SOFTWARE DEFECT PREDICTION WITH BAYESIAN APPROACHES

การทำนายข้อบกพร่องของซอฟต์แวร์ด้วยแนวทางแบบเบย์

LINK

นายปัญญวัตร สุวรรณทัต

6614450042

Software Defect Prediction with Bayesian Approaches.

Authors:

Hernández-Molinos, María José[1] (AUTHOR) *majohdezmol@gmail.com*
Sánchez-García, Angel J.[1] (AUTHOR) *juaperez@uv.mx*
Barrientos-Martínez, Rocío Erandi[2] (AUTHOR) *rbarrientos@uv.mx*
Pérez-Arriaga, Juan Carlos[1] (AUTHOR) *jocharan@uv.mx*
Ocharán-Hernández, Jorge Octavio[1] (AUTHOR)

Source:

Mathematics (2227-7390). Jun2023, Vol. 11 Issue 11, p2524. 18p.

เป้าหมายงานวิจัย

- งานวิจัยนี้มีเป้าหมายที่จะประเมิน Algorithm 3 ตัว คือ Bayesian Networks, Decision Tree และ Random Forest เพื่อจำแนกว่า Project ใดๆ มีความเสี่ยงต่อข้อบกพร้อง (Software defects)

แรงจูงใจ

- การทำงานนี้มีจุดมุ่งหมายเพื่อประโยชน์ให้แก่ Software Engineer ในการสร้าง Model predict ข้อบกพร่องที่แม่นยำ การทำนายข้อบกพร่องที่ดีช่วยให้พวกเขาสามารถระบุพื้นที่และ Module ของ Software ที่มีความเสี่ยงต่อข้อบกพร่องได้ง่ายขึ้น

ที่มาของปัญหา

- Herzig et al. ได้กล่าวถึงผลวิจัยที่ทำโดยการตรวจสอบข้อมูลประมาณ 7,000 รายการ จากฐานข้อมูลข้อบกพร่องของ 5 open-source projects พบว่า 33.8% ของรายงานทั้งหมดถูกจัดลำดับผิดเนื่องจากไม่มีข้อบกพร่องจริง
- ไม่พบการใช้ Bayesian Networks ในงานที่อ้างถึง แต่พบว่ามีอัลกอริทึม Naive Bayes แบบคลาสสิก

แนวทางการวิจัย

- เลือกใช้ Bayesian Networks เนื่องจากความสามารถในการแสดงความสัมพันธ์ระหว่างตัวแปร มีการแสดงผลที่กระชับ มีความยืดหยุ่น และสามารถอ่านความสัมพันธ์แบบตรงระหว่างตัวแปรร่วมกัน กล่าวคือแสดงความสัมพันธ์ของข้อมูลได้ดี
- Bayesian Networks เป็นวิธีการที่ยืดหยุ่นและสามารถจัดการกับประเภทต่างๆ ของตัวแปรได้ รวมถึงตัวแปรแบบต่อเนื่องและแบบไม่ต่อเนื่อง ซึ่งไม่จำกัดประเภทของข้อมูลที่ได้จากการวัดค่าทางซอฟต์แวร์

DATA SET

DATA SET

- ชุดข้อมูลที่ใช้ในการประเมินอัลกอริทึมที่เลือกได้มาจากคลังข้อมูล PROMISE repository โดยเหตุผลที่เลือกใช้ชุดข้อมูลเหล่านี้คือเนื่องจากเป็นข้อมูลสาธารณะ และเป็นชุดข้อมูลที่ถูกใช้มากที่สุดในการทำนายข้อบกพร่องของซอฟต์แวร์

CM1 เป็นเครื่องมือในยานอวกาศ NASA ที่เขียนด้วยภาษา "C“

JM1 เขียนด้วย "C" และเป็นระบบภาคพื้นดินคาดการณ์แบบ real time

KC1 คือระบบ "C++" ที่ใช้การจัดการพื้นที่เก็บข้อมูลสำหรับการรับและประมวลผลข้อมูลภาคพื้นดิน

Class Target

Table 3. Distribution of classes by data set.

Data Set	Number of Instances	Distributions of Class (Defects)	
		False	True
CM1	498	90.16%	9.83%
JM1	10,885	19.35%	80.65%
KC1	2109	15.45%	84.54%

DATA SET

- ชุดข้อมูลมีตัวแปรหรือคุณลักษณะทั้งหมด 21 รายการ ถูกแบ่งเป็นหมวดหมู่และอธิบายดังนี้

Table 4. Distribution of attribute types.

Type of Attributes	Number of Metrics
Line of Code	5
McCabe measure	3
Base Halstead measure	4
Derived Halstead measure	8
Branch count	1
Total	21

- **Line of Code (LOC):** จำนวนบรรทัดของโค้ด project
- **McCabe measure:** มาตรวัดจำนวนบรรทัด และมาตรวัดความซับซ้อน
- **Base Halstead measure:** เป็นวิธีการทางซอฟต์แวร์ที่ใช้ในการวัดความซับซ้อนของโค้ด project
- **Derived Halstead measure:** การคำนวณค่า Base Halstead measure ใช้ในการวิเคราะห์ประเมินคุณภาพของ project
- **Branch count:** จำนวน Branch ใช้เพื่อวัดความซับซ้อน project

MODEL

BAYESIAN APPROACH

- ทฤษฎีบทของเบย์เป็นข้อเสนอที่ใช้ในการคำนวณความน่าจะเป็นแบบมีเงื่อนไขของเหตุการณ์ ได้รับการพัฒนาโดยนักคณิตศาสตร์และนักเทววิทยาชาวอังกฤษ Thomas Bayes วัตถุประสงค์หลักของทฤษฎีบทนี้คือเพื่อกำหนดความน่าจะเป็นของเหตุการณ์หนึ่งโดยเปรียบเทียบกับความน่าจะเป็นของเหตุการณ์อื่นที่คล้ายคลึงกัน

BAYESIAN NETWORKS

- Bayesian Networks เป็น Model ที่ใช้ Bayesian inference ในการคำนวณความน่าจะเป็นแบบมีเงื่อนไข และแสดงเส้นเชื่อมกราฟแบบมีทิศทางในแต่ละเหตุการณ์

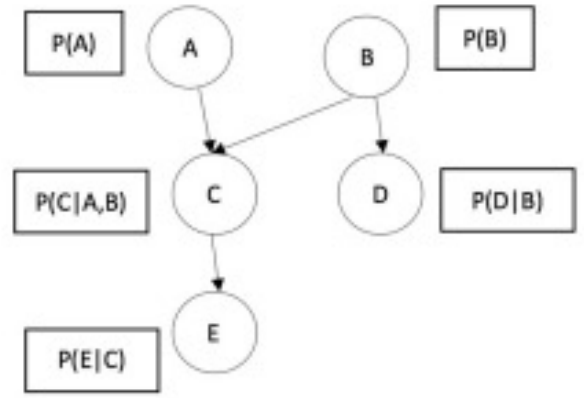

Figure 1. Bayesian Network as a DAG.

BAYESIAN NETWORKS

- การสร้าง Bayesian Network ไม่มีวิธีเดียวเสมอ งานวิจัยนี้นำเสนอวิธีการสร้าง Bayesian Network ทั้งหมด 3 ได้แก่ TAN, Hill Climbing, K2

TAN (TREE AUGMENTED NAÏVE BAYESIAN NETWORK)

- Algorithm TAN เป็น Bayesian Network ที่สร้างโครงสร้างแสดงเชื่อมโยงระหว่างตัวแปรที่ต้องการทำนาย ความน่าจะเป็นของตัวแปรเหล่านี้จะถูกคำนวณโดยใช้ Bayes' theorem โดยอิงตามความน่าจะเป็นของ class taget

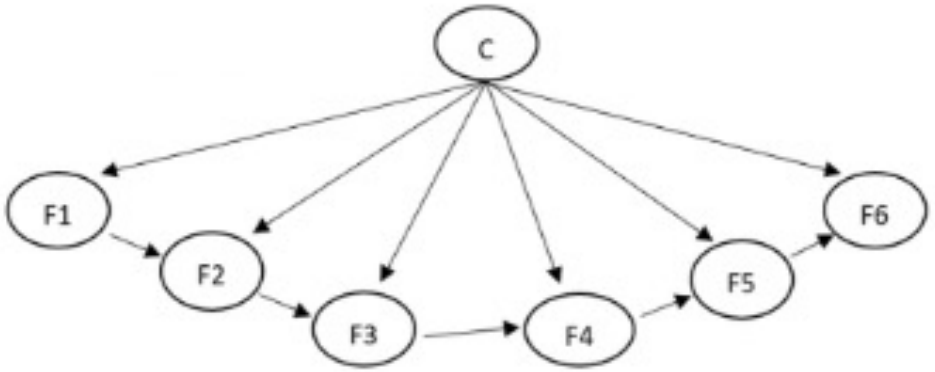

Figure 2. Bayesian network structure initialized using TAN.

HILL CLIMBING

- Algorithm นี้จะเพิ่มหรือลบความสัมพันธ์สำหรับแต่ละ Node หรือคุณลักษณะอย่างสุ่ม โดยคำนวณความน่าจะเป็นของแต่ละ Node ที่ประกอบด้วยในเครือข่ายจากความน่าจะเป็นร่วมของ class target
- Algorithm จะเลือกเครือข่ายที่เหมาะสมที่สุดด้วยคุณภาพที่ดีที่สุด โดยกำจัดเครือข่ายที่ไม่เข้าเกณฑ์

K2

- Algorithm K2 ใช้แนวคิดของ Algorithm greedy
- จะเริ่ม Node จากตัวแปรที่มีความน่าจะเป็นต่ำสุด กำหนดเป็น Node แรก และ Node ต่อๆ ไปจะมีความน่าจะเป็นสูงขึ้นตามลำดับ กระบวนการนี้ทำซ้ำจนกว่า Performance ไม่เพิ่มขึ้นอีก จึงจะหยุดทำงาน

การทดลองและผลลัพธ์

- การทดลองถูกดำเนินการบน Weka 3.9.6 รันบนระบบปฏิบัติการ Windows 10 ด้วย CPU Intel Core i7 3.6 GHz และ RAM 8 GB
- ชุด Parameter ที่กำหนดบน Weka software

Table 8. Setup of the experimental parameters in Weka software.

Parameter	Value	Search Algorithm
Batch size	100	All
Score Type	MDL	All
Random order	False	K2
Init as Naïve Bayes	True	K2
Use Arc Reversal	False	Hill Climbing
Markov Blanket Classifier	False	K2 and Hill Climbing

Parameter บางตัวไม่สามารถใช้ได้กับ Search Algorithm

การทดลองและผลลัพธ์

- แสดงการเปรียบเทียบความแม่นยำระหว่าง Algorithm ที่เสนอกับตัวจำแนกอื่น ๆ เช่น Decision Tree และ Random Forest ซึ่งสามารถเห็นได้ว่าตัวจำแนกสองตัวสุดท้ายได้รับค่าความแม่นยำสูงกว่าจากตัวจำแนกแบบ Bayesian
- อย่างไรก็ตาม การทดสอบ Cross-validation บ่งชี้ให้เห็นว่า Decision Tree และ Random Forest มีความแปรปรวนสูงและผลลัพธ์ไม่คงที่
- จาก Dataset JM1 ผลลัพธ์จากตัวจำแนกมีความสมดุลมากขึ้น ตัวจำแนก TAN ได้ผลลัพธ์ที่สูงกว่าตัวจำแนก Decision Tree แต่ Random Forest ยังคงสูงสุด

Table 13. Accuracy results for the different data sets with other approaches.

Algorithm	CM1		JM1		KC1	
	Best Accuracy	Standard Deviation	Best Accuracy	Standard Deviation	Best Accuracy	Standard Deviation
K2	0.9183	0.563	0.8079	0.454	0.8483	0.225
Hill Climbing	0.9183	0.835	0.8079	0.454	0.8862	1.526
TAN	0.92	4.077	0.8236	0.767	0.8815	1.887
Decision Tree	0.94	2.865	0.8170	0.886	0.8957	1.904
Random Forest	0.94	2.084	0.8382	0.808	0.9004	1.547

END
THANK YOU.