a Part-of-Speech tagging using FreeLing[3]. The following features were extracted from the comment body, number of: (i) adjectives, (ii) numbers, (iii) dates, (iv) adverbs, (v) conjunctions, (vi) pronouns, (vii) punctuation marks, (viii) interjections, (ix) determinants, (x) abbreviations and (xi) verbs.

These features are intended to capture the user's type of language in a particular comment. For instance, a high-use of adjectives should be indicative of expressing an opinion. By capturing the type of language, the method may identify the controversy-level of the comment as well as the type of information contained in the comment.

Opinion Features Specifically, we used the following features:

[3] Available at `http://www.lsi.upc.edu/ nlp/freeling`

- **Number of positive and negative words:** We have counted the number of words in the comment with a positive meaning and the number of words in the comment with a negative meaning. We employed an external opinion lexicon[4]. Since the words in that lexicon are in English and 'Menéame' is written in Spanish, we have translated them into Spanish.
- **Number of votes:** The number of positive votes of the comment. The votes are given by other users in 'Menéame'.
- ***Karma*:** Computed by the website. Represents how important is the comment based on the amount of positive and negative votes to that comment.

We have used two features that are external to 'Menéame': the number of positive and negative words; and the opinion features that 'Menéame' has already computed. The latter ones are the number of positive votes of that comment and the 'karma', which is a concept used in 'Menéame' to moderate comments. These features are devoted to categorise the comment in its level of controversy because they indicate the opinion of the 'Menéame' community about the comment and, also, the polarisation of the comment by means of the number of positive/negative words.

3 Anomaly Detection

To represent the comments gathered from the website as points in the feature space, we employ our anomaly detection approach using the features described previously. Thereby, we are able to obtain a group of comments that represent normality (troll comments), and decide whether some comment is *Troll* or *Not Troll* measuring its deviation from the group.

In order to measure the similarity between different comments, we computed the following distance measures:

- **Euclidean Distance.** This distance is the length of the line segment connecting two points. It is calculated as: $d(x,y) = \sum_{i=0}^{n} \sqrt{x_i^2 - y_i^2}$ where x is the first point; y is the second point; and x_i and y_i are the i^{th} component of the first and second point, respectively.
- **Manhattan Distance.** This distance between two points x and y is the sum of the lengths of the projections of the line segment between the two points onto the coordinate axes: $d(x,y) = \sum_{i=0}^{n} |x_i - y_i|$ where x is the first point; y is the second point; and x_i and y_i are the i^{th} component of the first and second point, respectively.

These distances provide a method for measuring the deviation between 2 comments (i.e., the distance between any comment and one single comment in the group that represents normality: troll). In order to be able to compare a single comment against a group of various comments, it is necessary to apply a distance selection rule to obtain a unique value dependant on every distance

[4] Available at `http://www.cs.uic.edu/~liub/FBS/opinion-lexicon-English.rar`

measure performed. To this end, we employ 3 different rules: (i) **Mean selection rule** computes the average of the distances to all the members of the normal group, (ii) **Max. selection rule** returns the distance to the furthest point in the normality representation and (iii) **Min. selection** rule selects the distance to the nearest normal comment.

The final deviation value of the comment under inspection depends on the distance measure computed and the selection rule applied. Therefore, when our method inspects a comment a final distance value is acquired, which will depend on both the distance measure and the combination metric.

4 Empirical Validation

This section describes the validation of our approach against a comment dataset gathered from 'Menéame'. We gathered a collection of comments from the 5th of April, 2011 to 12th of April, 2011. This dataset of comments comprises one week of stories filled by 9,044 comment instances.

We labelled each of the comments in one category: Controversy level. This category refers to a comment can be *Not Troll* or *Troll*. *Not Troll* means that the comment is not hurtful or hurting, using in its argument a restrained tone. Moreover, *Troll* refers to a comment which seeks to create polemic in a exaggerated way. To this end, we built a dataset, following the next distribution: 6,857 examples of 'not troll' comments and 2,187 of troll comments.

4.1 Methodology

In order to extract all the features described in Section 2, we developed two different procedures to construct the VSM of the comment body: (i) VSM with words and terms, and (ii) n-grams with different values of n (n=1, n=2, n=3). Furthermore, we removed every word devoid of meaning in the text, called stop words, (e.g., 'a','the','is') [15]. To this end, we employed an external stop-word list of Spanish words[5]. Subsequently, we evaluated the precision of our proposed method. To this end, we conducted the following methodology:

1. **Cross validation.** We performed a 5-fold cross-validation [17] to divide the troll comment dataset into 5 different divisions of 1750 comments for representing normality and 437 for measuring deviations. In this way, each fold is composed of 1,750 troll comments that will be used as representation of normality and 1,808 testing comments, from which 437 are troll comments and 1371 are 'not troll' comments.
2. **Calculating distances and combination rules.** We extracted the aforementioned features and employed the 2 different measures and the 3 different combination rules described in Section 3 to obtain a final measure of deviation for each testing evidence. More accurately, we applied the following

[5] The list of stop words can be downloaded at: `http://paginaspersonales.deusto.es/isantos/resources/stopwords.txt`

distances: (i) Euclidean Distance and (ii) Manhattan Distance. For the combination rules we tested the following: (i) the mean value, (ii) the lowest distance and (iii) the highest value.

3. **Defining thresholds.** For each measure and combination rule, we established 10 different thresholds to determine whether a comment is troll or not. These thresholds were selected by first establishing the lowest one. This number was the highest possible value with which no troll comments were misclassified. The highest one was selected as the lowest possible value with which no 'not troll' comments were misclassified. The rest of thresholds were selected by equally dividing the range between the first and the last threshold. In this way, the method is configurable in both reducing false positives or false negatives.
4. **Testing the method.** We measured the precision of the troll comments identification as the number of correctly classified troll comments divided by the sum of the number of correctly classified troll comments and the number of 'not troll' comments misclassified as troll:

$$Precision = \frac{N_{t \to t}}{N_{t \to t} + N_{nt \to t}} \tag{1}$$

where $N_{t \to t}$ is the number of correctly classified troll comments and $N_{nt \to t}$ is the number of 'not troll' comments misclassified as troll. Additionally, we measured the recall of the troll comments, which is the number of correctly classified troll comments divided by the number of correctly classified troll comments and the number of troll comments misclassified as 'not troll':

$$Recall = \frac{N_{t \to t}}{N_{t \to t} + N_{t \to nt}} \tag{2}$$

We also computed the f-measure, which is the harmonic mean of both the precision and recall, simplified as follows:

$$F - measure = \frac{2N_{t \to t}}{2N_{t \to t} + N_{nt \to t} + N_{t \to nt}} \tag{3}$$

4.2 Results

We compared the detection capabilities of our method with some of the most used supervised machine-learning algorithms. Specifically, we use the next ones:

- *Bayesian networks (BN):* We used different structural learning algorithms: K2 [18] and Tree Augmented Naïve (TAN) [19]. Moreover, we also performed experiments with a Naïve Bayes Classifier [20].
- *Support Vector Machines (SVM):* We launched with a polynomial kernel [21], a normalised polynomial kernel [22], a Pearson VII function-based universal kernel (PUK) [23] and radial basis function (RBF) based kernel [24].
- *K-nearest neighbour (KNN):* We experimented with $k = 10$.

Table 1: Best results for different combination rules and distance measures in terms of Threshold (Thres.), Precision (Prec.), Recall (Rec.) and F-Measure (F-Mea.) of the Controversy Level for word VSM approach.

Metric	**Euclidean Distance**				**Manhattan Distance**			
	Thres.	**Prec.**	**Rec.**	**F-Mea.**	**Thres.**	**Prec.**	**Rec.**	**F-Mea.**
Mean	13749260	61.41%	99.81%	76.04%	24215126	61.34%	99.50%	75.89%
Maximum	19132703	61.46%	100.00%	76.13%	37752636	60.62%	96.52%	74.47%
Minimum	9884322	60.89%	97.64%	75.01%	13948167	60.82%	97.36%	74.87%

Table 2: Best results for different combination rules and distance measures in terms of Threshold (Thres.), Precision (Prec.), Recall (Rec.) and F-Measure (F-Mea.) of the Controversy Level for N-gram VSM approach.

Metric	**Euclidean Distance**				**Manhattan Distance**			
	Thres.	**Prec.**	**Rec.**	**F-Mea.**	**Thres.**	**Prec.**	**Rec.**	**F-Mea.**
Mean	19074560	61.40%	99.73%	76.00%	46329394	60.97%	97.94%	75.15%
Maximum	26596267	61.46%	100.00%	76.13%	85973317	60.79%	97.21%	74.80%
Minimum	16713191	61.25%	99.11%	75.71%	36454947	60.82%	97.34%	74.86%

Table 3: Best results for Precision (%), Recall (%) and F-Measure (%) of the Controversy Level for Word VSM and N-gram VSM, using Supervised Machine-learning algorithms.

VSM Approach	**Classifier**	**Precision**	**Recall**	**F-Measure**
Words as terms	SVM:Normal. Polykernel	84.92%	95.77%	90.02%
N-grams as terms	BayesNet TAN	77.95%	97.89%	86.79%

– *Decision Trees (DT):* We executed experiments with J48 (the *Weka* [25] implementation of the *C4.5* algorithm [26]) and Random Forest [27], an ensemble of randomly constructed decision trees. We employed $N = 100$.

Table 1 shows the best results achieved with words as tokens when we consider troll comments as 'normality'. Table 2 shows the best results achieved with n-gram VSM approach. Table 3 shows the best results applying both VSM approaches: words as terms and N-grams as terms, and using the supervised machine-learning classifiers.

Regarding the results obtained in the Table 1 (in F-measure terms), the best result in anomaly detection when a troll comment indicates normality were offered by the the Euclidean Distance, with the maximum combination rule and 19132703 as threshold: 61.46% of precision, 100% of recall and 76.13% of f-measure. Moreover, in Table 2, the highest result was obtained the Euclidean Distance with the maximum combination rule, this time with a 26596267 threshold: 61.46% of precision, 100% of recall and 76.13% of f-measure. Finally, employing supervised machine-learning methods, in the Table 3 the highest result was achieved by Word VSM approach, using a SVM with a normalised polynomial kernel as classifier: 84.92% of precision, 95.77% of recall and 90.02% of f-measure. With regards to the use of anomaly classification, comparing with the supervised approaches, it achieved close results. We can maintain the results of the best supervised learning algorithm whilst the labelling efforts are reduced significantly, in this case a 75% of the dataset.

5 Conclusions

In our previous approach [13], we categorised the comments made by users using supervised machine-learning techniques. This method may be employed by administrators of webpages in order to moderate their website. For instance, it can be used to adequate the comments and visualisation of the page regarding the viewer, filter content that may damage the brand image of the page and also to categorise the users via their comments.

However, the use of the classic machine-learning-based text categorization and filtering have a very time-consuming step of labelling text. In our case, a previous work of comments labelling is required. This process in the field of web filtering can suppose a great inconvenient of performance overhead due to the number of new comments that appear everyday.

In this paper, we have proposed the first anomaly-detection-based trolling comments filtering method that based upon statistical, syntactic and opinion features, that is able to determine when a comment is troll or not. The results show that considering troll comments as base model (denominated 'normality') achieved a close performance average, in terms of f-measure, than supervised machine-learning approach, while the efforts of labelling are minimising.

The avenues of future work are oriented in three main ways. Firstly, applying additional algorithms to extend the study of filtering trolling comments in social websites. Secondly, incorporating new different extracted features from the comment dataset to train the models. And finally, we will improve the anomaly method scalability to reduce the number of distance computations required.

References

1. OReilly, T.: What is web 2.0: Design patterns and business models for the next generation of software. Communications & strategies (1) (2007) 17
2. Dadvar, M., Trieschnigg, D., Ordelman, R., de Jong, F.: Improving cyberbullying detection with user context. In: Advances in Information Retrieval. Springer (2013) 693–696
3. Smith, P.K., Mahdavi, J., Carvalho, M., Fisher, S., Russell, S., Tippett, N.: Cyberbullying: Its nature and impact in secondary school pupils. Journal of Child Psychology and Psychiatry **49**(4) (2008) 376–385
4. Dinakar, K., Reichart, R., Lieberman, H.: Modeling the detection of textual cyberbullying. In: The Social Mobile Web. (2011)
5. Shachaf, P., Hara, N.: Beyond vandalism: Wikipedia trolls. Journal of Information Science **36**(3) (2010) 357–370

6. Bergstrom, K.: dont feed the troll: Shutting down debate about community expectations on reddit. com. First Monday **16**(8) (2011)
7. Fisher, D., Smith, M., Welser, H.T.: You are who you talk to: Detecting roles in usenet newsgroups. In: System Sciences, 2006. HICSS'06. Proceedings of the 39th Annual Hawaii International Conference on. Volume 3., IEEE (2006) 59b–59b
8. Lea, M., O'Shea, T., Fung, P., Spears, R.: 'Flaming'in computer-mediated communication: Observations, explanations, implications. Harvester Wheatsheaf (1992)
9. Postmes, T., Spears, R., Lea, M.: Breaching or building social boundaries? side-effects of computer-mediated communication. Communication research **25**(6) (1998) 689–715
10. Lerman, K.: User participation in social media: Digg study. In: Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Workshops, IEEE Computer Society (2007) 255–258
11. Jindal, N., Liu, B.: Review spam detection. In: Proceedings of the 16th international conference on World Wide Web, ACM (2007) 1189–1190
12. Jindal, N., Liu, B.: Opinion spam and analysis. In: Proceedings of the international conference on Web search and web data mining, ACM (2008) 219–230
13. Santos, I., de-la Peña-Sordo, J., Pastor-López, I., Galán-García, P., Bringas, P.: Automatic categorisation of comments in social news websites. Expert Systems with Applications (2012)
14. Baeza-Yates, R.A., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1999)
15. Salton, G., McGill, M.: Introduction to modern information retrieval. McGraw-Hill New York (1983)
16. Tata, S., Patel, J.M.: Estimating the selectivity of tf-idf based cosine similarity predicates. ACM SIGMOD Record **36**(2) (2007) 75–80
17. Kohavi, R., et al.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: IJCAI. Volume 14. (1995) 1137–1145
18. Cooper, G.F., Herskovits, E.: A bayesian method for constructing bayesian belief networks from databases. In: Proceedings of the 1991 conference on Uncertainty in artificial intelligence. (1991)
19. Geiger, D., Goldszmidt, M., Provan, G., Langley, P., Smyth, P.: Bayesian network classifiers. In: Machine Learning. (1997) 131–163
20. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press (1995)
21. Amari, S., Wu, S.: Improving support vector machine classifiers by modifying kernel functions. Neural Networks **12**(6) (1999) 783–789
22. Maji, S., Berg, A., Malik, J.: Classification using intersection kernel support vector machines is efficient. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE (2008) 1–8
23. Üstün, B., Melssen, W., Buydens, L.: Visualisation and interpretation of support vector regression models. Analytica chimica acta **595**(1-2) (2007) 299–309
24. Cho, B., Yu, H., Lee, J., Chee, Y., Kim, I., Kim, S.: Nonlinear support vector machine visualization for risk factor analysis using nomograms and localized radial basis function kernels. IEEE Transactions on Information Technology in Biomedicine **12**(2) (2008) 247–256
25. Garner, S.: Weka: The waikato environment for knowledge analysis. In: Proceedings of the 1995 New Zealand Computer Science Research Students Conference. 57–64
26. Quinlan, J.: C4.5 programs for machine learning. Morgan Kaufmann (1993)
27. Breiman, L.: Random forests. Machine learning **45**(1) (2001) 5–32