1st International Electronic Conference on Entropy and Its Applications
3 - 21 November 2014

Variations of Neighbor Diversity for Fraudster Detection in Online Auction

Laksamee Khomnotai (Nakhon Ratchasima Rajabhat University)

Jun-Lin Lin (Yuan Ze University)

Introduction (1)

- Online shopping/auction websites have attracted both legitimate users and fraudsters.
- To evaluate the trustworthyness of a user, online shopping/auction websites often provide a reputation system
    - The reputation system requests the buyer and the seller of a transaction to give each other a rating
    - Users with higher reputation scores are more trustworthy, and consequently are more likely to attract sales
- To gain the higher reputation in a short period of time, fraudsters often commits the so-called “inflated reputation fraud”

Introduction (2)

- The inflated reputation fraud is accomplished by a group of collusive users who conduct many fake transactions for low-price merchandises and give each other good ratings
- In our recent work, we adopted Shannon entropy to quantify the neighbor diversity
- However, different ways to define and calculate diversity exist in the literature
- In this study, we adopt the four different definitions of diversity to calculate the neighbor diversity

Related Work (1)

- The earlier approaches used the properties derived from the transaction history, e.g. sum, average, and standard deviation of buying or selling price of merchandises in a period of time
- Most of the recent approaches used SNA to detect group of fraudsters
  - The characteristics such as *k*-plex, clique, betweenness, and *k*-core are often used to detect cohesive groups
  - *k*-core has been found to be the most effective for detecting fraudsters
  - Fraudsters frequently usually appear in *k*-core with $k \geq 2$

Related Work (2)

- Problem with *k*-core
  - Using *k*-core alone results in low precision
  - Applying both center weight (CW) and *k*-core improves the precision, but the recall is reduced
- Neighbor diversity
  - It was proposed to improve both precision and recall
  - The neighbor diversity on the number of received ratings provides an effective way to discern fraudsters from normal users

Variants of Neighbor Diversity (1)

- x denote a user
- x's neighbors are the users who gave at least one rating to x
- The neighbors of x are partitioned into several classes based on the number of received ratings
  - r denote the number of received ratings of a userIf $0 \leq r < 50$, then the user is placed into class 1
  - If $50 \times 2^{i-2} \leq r < 50 \times 2^{i-1}$, then the user is placed into class i, where $i > 1$
- $p_i(x)$ denote the proportion of the x's neighbors in the i-th class, and n denote the total number of classes. Then, all diversity constraints must hold:

$$0 \leq p_i(x) \leq 1, \text{for } i = 1 \text{ to } n$$

$$\sum_{i=1}^{n} p_i(x) = 1$$

Variants of Neighbor Diversity (2)

- **Shannon Entropy Diversity**
  - The neighbor diversity of *x* based on Shannon entropy is denoted as $D_s(x)$ and calculated as:

$$D_s(x) = -\sum_{i=1}^{n} p_i(x) \log_2 p_i(x)$$

- **Max Weight Diversity and Min Weight Diversity**
  - The max weight diversity, denoted as $D_{max}(x)$, is the maximum of all $p_i(x)$ for i=1 to n, and defined as:

$$D_{max}(x) = \max_{i=1 \ to \ n} p_i(x)$$

  - The min weight diversity, denoted as $D_{min}(x)$, is calculated using the minimum of all $p_i(x)$ for i=1 to n, and defined as:

$$D_{min}(x) = 1 + (1 - n) \min_{i=1 \ to \ n} p_i(x)$$

Variants of Neighbor Diversity (3)

- **Canonical L^p-norm Diversity**
  - The Canonical L^p-norm diversity, denoted as $D_{pow}(x)$, is similar to the L^p-norm except the outer exponent is $\frac{1}{pow-1}$ instead of $\frac{1}{pow}$, as shown below:

$$D_{pow}(x) = \left(\sum_{i=1}^{n} |p_i(x)|^{pow}\right)^{\frac{1}{pow-1}}$$

- **Canonical Shannon Entropy Diversity**
  - The max weight diversity, denoted as $D_{cs}(x)$ and defined as:

$$D_{cs}(x) = e^{-D_s(x)} = e^{\sum_{i=1}^{n} p_i(x) \log_2 p_i(x)}$$

Experimental Settings (1)

- Data was collected from Ruten (www.ruten.com.tw), which is one of the largest online auction websites in Taiwan
- The dataset grows from a list of suspended users, and then conducts a level-wise expansion to include more users
- The dataset consists of 4,407 users
  - 1,080 are fraudsters
  - 3,327 are non-fraudsters (i.e. normal accounts)

Experimental Settings (2)

- Each neighbor diversity was calculate (i.e. $D_s(x)$, $D_{max}(x)$, $D_{min}(x)$, $D_2(x)$, $D_3(x)$ and $D_{cs}(x)$) and used to build the classifier
- Three classification algorithms from Weka were used to perform 10-fold cross-validation
  - J48 decision tree
  - Neural Networks (NN)
  - Support Vector Machine (SVM)

Experimental Results (1)

- Part one
  - Used only one of the neighbor diversities to build classifiers
  - The results are shown in Tables 1, 2 and 3
  - The best results of each classification algorithm are shown in bold
  - D_{min} performs the worst
  - D_{max} performs the best

Experimental Results (2)

- Table 1 J48 Performance (Part one)

Diversity	Accuracy(%)	Recall	Precision	F_1-measure
D_s	84.1843	0.8019	0.6420	0.7131
D_{max}	84.1616	0.8009	0.6417	0.7125
D_{min}	82.0059	0.6639	0.6251	0.6439
D_2	84.1162	0.7944	0.6422	0.7103
D_3	84.1162	**0.8028**	0.6403	0.7124
D_{cs}	**84.2523**	**0.8028**	**0.6432**	**0.7142**

Experimental Results (3)

- Table 2 Neural Network performance (Part one)

Diversity	Accuracy(%)	Recall	Precision	F_1-measure
D_s	83.1405	0.7620	0.6287	0.6890
D_{max}	**83.8212**	**0.8120**	0.6323	**0.7110**
D_{min}	82.0286	0.6648	0.6254	0.6445
D_2	83.7077	0.7870	**0.6353**	0.7031
D_3	83.7985	0.7991	0.6346	0.7074
D_{cs}	83.5943	0.7713	0.6364	0.6974

Experimental Results (4)

- Table 3 Support Vector Machine performance (Part one)

Diversity	Accuracy(%)	Recall	Precision	F_1-measure
D_s	83.1405	0.7306	0.6358	0.6799
D_{max}	**83.5716**	**0.7556**	**0.6395**	**0.6927**
D_{min}	82.0059	0.6639	0.6251	0.6439
D_2	83.0270	0.7222	0.6352	0.6759
D_3	83.2539	0.7361	0.6370	0.6830
D_{cs}	82.6639	0.6944	0.6334	0.6625

Experimental Results (5)

- Part one
  - Used *k*-core and CW and one of the neighbor diversities to build classifiers
  - The results are shown in Tables 4, 5 and 6
  - Compared to Part one, the addition of *k*-core and CW slightly improves
  - The improvement on accuracy is most significant with J48
  - The improvement on accuracy is less significant with NN and SVM

Experimental Results (6)

- Table 4 J48 Performance (Part two)

Diversity	Accuracy(%)	Recall	Precision	F_1-measure
k-core+CW+D_s	85.8180	0.8731	0.6590	0.7511
k-core+CW+D_{max}	85.8861	0.8731	0.6604	0.7520
k-core+CW+D_{min}	84.1162	0.8278	0.6349	0.7186
k-core+CW+D_2	86.1130	0.8685	0.6662	0.7540
k-core+CW+D_3	**86.2038**	0.8704	**0.6676**	**0.7556**
k-core+CW+D_{cs}	85.8180	**0.8741**	0.6588	0.7513

Experimental Results (7)

- Table 5 Neural Network performance (Part two)

Diversity	**Accuracy(%)**	**Recall**	**Precision**	**F_1-measure**
k-core+CW+D_s	83.7758	0.7787	0.6386	0.7017
k-core+CW+D_{max}	**84.1616**	**0.8083**	**0.6400**	**0.7144**
k-core+CW+D_{min}	82.3916	0.6620	0.6350	0.6482
k-core+CW+D_2	83.9120	0.7981	0.6371	0.7086
k-core+CW+D_3	83.9573	0.8028	0.6370	0.7104
k-core+CW+D_{cs}	83.8212	0.7843	0.6383	0.7038

Experimental Results (8)

- Table 5 Neural Network performance (Part two)

Diversity	Accuracy(%)	Recall	Precision	F_1-measure
k-core+CW+D_s	83.7758	0.7787	0.6386	0.7017
k-core+CW+D_{max}	**84.1616**	**0.8083**	**0.6400**	**0.7144**
k-core+CW+D_{min}	82.3916	0.6620	0.6350	0.6482
k-core+CW+D_2	83.9120	0.7981	0.6371	0.7086
k-core+CW+D_3	83.9573	0.8028	0.6370	0.7104
k-core+CW+D_{cs}	83.8212	0.7843	0.6383	0.7038

Experimental Results (7)

- Table 6 Support Vector Machine performance (Part two)

Diversity	Accuracy(%)	Recall	Precision	F_1-measure
k-core+CW+D_s	**84.4112**	**0.7685**	**0.6551**	**0.7073**
k-core+CW+D_{max}	83.0043	0.6835	0.6428	0.6625
k-core+CW+D_{min}	83.4581	0.7630	0.6353	0.6933
k-core+CW+D_2	83.2539	0.7370	0.6368	0.6833
k-core+CW+D_3	83.2993	0.7398	0.6372	0.6847
k-core+CW+D_{cs}	83.0951	0.7426	0.6320	0.6828

Conclusions

- This paper proposes to use various methods to calculate diversity, and study whether these methods cause significant difference on the classification performance of fraudster detection
- The experimental results show that the diversity D_{min} performs the worst.
- The remaining five diversities (i.e., D_s, D_{max}, D_2, D_3 and D_{cs}) achieve similar performance
- The addition of *k*-core and CW only slightly improves the classification performance of the neighbor diversity

Future Study

- Finding new features to work better with the neighbor diversity for fraudster detection is planned for future work