รายงานวิจัยฉบับสมบูรณ์

โครงการอัลกอริทึมที่รับประกันคุณภาพคำตอบ สำหรับปัญหาการหาค่าที่ดีที่สุดเชิงการจัด การคำนวณเชิงออนไลน์ และการเรียนรู้ด้วยเครื่องจักร

โดย จิตร์ทัศน์ ฝักเจริญผล และคณะ

มีนาคม พ.ศ. 2553

สัญญาเลขที่ MRG5080318

รายงานวิจัยฉบับสมบูรณ์

อัลกอริทึมที่รับประกันคุณภาพคำตอบ สำหรับปัญหาการหาค่าที่ดีที่สุดเชิงการจัด การคำนวณเชิงออนไลน์ และการเรียนรู้ด้วยเครื่องจักร

Algorithms with performance guarantees for combinatorial optimization, on-line computation, and machine learning

โดย

นายจิตร์ทัศน์ ฝักเจริญผล **หัวหน้าโครงการวิจัย**
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์
50 พหลโยธิน แขวงลาดยาว เขตจตุจักร กรุงเทพ 10900

นายบุญเสริม กิจศิริกุล **นักวิจัยที่ปรึกษา**
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
254 ถนนพญาไท แขวงวังใหม่ เขตปทุมวัน กทม. 10330

สนันสนุนโดยสำนักงานคณะกรรมการการอุดมศึกษา
และสำนักงานกองทุนสนับสนุนการวิจัย

Abstract (บทคัดย่อ)

Project Code : MRG5080318
(รหัสโครงการ)

Project Title : Algorithms with performance guarantees for combinatorial optimization, on-line computation, and machine learning
(ชื่อโครงการ) อัลกอริทึมที่รับประกันคุณภาพคำตอบ สำหรับปัญหาการหาค่าที่ดีที่สุดเชิงการจัด การคำนวณเชิงออนไลน์ และการเรียนรู้ด้วยเครื่องจักร

Investigator : Jittat Fakcharoenphol
(ชื่อนักวิจัย) นายจิตร์ทัศน์ ฝักเจริญผล

E-mail Address : jtf@ku.ac.th

Project Period : 2 ปี
(ระยะเวลาโครงการ)

We present two results related to machine learning and online computation.

The first result presents short proofs on the mistake bounds of the 1-nearest neighbor algorithm on an online prediction problem of path labels. The algorithm is one of key ingredients in the algorithm by Herbster, Lever, and Pontil for general graphs. Our proofs are combinatorial and naturally show that the algorithm works when the set of labels is not binary.

The second result is related to our previous work on learning reductions. We present a counter example showing that an algorithm for constructing multiclass predictors from binary predictors cannot preserve the performance of the binary predictors. Through this example, we discover that our previous result contains errors.

Keywords: Multiclass learning, On-line learning, Algorithms

เนื้อหางานวิจัย

1. ความสำคัญและที่มาของปัญหาที่ทำการวิจัย

อัลกอริทึม คือวิธีการที่ชัดเจนในการหาคำตอบของปัญหาหนึ่ง ๆ อย่างไรก็ตาม ปัญหาส่วนมากมิได้ต้องการแค่ “คำตอบ” เท่านั้น หากแต่ต้องการคำตอบที่ “ดีที่สุด” ซึ่งอาจจะเป็นในแง่ของความถูกต้อง หรือการเป็นคำตอบที่มีค่าใช้จ่ายน้อยที่สุด เป็นต้น

อย่างไรก็ตาม เราไม่จำเป็นที่จะต้องออกแบบอัลกอริทึมที่คำนวณหาคำตอบที่ดีที่สุดได้เสมอไป ทั้งนี้อาจมีสาเหตุเช่น

1. ความ “ยาก” ของปัญหา กล่าวคือปัญหาที่เราต้องการหาคำตอบเป็นปัญหาที่อยู่ในกลุ่ม NP-hard ซึ่งเราไม่คาดหวังว่าจะมีอัลกอริทึมที่มีประสิทธิภาพ (polynomial-time algorithm) สำหรับหาคำตอบ

2. ความไม่ครบถ้วนของข้อมูล กล่าวคือ ในการหาคำตอบนั้น เราไม่ได้รับข้อมูลป้อนเข้าที่ครบถ้วน ทั้งนี้เนื่องมาจากลักษณะของปัญหา

อย่างไรก็ตาม อุปสรรคทั้งสองประเภททำให้เกิดกลุ่มของปัญหาที่น่าสนใจ ทั้งในทางปฏิบัติ (การนำไปใช้จริง) และในเชิงทฤษฎี กล่าวคือ

1. ปัญหาการหาค่าที่ดีที่สุดที่ยากระดับเอ็นพี (NP-hard optimization problems) ตัวอย่างของปัญหาที่สำคัญที่มีลักษณะดังกล่าว เช่น ปัญหาคนขายของ (Traveling Salesman Problem), ปัญหาคนซ่อมของ (Traveling Repairman Problem), ต้นไม้สไตเนอร์ (Steiner Tree Problem), ตัวแบ่งกราฟ (Graph Separator), ต้นไม้ทอดข้ามที่จำกัดเส้นผ่านศูนย์กลาง (Bounded-diameter MST) ปัญหาเหล่านี้มีข้อจำกัดในแง่ของความยากของปัญหา

2. ปัญหาออนไลน์ (on-line problems) เป็นกลุ่มของปัญหาที่อัลกอริทึมจะต้องตัดสินใจบางอย่างก่อนที่จะเห็นข้อมูลป้อนเข้าทั้งหมด ตัวอย่างของปัญหาออนไลน์ เช่น ปัญหาการจัดการแคช (cache management) ปัญหาการจัดตารางการทำงานแบบออนไลน์ (on-line scheduling) หรือ ปัญหาบนเครือข่ายต่าง ๆ เช่น การหาเส้นทางเพื่อส่งข้อมูลในเครือข่าย (routing) หรือ การสร้างต้นไม้มัลติคาสต์แบบออนไลน์ (multicast tree) เป็นต้น

3. ปัญหาการเรียนรู้ด้วยเครื่องจักร (learning problems) ที่อัลกอริทึมจะเห็นแค่เพียงตัวอย่างสำหรับการเรียนรู้เท่านั้น เช่น ปัญหาการเรียนรู้จากตัวอย่าง (learning from examples หรือเรียกว่า supervised learning)

สำหรับปัญหามากมายที่อยู่ในกลุ่มเหล่านี้ ได้มีการพัฒนาอัลกอริทึมที่พยายามจะหาคำตอบให้ได้ “ใกล้เคียง” กับคำตอบที่ดีที่สุด โดยส่วนมากแล้ว การแสดงคุณภาพของอัลกอริทึมเหล่านี้จะใช้การทดลอง และเป็นที่น่าสนใจที่อัลกอริทึมเหล่านี้มักทำงานได้ดีในทางปฏิบัติ

ถึงแม้ว่าการวัดผลด้วยการทดลองจะสามารถยืนยันประสิทธิภาพของอัลกอริทึมได้ในระดับที่น่าพอใจ ผลลัพธ์ที่ได้จากการทดลองมักขึ้นกับกระบวนการที่ใช้ในการทดลอง โดยเฉพาะการเลือกหรือการสุ่มข้อมูลป้อนเข้า (input) ทำให้ผลที่วัดได้จำกัดอยู่ภายใต้โมเดลที่สร้างข้อมูลเหล่านั้น และหลาย ๆ ครั้ง นอกจากโมเดลสำหรับการสร้างข้อมูลป้อนเข้าจะไม่ถูกระบุออกมาอย่างชัดเจนแล้ว โมเดลดังกล่าวก็มักไม่ใช่ตัวแทนของข้อมูลป้อนเข้าจริง

ปัญหาที่สำคัญที่สุดของการออกแบบและวัดผลอัลกอริทึมด้วยการทดลองโดยทั่วไปก็คือ การทดลองไม่สามารถตอบคำถามว่า ทำไมอัลกอริทึมดังกล่าวจึงมีประสิทธิภาพดังที่ทดลองได้ แม้ว่าคำถามนี้จะไม่เป็นที่สนใจมากนักสำหรับผู้ใช้งานซึ่งอาจพอใจกับผลลัพธ์ที่ได้ และมักเป็นสิ่งที่ถูกข้ามผ่านไปโดยนักวิจัยเชิงทดลองด้วยเหตุผลต่าง ๆ แต่คำถามที่ลึกซึ้งเหล่านี้นอกจากจะเป็นพื้นฐานที่สำคัญสำหรับการพัฒนาอัลกอริทึมที่ดีขึ้นและทำให้เราวางแผนและประเมินความสำเร็จของแนวทางการวิจัยต่อไปได้แล้ว ความเข้าในธรรมชาติของปัญหาและขีดจำกัดของการคำนวณ ยังเป็นชิ้นส่วนเล็ก ๆ อีกชิ้นหนึ่ง ที่อาจช่วยให้เราเข้าใจธรรมชาติของ “การคำนวณ” ก็ได้

สำหรับงานวิจัยนี้ ผู้วิจัยได้ศึกษาปัญหาที่เกี่ยวข้องกับกลุ่มปัญหาทั้งสามที่กล่าวมาข้างต้น รวม 2 ปัญหา คือ

1. ปัญหาการการสร้างตัวจำแนกแบบหลายประเภทจากตัวจำแนกแบบทวิภาค ซึ่งเป็นปัญหาที่เกี่ยวข้องกับงานวิจัยเดิมที่เคยเสนอขอทุนวิจัยเมื่อปี พ.ศ. 2547 ของผู้วิจัย

การสร้างตัวจำแนนแบบทวิภาคเป็นปัญหาที่มีเครื่องมือมากมายและสามารถทำได้อย่างมีประสิทธิภาพ อย่างไรก็ตามการสร้างตัวจำแนกในกรณีที่มีประเภทที่ต้องการจำแนกมากกว่าสองประเภทยังเป็นสิ่งที่ยากจะหาคำตอบว่าปัจจัยใดเป็นตัวกำหนดความยากหรือง่ายในการฝึกสอนตัวจำแนก แนวทางหนึ่งในการสร้างตัวจำแนกคือการสร้างตัวจำแนกแบบหลายประเภทจากตัวจำแนกแบบทวิภาค ปัญหาที่ผู้วิจัยสนใจคือการพิสูจน์ประสิทธิภาพของตัวจำแนกที่สร้างได้ โดยพิจารณาตัวจำแนกแบบทวิภาคเป็นกล่องดำ หรือที่เรียกว่าการลดรูปการเรียนรู้ (learning reductions)

2. **ปัญหาการทำนายแบบออนไลน์บนกราฟ** ซึ่งเป็นปัญหาที่เกี่ยวข้องกับทั้งสามกลุ่มปัญหาที่เสนอในหัวข้อวิจัย ปัญหาดังกล่าวศึกษาการทำนายป้ายชื่อของจุดยอดในกราฟแบบออนไลน์ ในปัญหาดังกล่าวจะมีกระบวนการในการเรียนรู้และการทำนายที่เกิดขึ้นพร้อม ๆ กัน และต้องการจะทำนายให้มีจำนวนครั้งที่ผิดพลาดน้อยที่สุด

2. วัตถุประสงค์ของโครงการ

โครงงานวิจัยนี้มีเป้าหมายเพื่อศึกษาคุณสมบัติเชิงทฤษฎีของอัลกอริทึมในแง่ของคุณภาพของผลลัพธ์และออกแบบอัลกอริทึมที่ดีขึ้น โดยมุ่งเน้นที่ปัญหาสามกลุ่มสำคัญคือ กลุ่มปัญหาการหาคำตอบที่ดีที่สุด, กลุ่มปัญหาออนไลน์, และกลุ่มปัญหาการเรียนรู้ด้วยเครื่องจักร

3. การดำเนินงานและผลการวิจัยอย่างย่อ

ในส่วนนี้ จะกล่าวถึงภาพรวมของผลการวิจัย สำหรับผลการวิจัยอย่างเป็นทางการแสดงในส่วนภาคผนวก

3.1 การวิเคราะห์อัตราความผิดหวัง และตัวอย่างโต้แย้ง

ในช่วงปีที่ 1 ได้พยายามขยายขอบเขตของผลงานวิจัยเดิมที่ได้รับการสนับสนุนจากทุนวิจัยเมื่อปี พ.ศ. 2547 ที่ได้ศึกษาการสร้างตัวจำแนกแบบหลายประเภทจากตัวจำแนกแบบทวิภาค โดยศึกษาความเปลี่ยนแปลงของอัตราความผิดพลาด (error) ของตัวจำแนกที่สร้างขึ้น เทียบกับตัวจำแนกเชิงทวิภาคเดิม

ผู้วิจัยพยายามขยายขอบเขตที่ได้ให้ครอบคลุมถึงคุณสมบัติที่สำคัญอีกคุณสมบัติหนึ่งของตัวจำแนก คืออัตราความผิดหวัง (regret) ซึ่งนิยามเป็นผลต่างระหว่างอัตราความผิดพลาดของตัวจำแนก กับอัตราความผิดพลาดของตัวจำแนกที่ดีที่สุด

อย่างไรก็ตาม เมื่อได้พยายามปรับปรุงบทพิสูจน์ให้สามารถใช้ได้ในกรณีทั่วไป กลับพบตัวอย่างโต้แย้ง (counter example) ที่แสดงว่าภายใต้นิยามของความผิดหวังที่ใช้ อัลกอริทึมดังกล่าวไม่สามารถแปลงอัลกอริทึมแบบทวิภาคให้เป็นอัลกอริทึมแบบหลายกลุ่มในขณะที่ยังรักษาอัตราความผิดหวังได้เท่าที่คาดไว้

ตัวอย่างดังกล่าวยังแสดงให้เห็นถึงข้อผิดพลาดในบทพิสูจน์ประสิทธิภาพเดิมที่แสดงว่า ADAG สามารถลดรูปได้โดยรักษาอัตราความผิดพลาด (error) ในส่วนนี้ผู้วิจัยได้เขียน Erratum และส่งไปตีพิมพ์ในการประชุมวิชาการที่เคยได้นำเสนอผลงานเดิมไปแล้ว

3.2 ปัญหาการทำนายแบบออนไลน์บนกราฟ

ในช่วงปีที่ 2 ผู้วิจัยได้ศึกษาปัญหาการทำนายบนกราฟแบบออนไลน์ ปัญหานี้จะพิจารณาเกมบนกราฟที่มีการแปะป้ายชื่อบนจุดยอดระหว่างผู้ทำนายและผู้ถาม เมื่อเริ่มต้น ผู้ทำนายจะได้รับกราฟที่ซ่อนป้ายชื่อของทุก ๆ โหนดไว้ จากนั้นเกมจะดำเนินไปเป็นรอบ ๆ ผู้ถามจะถามป้ายชื่อของจุดยอดบนกราฟ ผู้ทำนายจะต้องทำนายป้ายชื่อดังกล่าว หลังจากที่ได้ทำนายแล้ว ผู้ถามจะเฉลยคำตอบให้กับผู้ทำนาย เป้าหมายของเกมนี้คือต้องการจะทำนายให้มีความผิดพลาดน้อยที่สุด

ปัญหาออนไลน์ดังกล่าวสามารถประยุกต์ใช้ได้กับงานหลายประเภท เช่น การจำแนกจดหมายขยะ การจำแนกแบบหลายกลุ่ม เป็นต้น

ผู้วิจัยได้นำเสนอแนวทางแก้ปัญหาใหม่ โดยทั้งอัลกอริทึมและวิธีการพิสูจน์เป็นแบบ combinatorial ซึ่งแตกต่างจากแนวทางเดิมที่ใช้กันอยู่ ที่เน้นการแปลงกราฟซึ่งเป็นวัตถุเชิง combinatorial ให้อยู่ในรูปที่จัดการได้ด้วยคณิตวิเคราะห์และพีชคณิตเชิงเส้น นอกจากนี้ อัลกอริทึมยังสามารถทำงานได้ในกรณีที่ปัญหาการทำนายไม่ใช่ปัญหาเชิงทวิภาคด้วย

อย่างไรก็ตาม ระหว่างรอการตีพิมพ์ Herbster, Lever, และ Pontil ก็ได้นำเสนอวิธีการคล้าย ๆ กัน แต่ใช้การพิสูจน์ที่ค่อนข้างซับซ้อน (ดูการอ้างอิงได้จาก manuscript ที่แนบในภาคผนวก) และให้ขอบจำกัดที่ดีกว่า อย่างไรก็ตาม การพิสูจน์ของ Herbster และคณะไม่สามารถแสดงได้อย่างชัดเจนว่าวิธีการดังกล่าวสามารถใช้ได้กับการทำนายที่ไม่ใช่ปัญหาเชิงทวิภาค

Output ที่ได้จากโครงการวิจัยที่ได้รับทุนจาก สกอ. และ สกว.

1. ผลงานตีพิมพ์ในวารสารวิชาการนานาชาติ

ผลงานในส่วนที่สองได้รับยอมรับให้ตีพิมพ์ในวารสาร Information Processing Letters โดยสำนักพิมพ์ Elsevier วารสารดังกล่าวมี Impact Factor 0.706 (ระบุที่หน้าเว็บของวารสาร)

ขณะนี้ยังไม่ทราบฉบับที่ลงพิมพ์ อย่างไรก็ตามบทความดังกล่าวได้รับการเผยแพร่แบบออนไลน์แล้วในหน้าเว็บของวารสารดังกล่าว

บทความออนไลน์: http://dx.doi.org/10.1016/j.ipl.2010.02.008

2. การนำผลงานวิจัยไปใช้ประโยชน์

N/A

3. อื่นๆ

N/A

ภาคผนวก

Manuscripts

1. J. Fakcharoenphol, B. Kijsirikul, Short proofs for online multiclass prediction on graphs, *Information Processing Letters* (2010), doi:10.1016/j.ipl.2010.02.008

Impact factor: 0.706

2. J. Fakcharoenphol, B. Kijsirikul, *Erratum: Constructing Multiclass Learners from Binary Learners: A Simple Black-Box Analysis of the Generalization Errors*. In *Proceedings of the 19th International Conference on Algorithmic Learning Theory*. Springer. 2008

Information Processing Letters ••• (••••) •••–•••

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Short proofs for online multiclass prediction on graphs

Jittat Fakcharoenphol [a,*,1], Boonserm Kijsirikul [b]

[a] Department of Computer Engineering, Kasetsart University, Bangkok 10900, Thailand
[b] Department of Computer Engineering, Chulalongkorn University, Bangkok 10330, Thailand

ARTICLE INFO

Article history:
Received 24 January 2009
Received in revised form 8 February 2010
Accepted 16 February 2010
Available online xxxx
Communicated by B. Doerr

Keywords:
Analysis of algorithms
On-line algorithms
On-line learning
Prediction on graphs

ABSTRACT

We present short proofs on the mistake bounds of the 1-nearest neighbor algorithm on an online prediction problem of path labels. The algorithm is one of key ingredients in the algorithm by Herbster, Lever, and Pontil for general graphs. Our proofs are combinatorial and naturally show that the algorithm works when the set of labels is not binary.

1. Introduction

We consider the online prediction of graph labels which can be described briefly as follows. There is a graph $G = (V, E)$ with a fixed node labeling $\mathbf{u} : V \to L$ for some label set L; initially, the algorithm does not have any information of $\mathbf{u}$. The learning process proceeds in rounds. For each round t, Nature asks for a label of node $q_t \in V$. The algorithm predicts the label $\hat{u}(q_t)$ and later receives the true label $\mathbf{u}(q_t)$. It makes a mistake when $\hat{u}(q_t) \neq \mathbf{u}(q_t)$. The goal is to minimize the number of times the algorithm makes mistakes. For motivation and applications of the problem, see, e.g., [1,6].

The performance of the algorithm is measured against the number of cut edges on the partition of graph nodes induced by the true labeling. We denote by $\Phi_G(\mathbf{u})$ the number of edges in G whose labels on both ends are different.

The recent result of Herbster, Lever, and Pontil [6] gives an efficient algorithm with a mistake bound of

$$2\Phi_G(\mathbf{u}) \max\left[0, \log_2\left(\frac{n-1}{2\Phi_G(\mathbf{u})}\right)\right] + \frac{2\Phi_G(\mathbf{u})}{\ln 2} + 1.$$

They first embed G into a path graph S, called a *spine of G*, then they use the 1-nearest neighbor (1-NN) algorithm for label prediction. The first step only incurs a factor of 2 on the cut size ($\Phi_G(\mathbf{u})$); their mistake bound follows from the proof of the second step.

Herbster et al.'s proof of the mistake bound of the 1-NN algorithm is based on the result on the Halving algorithm [9]. To do so, they define a probability distribution over the possible hypotheses so that the Halving algorithm implements the 1-NN algorithm.

In this manuscript we give a short combinatorial proof that the 1-NN algorithm on an n-node path with k cut edges makes at most $O(k \log(n/k) + k)$ mistakes. This bound is off by a constant factor from the bound in [6]. We also show that with a more careful analysis, this bound can be improved to almost match the bound of [6].

Apart of being very short and combinatorial, another nice property of our proof is that the bound does not depend on the set of labels. Therefore, they also imply that the algorithm of Herbster et al. also works when labels are not binary.

We present our proofs in Section 2. The next section reviews other closely related work.

* Corresponding author.
E-mail addresses: jittat@gmail.com (J. Fakcharoenphol), Boonserm.K@chula.ac.th (B. Kijsirikul).

[1] Supported by the Thailand Research Fund Grant MRG5080318.

0020-0190/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.ipl.2010.02.008

Please cite this article in press as: J. Fakcharoenphol, B. Kijsirikul, Short proofs for online multiclass prediction on graphs, Information Processing Letters (2010), doi:10.1016/j.ipl.2010.02.008

1.1. Other related work

Early works [8,7] on graph prediction use algorithms based on the Perceptron algorithm using pseudoinverse of graph Laplacian as a kernel and provide mistake bounds that depend on the cut size and the largest effective resistance between any pair of vertices in the graph. Herbster [4] exploits the cluster structure of the labeling on the graph, and provides an improved mistake bounds. However, there is an example by [6] that shows that the algorithm based on this approach may make $\Theta(\sqrt{n})$ on some n-node graph. Recently, Herbster and Lever [5] explore another class of seminorms, called Laplacian p-seminorms, and show that with the right setting of p (which depends only on the graph) the mistake bound is logarithmic.

Recent work of Cesa-Bianchi, Gentile, and Vitale [1] presents an algorithm for prediction on trees whose worst-case number of mistakes over all labeling and all query sequence is optimal up to a constant factor.

We also note that our work in this paper stems from the proof of a slightly weaker bound appeared in [3] based on result in [2].

2. The 1-NN algorithm on paths

We are given a line graph $G = (V, E)$; let $n = |V|$. Without loss of generality, we label nodes in G as $1, 2, \ldots, n$, where nodes 1 and n are the only two degree-1 nodes, and there is an edge $(i, i+1) \in E$, for all $1 \leqslant i < n$.

The online prediction problem proceeds in rounds. Each round t, when Nature asks for a label of node q_t, the 1-NN algorithm finds the closest node s whose true label is known and returns s's label. For that round, we call q_t the query node and s the source node. Later on, when the true label of i is revealed, the algorithm updates i's label on the graph. If the predicted label of q_t is not the same as the revealed label, the algorithm makes a mistake.

In our analysis, a *distance* from a given node i to another node j is the number of edges on the unique path from i to j, i.e., it is $|i - j|$. A *distance from i to edge $(j, j+1)$* is the minimum of the distances from i to j or from i to $j+1$, i.e., it is $\min(|i - j|, |i - j - 1|)$.

We first present a simpler theorem that shows the same asymptotic bound but with a higher constant.

Theorem 1. *The 1-NN algorithm makes at most $O(k + k \log(\frac{n}{k}))$ mistakes where $n = |V|$ and k is the number cut edges.*

Proof. First assume that $k < n - 1$, otherwise the bound holds trivially.

Denote all cut edges by $e_1, e_2, \ldots, e_k$; we assume that they are ordered by the smaller indices of their end points. These cut edges partition G into $k + 1$ connected subgraphs. Call them $C_0, C_1, \ldots, C_k$; note that each edge e_i is adjacent to C_{i-1} and C_i.

We start our analysis after the first mistake is made.

For each mistake the algorithm makes after that, we have that the true labels of query node i and source node s are different. Therefore, there exists some cut edge along the unique path P from i to s. We charge this mistake to the closest cut edge on P from i.

To see that each edge e_i is charged by at most $1 + \log|C_i|$ times by nodes in C_i, consider the sequence of nodes that charge to e_i: $v_1, v_2, \ldots$. For $j > 1$, in order for v_j to make a mistake, e_i must be closer to v_j than all other known nodes, including v_{j-1}. Thus, we have that the distance from a node in C_i charging to e_i decreases by at least a factor of 2 each time e_i is charged. Thus, e_i can be charged at most $1 + \log|C_i|$ times by nodes in C_i. We can use the same argument to show that e_i is charged by at most $1 + \log|C_{i-1}|$ times from nodes in C_{i-1}.

Note that only mistakes on nodes in C_{i-1} or C_i can be charged to e_i. Therefore e_i is charged by at most $2 + \log|C_{i-1}| + \log|C_i|$ times.

Summing over all cut edges, the number of mistakes charged to any cut edge is at most $2k + \sum_{i=0}^{k} 2\log|C_i|$. Since $\sum_{i=0}^{k} |C_i| = n$, the number of mistakes maximized when every subgraph is of the same size, i.e., the number of mistakes is at most $2k + (k+1)(2\log(n/(k+1)))$.

Accounting for the first mistake, we have that the number of mistake is at most $1 + 2k + (k+1)(2\log(n/(k+1))) = O(k\log(n/k) + k)$ as claimed. □

The next theorem shows a tighter bound. To prove it, we need more notations.

First denote the end points of each edge e_i, for $1 \leqslant i \leqslant k$, by p_i and $p_i + 1$. For simplicity, we set $p_0 = 0$. Note that nodes in C_i are $p_{i-1} + 1, p_{i-1} + 2, \ldots, p_i$. We also refers to a set of contiguous nodes as an *interval* of nodes.

We call any node on which the algorithm makes a mistake a *blue* node; note that the number of blue nodes at any time equals the number of mistakes the algorithm makes so far.

As in the proof of Theorem 1, we shall trace the execution of the algorithm.

Theorem 2. *The 1-NN algorithm makes at most $2k + k\log(n/k) + 1$ mistakes where $n = |V|$ and k is the number cut edges.*

Proof. We use a slightly different charging scheme. For each component C_i, we charge the first mistake from nodes in C_i to the component itself. For other mistakes, we use the same charging scheme, i.e., we charge them to the first cut edges encountered on the paths to the source nodes.

For each i, $1 \leqslant i \leqslant k$, we will define an interval of nodes W_i that contains all node charging to e_i such that the sets $W_1, W_2, \ldots, W_k$ are pair-wise disjoint.

The total number of mistakes charged to the components is at most $k + 1$. Using the same argument on the maximum of the sum of logarithms as in the end of the proof of Theorem 1, to prove the theorem, it suffices to show that the number of times each cut edge e_i is charged is $1 + \log|W_i|$.

Consider edge e_i that has at least one mistake charged to it.

Let v be the first blue node that charges to e_i. Note that v is either from C_{i-1} or C_i; let C' be one of these subgraphs that contains v, and let C'' be another subgraph.

We assume that v is the first blue node in C'; thus the mistake on v can be charged to C'. We shall deal with the case when v is not the first one later.

Please cite this article in press as: J. Fakcharoenphol, B. Kijsirikul, Short proofs for online multiclass prediction on graphs, Information Processing Letters (2010), doi:10.1016/j.ipl.2010.02.008

There are two cases. The first case is when every node charging to e_i is from C'. Define W_i to be the minimal interval containing one of the endpoint of e_i in C' and v. The proof from Theorem 1 shows that e_i is charged at most $1 + \log|W_i|$ times.

Now consider the second case. Let u be the first node in C'' that charges to e_i. We also assume that u is also the first blue node C''. Recall that u's mistake can be charged to C''. We define W_i to be the minimal interval containing u and v.

At any step t in the execution of the algorithm, let D_i^t be a set of nodes that can possibly charge to e_i *after* step t. The argument as in Theorem 1 implies that after u charges to e_i, in each step t that some node charges to e_i, the size of D_i^t decreases at least by a factor of two, i.e., $|D_i^t| \leqslant |D_i^{t-1}|/2$.

Now consider each step t before u charges to e_i. Observe that every candidate discarded in this step must be in W_i, i.e., $D_i^{t-1} - D_i^t \subseteq W_i$; thus, in those steps, we also have that

$$\left|D_i^t \cap W_i\right| \leqslant \left|D_i^{t-1} \cap W_i\right|/2.$$

Since u does not charges to e_i, we have that e_i is charged by at most $1 + \log|W_i|$ times.

We are left with the case that v or u (or both) are not the first blue nodes on C' or C''. We only consider the case where there exists some mistake in C'' that charges to e_i. Similar argument can be used when all mistakes charged to e_i are from nodes in C' but u is not the first blue node in C'.

Let w' and w'' be the first blue nodes in C' and C''. If $v \neq w'$, we let v' be the node adjacent to w' which is closer to e_i; otherwise we let $v' = v$. We define u' similarly, i.e., u' is the node adjacent to w'' closer to e_i or u itself if $u' = w''$. We define W_i to be the minimal interval containing v' and u'.

We are done if we can show that for each $x \in \{u, v\}$ such that x is not the first node, when the algorithm makes mistakes on x, the set $|D_i^t \cap W_i|$ also shrinks by a factor of two. We look at the case where $x = u$, the other case is similar. To see that this claim is true, note that one of the candidate sources of u is w'', but u chooses another node which is as far as the furthest node on the direction to e_i in $D_i^t \cap W_i$. □

Using Theorem 2 with the algorithm of Herbster et al., we obtain the mistake bound of

$$2\big(\Phi_G(\mathbf{u})\big)\max\left[0, \log_2\left(\frac{n}{2\Phi_G(\mathbf{u})}\right)\right] + 4\Phi_G(\mathbf{u}) + 1,$$

which is comparable to the original bound except on the constant of the second term. (We have 4, [6] has $\frac{2}{\ln 2} \approx 2.88$.)

Acknowledgements

We would like to thank anonymous referees who gave us useful feedback and suggested the idea for the proof of Theorem 2. We also thank Parinya Chalermsook and Mark Herbster for useful discussions.

This work is supported by the Thailand Research Fund Grant MRG5080318.

References

[1] Nicolò Cesa-Bianchi, Claudio Gentile, Fabio Vitale, Fast and optimal prediction on a labeled tree, in: Proceedings of the 22nd Annual Conference on Learning Theory (COLT'09), 2009.

[2] Parinya Chalermsook, Jittat Fakcharoenphol, Simple distributed algorithms for approximating minimum steiner trees, in: Proceedings of the 11th Annual International Conference on Computing and Combinatorics (COCOON'05), 2005, pp. 380–389.

[3] Jittat Fakcharoenphol, Boonserm Kijsirikul, Low congestion online routing and an improved mistake bound for online prediction of graph labeling, CoRR, abs/0809.2075, 2008.

[4] Mark Herbster, Exploiting cluster-structure to predict the labeling of a graph, in: Proceedings of the 19th International Conference on Algorithmic Learning Theory (ALT'08), 2008, pp. 54–69.

[5] Mark Herbster, Guy Lever, Predicting the labelling of a graph via minimum p-seminorm interpolation, in: Proceedings of the 22nd Annual Conference on Learning Theory (COLT'09), 2009.

[6] Mark Herbster, Guy Lever, Massimiliano Pontil, On-line prediction on large diameter graphs, in: Proceedings of the 22th Annual Conference on Neural Information Processing Systems (NIPS'08), 2008.

[7] Mark Herbster, Massimiliano Pontil, Prediction on a graph with a perceptron, in: B. Schölkopf, J. Platt, T. Hoffman (Eds.), Advances in Neural Information Processing Systems, vol. 19, MIT Press, Cambridge, MA, 2006, pp. 577–584.

[8] Mark Herbster, Massimiliano Pontil, Lisa Wainer, Online learning over graphs, in: Proceedings of the 22nd International Conference on Machine Learning (ICML'05), ACM, New York, NY, USA, 2005, pp. 305–312.

[9] J.M. Barzdin, R.V. Frievald, On the prediction of general recursive functions, Soviet Math. Dokl. 13 (1972) 1224–1228.

Please cite this article in press as: J. Fakcharoenphol, B. Kijsirikul, Short proofs for online multiclass prediction on graphs, Information Processing Letters (2010), doi:10.1016/j.ipl.2010.02.008

Erratum: Constructing Multiclass Learners from Binary Learners: A Simple Black-Box Analysis of the Generalization Errors

Jittat Fakcharoenphol[1] and Boonserm Kijsirikul[2]

[1] Department of Computer Engineering, Kasetsart University, Bangkok, Thailand
jtf@ku.ac.th
[2] Department of Computer Engineering, Chulalongkorn University, Bangkok, Thailand
Boonserm.K@chula.ac.th

Abstract. There are errors in our paper "Constructing Multiclass Learners from Binary Learners: A Simple Black-Box Analysis of the Generalization Errors," which appeared in ALT'05 [3]. The errors are related to our uses of union bounds. We briefly describe the problem and discuss which of our results can be shown to hold. We also provide a counter example for our previous claim in the full version of the erratum.

1 Background

In [3], we analyzed various multiclass learning algorithms that use binary classifications as subroutines. We viewed binary classifiers as black-boxes and analyzed the error rate of the multiclass construction as a function of binary error rates. This approach is mainly known as learning reductions [1].

In what follows, we assume that the readers are familiar with the pair-wise reductions such as Decision Directed Acyclic Graphs [5] (DDAG), Adaptive Directed Acyclic Graphs [4] (ADAG).

2 The Errors

The problems in that paper is in our analysis of adaptive constructions, i.e., those whose set of invoked binary classifiers changes over the input. They include DDAG, ADAG, and Randomized Decision Directed Acyclic Graphs (R-DDAG). More specifically, the errors are regarding our use of the union bound.

To see the problem, we consider our analysis of ADAG for the problem with k classes. We start with the setting. Let D be the distribution over X, set of all data points. There are $\binom{k}{2}$ classifiers: there is a binary classifier $A_{i,j}$ for each pair $i \neq j$. Error rate of classifier $A_{i,j}$, $\epsilon_{i,j}$ is defined to be

$$\Pr_{x \sim D}[A_{i,j}(x) \text{ gives a wrong prediction} | x \text{ belongs to class } i \text{ or class } j].$$

Y. Freund et al. (Eds.): ALT 2008, LNAI 5254, pp. 464–466, 2008.

In the ADAG reduction, we have a full binary tree T, with k leaves initially labelled with all classes, while all internal nodes are unlabelled. Call the leaf labelled with i, L_i. Given a data point x, the prediction algorithm picks any unlabelled node u whose children are all labelled. Suppose that they are labelled with i and j. We then call $A_{i,j}(x)$ and assign the result of the classifier as the label of u. The algorithm iterates until there is no unlabelled node left. The multiclass prediction is the label of the root node.

Our analysis first assume that the data point belongs to class i. Let $\mathcal{I}(i)$ denote the set of internal nodes of T on the path from L_i to the root. The algorithm makes a wrong prediction if any classifier called on these nodes make mistakes. Denote by $\mathcal{E}(u)$ the event that the classifier on node u makes a wrong prediction; thus, the multiclass error rate is $\Pr\left[\bigcup_{u\in\mathcal{I}(i)} \mathcal{E}(u)\right] \leq \sum_{u\in\mathcal{I}(i)} \Pr[\mathcal{E}(u)]$.

For any node u, let $\mathcal{L}(i)$ denote the set of leaf labels in the subtree rooted at u. We claim, erroneously, that $\max_{j\in\mathcal{L}(u)} \epsilon_{i,j}$ is an upperbound on $\Pr[\mathcal{E}(u)]$.

If this were true, we would have that, since $|I(i)| \leq \lceil \log k \rceil$, the error rate is at most $\sum_{j=1}^{\lceil \log k \rceil} \epsilon_{i,r_j}$, when r_j is the class c with the j-th largest error rate $\epsilon_{i,c}$.

The above claim would have work if each classifiers $A_{i,j}$ is randomized and for any data point x, it makes a mistake with probability $\epsilon_{i,j}$. However, usually for a fixed x, the error is not random.

The correct analysis of $\Pr[\mathcal{E}(u)]$ must consider all binary classifiers $A_{i,j}$ for $j \in \mathcal{L}(u)$. Let event $\mathcal{F}(i,j)$ denote the event that the classifier $A_{i,j}$ is used at node u and makes a wrong prediction. Thus, $\Pr[\mathcal{E}(u)] = \Pr[\bigcup_{j\in\mathcal{L}(u)} \mathcal{F}(i,j)] = \sum_{j\in\mathcal{L}(u)} \Pr[\mathcal{F}(i,j)]$. With no further assumption, we can only bound this with $\sum_{j\in\mathcal{L}(u)} \epsilon_{i,j}$, using the union bound. Thus, the probability of making mistake is at most $\sum_{j\neq i} \epsilon_{i,j}$, using again the union bound. In the case of uniform error rate, this only gives the bound of $(k-1)\epsilon$. This analysis is tight (see an example in [2]).

The erroneous theorems are Theorems 2, 3, and 4. In Theorem 2, we claim an upper bound δ_i for input from class i to be at most $\max\{\sum_{j<i} \epsilon_{i,j}, \sum_{j>i} \epsilon_{i,j}\}$. The correct upperbound is $\sum_{j\neq i} \epsilon_{i,j}$. This is the correct bounds for Theorems 3 and 4 as well.

Our analyses of non-adaptive constructions (Theorems 1, 5, and 6) remain correct.

3 A Tight Example

In the full version of the erratum [2], we describe the probability space of the input with k classes and a set of binary classifiers such that the binary error rate is $1/(k-1)$ while multiclass error rate of the constructions is 1, for $k > 2$.

Acknowledgment. We would like to thank John Langford for a useful discussion.

References

[1] Beygelzimer, A., Dani, V., Hayes, T., Langford, J., Zadrozny, B.: Error limiting reductions between classification tasks. In: ICML 2005, pp. 49–56 (2005)

[2] Fakcharoenphol, J., Kijsirikul, B.: Erratum: Constructing multiclass learners from binary learners: A simple black-box analysis of the generalization errors, http://www.cpe.ku.ac.th/~jtf/papers/blackbox-erratum.pdf

[3] Fakcharoenphol, J., Kijsirikul, B.: Constructing multiclass learners from binary learners: A simple black-box analysis of the generalization errors. In: Jain, S., Simon, H.U., Tomita, E. (eds.) ALT 2005. LNCS (LNAI), vol. 3734, pp. 135–147. Springer, Heidelberg (2005)

[4] Kijsirikul, B., Ussivakul, N., Meknavin, S.: Adaptive directed acyclic graphs for multiclass classification. In: Ishizuka, M., Sattar, A. (eds.) PRICAI 2002. LNCS (LNAI), vol. 2417, pp. 158–168. Springer, Heidelberg (2002)

[5] Platt, J., Cristianini, N., Shawe-Taylor, J.: Large margin DAGs for multiclass classification. In: Advance in Neural Information Processing System, vol. 12. MIT Press, Cambridge (2000)