**Weekly report**: FAST : Forecast and Simulated Trust

ผลงานช่วง วันศุกร์ ที่ 14 สิงหาคม 2558 ถึง วันอังคารที่ 18 สิงหาคม 2558

รายงาน วันอังคารที่ 18 สิงหาคม 2558

โดย

นางสาวนภวรรณ ดุษฎีเวทกุล เลขทะเบียนนักศึกษา 5509611637

นายวงศธร ทองถาวร เลขทะเบียนนักศึกษา 5509680061

1. **สิ่งที่ได้รับมอบหมายในช่วงที่แล้วและทำสำเร็จ**

* ศึกษา Improving the Performance of Mobile Phone Crowdsourcing Applications
* ศึกษา Brief survey of crowdsourcing for data mining
* ศึกษา SACRM: Social Aware Crowdsourcing with Reputation Management in mobile sensing

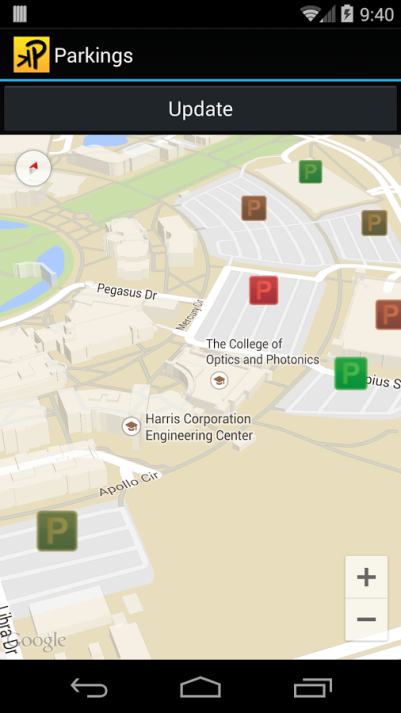
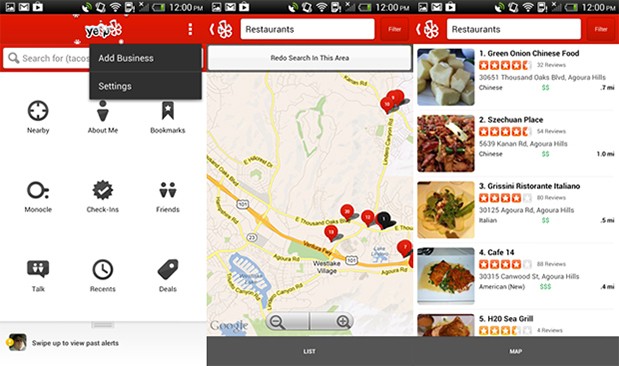
1. **สิ่งที่ได้รับมอบหมายในช่วงที่แล้วและยังไม่ได้ทำ หรือทำแล้วติดปัญหา**

* ในบทความ Improving the Performance of Mobile Phone Crowdsourcing Applications ติดปัญหาเรื่อง “Maximum likelihood and Bayesian Estimation
* ในบทความImproving the Performance of Mobile Phone Crowdsourcing Applications ติดปัญหาในข้อ 4.2 ตอนทำ case ตัวอย่างว่าทำไมค่าความน่าเชื่อถือของทุกคนถึงลดลง
* ศึกษา SACRM: Social Aware Crowdsourcing with Reputation Management in mobile sensing
* ศึกษา Brief survey of crowdsourcing for data mining

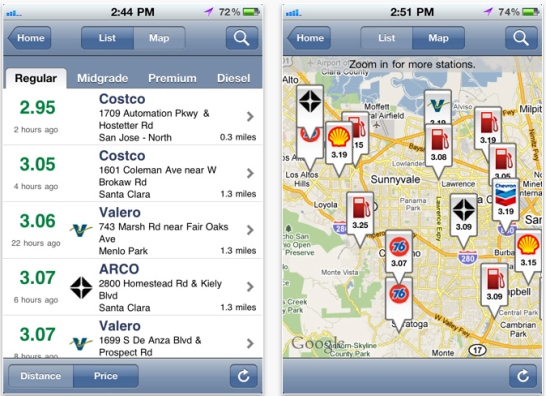
1. **รายละเอียดเพิ่มเติม**

* สรุปบทความ Improving the Performance of Mobile Phone Crowdsourcing Applications

ในบทความนี้ ทางผู้วิจัยได้ศึกษาวิธีที่เพิ่มประสิทธิภาพของระบบ Crowdsourcing ในโทรศัพท์มือถือ โดยใช้ Kpark application ซึ่งเป็น Software บนโทรศัพท์มือถือที่ทำให้ผู้ใช้หาที่จอดรถที่ว่างในมหาวิทยาลัย Central of Florida โดยให้ผู้ใช้หลายๆคนส่งข้อมูลมาว่าสถานที่จอดรถตรงบริเวณนั้นว่างหรือไม่ และใช้เทคนิควิธี Multiple trust based data fusion techniques ในการประมวลผลข้อมูล สาเหตุที่ต้องสร้างแอปพลิเคชันนี้เนื่องจากที่จอดรถในมหาวิทยาลัยไม่เพียงพอต่อจำนวนรถยนต์ถึงแม้ว่ามหาวิทยาลัยจะเพิ่งสร้างที่จอดรถมาเพิ่มแล้วก็ตาม ข้อดีของการใช้เทคนิค Multiple trust based data fusion คือ ไม่ต้องเสียเงินค่าอุปกรณ์จำพวกเซนเซอร์ต่างๆ ถ้ามีผู้ใช้หลายๆคนส่งข้อมูลเข้ามา ระบบจะมีความแม่นยำมากขึ้น และได้ข้อมูลที่เป็นปัจจุบันทันที ส่วนระบบจัดการแอปพลิเคชัน Kpark จะมีการจัดการผู้ใช้โดยใช้ค่า Trustworthiness จาการนำข้อมูลทำนายด้วย Multi trust prediction และใช้ระบบ Real time ในการสร้างแผนผังที่จอดรถ ซึ่ง Multi trust prediction มีพื้นฐานมากจากระบบ Majority voting

ปัจจุบันมีแอปพลิเคชันที่ใช้ระบบ Crowdsourcing ในการประมวลผลหลายแอฟพลิเคชัน เช่น Waze(ทำนายสภาพถนน) GasBuddy(หาจุดที่น้ำมันราคาถูกที่สุด) และYelp(ให้คะแนนร้านอาหาร) ข้อดีของระบบ Crowdsourcing คือไม่รบกวน Privacy ของผู้ใช้ เช่น GPS เป็นต้น  
 

รูปภาพที่ 1 :Kpark รูปภาพที่ 2 :Yelp

รูปภาพที่ 3 : Waze รูปภาพที่ 4 : GasBuddy

แต่ข้อเสียของระบบ Crowdsourcing คือถ้าได้ Poor data(ข้อมูลเท็จ) เข้ามาจำนวนมากๆ จะทำให้ระบบมีปัญหาเนื่องจากระบบใช้การประมวลผลแบบ Majority voting ซึ่งเป็นวิธีการประมวลผลแบบเชื่อว่าถ้ามูลที่มีผู้ใช้ส่งมาให้ระบบมากที่สุดเป็นข้อมูลจริงเสมอ จากปัญหาดังกล่าวทำให้ระบบจำเป็นต้องมีค่าความน่าเชื่อถือ(Trust prediction)ในการประเมินว่าผู้ใช้ที่ส่งข้อมูลเข้ามาเป็นข้อมูลที่มีความน่าเชื่อถือมากน้อยเพียงใด

ระบบ Kpark แบ่งออกเป็น 3 ส่วนด้วยกันคือ Trust prediction algorithm(อัลกอริทึมในการทำนายความน่าเชื่อถือของผู้ใช้) อัลกอริทึมในการทำนายที่จอดรถที่ว่าง และการประมวลผลแบบ Real time

ในส่วนของ Trust prediction ที่มีใช้ในปัจจุบันมีเงื่อนไขคือถ้าในเวลาใดเวลาหนึ่งมีข้อมูลมากกว่า 1 ชุดส่งมาจากผู้ใช้คนเดียวกันจะถือว่าข้อมูลชุดนั้นมีแนวโน้มผิดพลาด(data error)

1. Beta Reputation Method เป็นการหาคะแนน2 ค่า คือคะแนนความพอใจและคะแนนความไม่พอใจ โดยจะวัดจากคำตอบโดยรวมที่ผู้ใช้ส่งมาในระบบ จากนั้นจะนำคะแนนไปแทนในสูตรทำให้ได้ค่าความน่าเชื่อถือของแต่ละบุคคลออกมา
2. Gompertz Method เป็นการให้ค่าความน่าเชื่อถือของแต่ละบุคคล โดยใช้ Gompertz Function ถ้าหากบุคคลนั้นให้ข้อมูลเท็จค่าความน่าเชื่อถือจะลดทีละมาก แต่ถ้าหากให้ข้อมูลจริงค่าความน่าเชื่อถือจะเพิ่มทีละน้อยๆ และควรเชื่อใจคนที่มีค่าความน่าเชื่อถือมากมากกว่าคนที่มีค่าความน่าเชื่อถือน้อย
3. Robust Averaging Method เป็นการหาความน่าเชื่อถือของแต่ละบุคคลโดยนำข้อมูลทั้งหมดที่มีทำให้เป็นหน่วยเดียวกัน (Normalization data)
4. Maximum Likelihood and Bayesian Estimation -->ติดเรื่องนี้ค่ะ^^

ต่อมาในเรื่องการทำนายตำแหน่งที่จอดรถที่ว่าง(Occupancy Calculation) ที่มีใช้ในปัจจุบันจะมี 2 อัลกอริทึมคือ

1. Weight average trust เป็นการเชื่อข้อมูลจากทุกๆคน โดยบุคคลที่มีความน่าเชื่อถือมากข้อมูลที่เขาส่งมาจะมีความน่าเชื่อถือสูงกว่าบุคคลที่มีความน่าเชื่อถือน้อย

2. Max trust เป็นการเชื่อข้อมูลจากคนที่มีคะแนนความน่าเชื่อถือมากที่สุดเพียงคนเดียวเท่านั้น

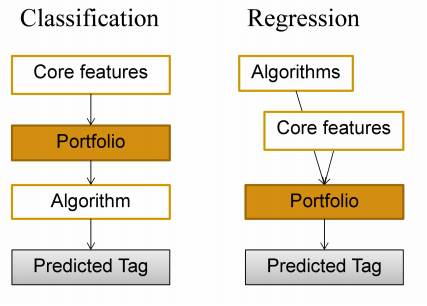
ส่วนต่อมาคือ Data freshness เป็นการประมวลผลแบบ Real time จะเป็นการจัดการข้อมูลให้มีความสดใหม่อยู่เสมอเพื่อป้องกันปัญหาเรื่องข้อมูลไม่อัพเดท เช่น หากมีผู้ใช้แจ้งสถานการณ์จอดรถมาช่วงเวลา 09.00-10.00 ว่าที่จอดรถ เต็มแต่พอเวลาหลังจากนั้นไม่มีผู้ใช้ใดแจ้งแล้วในฐานข้อมูลจะจดจำว่าสถานที่จอดรถทุกที่เต็มหมด แต่ในความเป็นจริงอาจจะมีบางคนขับรถออกจากที่จอดรถแล้วก็เป็นได้ ซึ่งปัญหานี้จำเป็นต้องทำให้ข้อมูลมีความสดใหม่อยู่เสมอ

ในส่วนอัลกอริทึมที่พัฒนาเพิ่มเติมในระบบ Kpark มี 2 อัลกอริทึม เนื่องจากอัลกอริทึมเดิมที่มีไม่สามารถทำให้ระบบแก้ปัญหาได้ดีที่สุดในทุกสถานการณ์ได้ จึงเกิดอัลกอริทึม

1. Algorithm Portfolio เป็นอัลกอริทึมที่จะเก็บอัลกอริทึมทั้งหมดที่มีอยู่ในปัจจุบัน ได้แก่ Beta Reputation Method ,Gompertz Method , Robust Averaging Method และ Maximum Likelihood and Bayesian Estimation ซึ่งในสถานการณ์ใดสถานการณ์หนึ่ง Algorithm Portfolio จะเลือกอัลกอริทึม ¼ มาประมวลผลเท่านั้น โดยใช้เทคนิค AdaBoost (Adaptive Boosting) ในการเลือกอัลกอริทึม ซึ่งเทคนิค AdaBoost มีปัจจัยในการเลือกอัลกอริทึมอยู่ 4 ปัจจัยด้วยกันคือ Hour(เวลาในขณะที่ทำนาย) ,Weekday(วันที่ทำนาย) ,Fusion method(วิธีการที่ใช้ในการเลือกที่จอดรถคือ Weight average trust หรือ max trust) และSection identifier(ตำแหน่งของที่จอดรถ)

- Algorithm Portfolio แบบ Classification version จะใช้ปัจจัย(Core feature) ทั้ง 4 ข้อในการสุ่มอัลกอริทึม และจะมีโอกาสสุ่มอัลกอริทึมในการให้คะแนนค่าความน่าเชื่อถือได้ทั้ง 4 อัลกอริทึม จากนั้นระบบจะทำการทำนายตำแหน่งจอดรถที่ว่าง

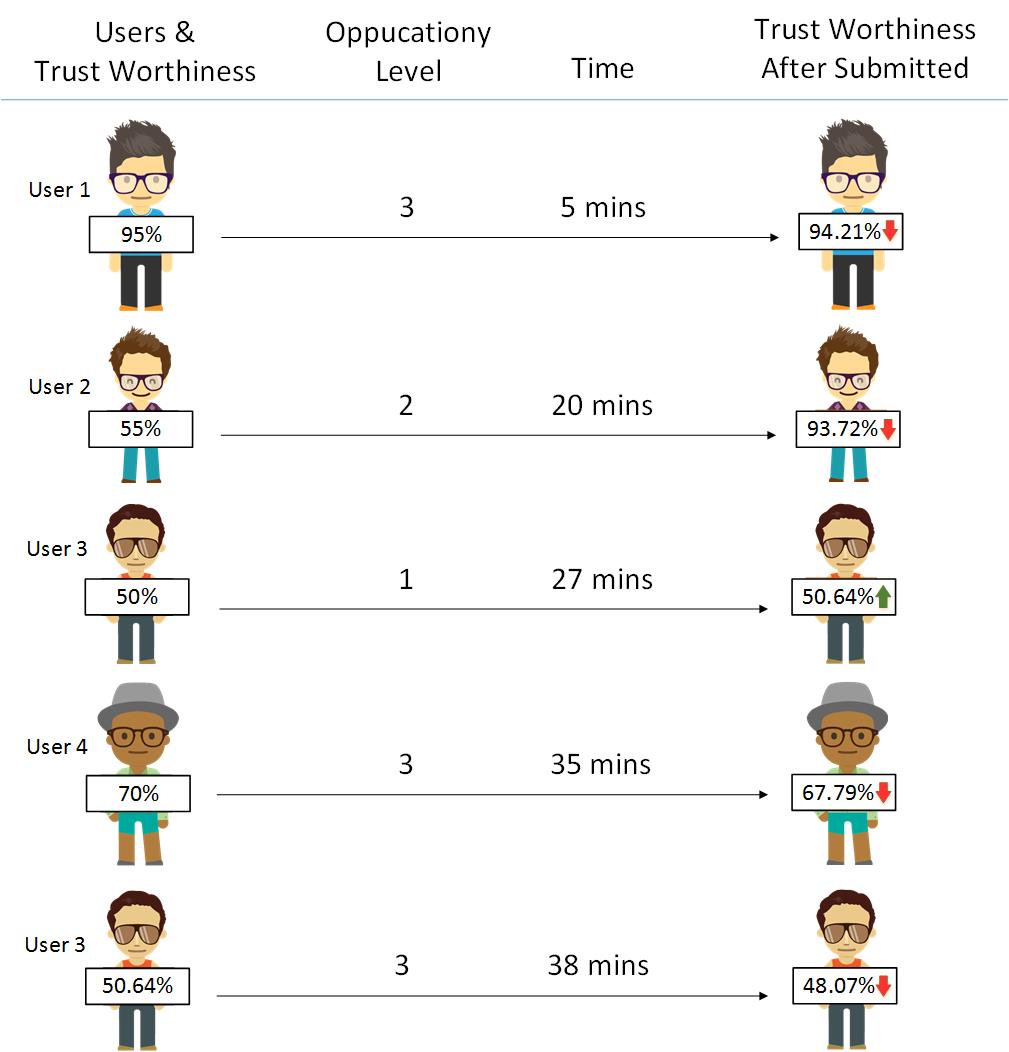
- Algorithm Portfolio แบบ Regression version จะใช้อัลกอริทึม Beta Reputation Method ในการให้คะแนนค่าความน่าเชื่อถือเพียงอัลกอริทึมเดียว จากนั้นจะใช้ปัจจัยทั้ง 4 ข้อในการทำนายตำแหน่งจอดรถที่ว่าง



รูปภาพที่ 5 : Algorithm Portfolio แบบ Classification

version และแบบ Regression version

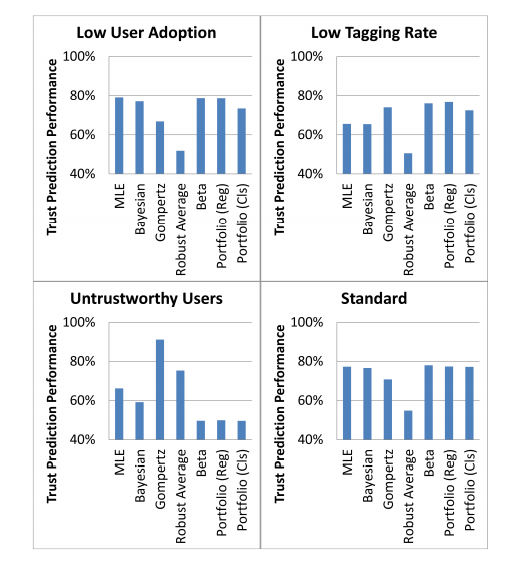
1. Real time update system เป็นอัลกอริทึมในการทำให้ข้อมูลให้มีความสดใหม่เสมอ สาเหตุที่ต้องพัฒนาอัลกอริทึมมาใหม่เนื่องจากการตัดข้อมูลเก่าทิ้งแบบ Data freshness จะแก้ปัญหาในกรณีที่ข้อมูลที่ส่งเข้ามาน้อยเนื่องจากเป็นเวลาที่คนไม่เข้าใช้ที่จอดรถ ซึ่งไม่สามารถแยกแยะในกรณีที่จอดรถเต็มและไม่มีคนเข้ามาใช้งานที่จอดรถ ทำให้เราไม่สามารถตัดข้อมูลที่จอดรถเป็นแบบว่างอยู่ได้ จึงต้องเพิ่มตัวแปรเวลา(time) คือหากข้อมูลที่ส่งมาเป็นเวลานานความน่าเชื่อถือของข้อมูลนั้นจะลดลง และเพิ่มตัวแปร Promote กับ Punish ซึ่งตัวแปรทั้ง 2 เป็นตัวแปรของค่าคะแนนความน่าเชื่อถือของแต่ละบุคคล โดยมีวิธการการเพิ่มลดคะแนนเหมือนเดิมคือ ค่า Promote จะขึ้นทีละน้อยๆ ส่วนค่า Punish จะลดทีละมากๆ จากนั้นนำอัลกอริทึม Real time update system มาทดสอบกับผู้ใช้ 4 คน จากการทดลองผลปรากฏว่าผู้ใช้ทั้ง 4 คนที่ส่งข้อมูลมาในระบบค่าคะแนนความน่าเชื่อถือทั้ง 4 คนลดลงทั้งหมด -->เข้าค่อยเข้าใจว่าทำไมถึงลดลงทุกคน><



รูปภาพที่ 5 : การจำลองวิธี Real time update system

โดยใช้ user 4 คน และจำนวนข้อมูล 5 ข้อมูล

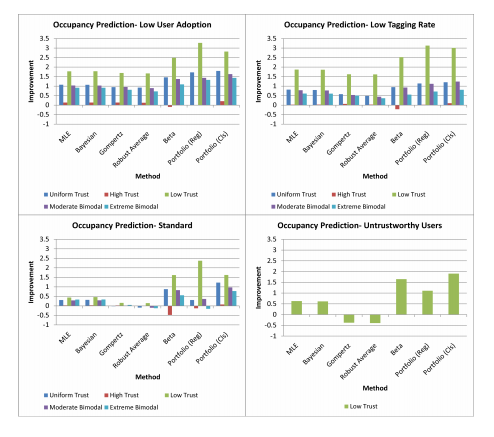
ในส่วนนี้เป็นการทดสอบความแม่นยำของระบบ จะแบ่งออกเป็น 2 ส่วนคือการทดสอบระบบในสถานการณ์จำลอง(Simulate agent base model) กับในสถานการณ์จริง โดยจะวัดอัลกอริทึมใน 2 ส่วนคือ ส่วนการให้ค่าคะแนนความน่าเชื่อถือ กับส่วนที่ทำนายตำแหน่งจอดรถที่ว่าง และมีบริบทที่เป็นส่งผลประสิทธิภาพของอัลกอริทึม ได้แก่ จำนวนผู้ใช้งานแอปพลิเคชัน (User Adoptation) จำนวนข้อมูลที่อยู่ในระบบ (Tagging Rate) และความเชื่อใจทั้งหมดระบบ (Trustworthy) ดังนั้นจึงวัดประสิทธิภาพของอัลกอริทึมต่างๆทั้งหมด 4 สถานการณ์ คือ สถานการณ์ปกติ สถานการณ์ที่คนใช้แอปพลิเคชันน้อย สถานการณ์ที่ข้อมูลที่ได้รับมาน้อย และสถานการณ์ที่ความเชื่อใจทั้งหมดในระบบต่ำ



รูปที่ 7 : ผลการทดสอบประสิทธิภาพของ

อัลกอริทึมที่ให้คะแนนความน่าเชื่อถือของแต่ละบุคคล

จากผลการทดสอบเรื่องการให้คะแนนความน่าเชื่อถือ สรุปผลอัลกอริทึมที่ให้คะแนนความน่าเชื่อถือของแต่ละบุคคลได้ดีที่สุด ¾ สถานการณ์ คือ Beta Reputation Method ส่วนในกรณีที่ระบบความน่าเชื่อถือต่ำ อัลกอริทึมที่ทำงานดีที่สุดคือ Gompertz Method



รูปที่ 8 : ผลการทดสอบประสิทธิภาพของ

อัลกอริทึมที่ใช้ในการหาตำแหน่งที่จอกรถที่ว่าง

จากผลการทดสอบเรื่องการหาตำแหน่งที่จอกรถที่ว่าง สรุปผลอัลกอริทึมที่ทำนายที่จอดรถที่ว่างได้มีประสิทธิภาพมากที่สุด ¾ สถานการณ์ คือ Algorithm Portfolio แบบ Regression version ส่วนในกรณีที่ระบบความน่าเชื่อถือต่ำ อัลกอริทึมที่ทำงานดีที่สุดคือ Algorithm Portfolio แบบ Classification version

ส่วนสุดท้ายคือนำอัลกอริทึมที่วัดจากการ Simulate มาแล้วมาทดสอบกับข้อมูลจริง ผลปรากฎว่า Algorithm Portfolio แบบ Regression version เป็นอัลกอริทึมที่มีประสิทธิภาพดีที่สุดในทั้งส่วนการหาคะแนนความน่าเชื่อถือ และการทำนายที่จอดรถที่ว่าง แต่อัลกอริทึมอื่นก็มีข้อดี และข้อเสียที่ต่างกัน

* + - Beta Reputation Method

ข้อดี : ประมวลได้อย่างรวดเร็ว ไม่ต้องตั้งค่าข้อมูลใดๆในการประมวลผล และทำนายค่าความน่าเชื่อถือได้อย่างแม่นยำ

ข้อเสีย : ทำนายตำแหน่งที่จอดรถได้ไม่ดี

* + - Real time update system

ข้อดี : หาค่าความน่าเชื่อถือ และทำนายตำแหน่งที่จอดรถได้พอใช้

ข้อเสีย : ต้องตั้ง parameter ให้เหมาะสม ถึงจะได้ผลลัพธ์ที่มีประสิทธิภาพ

* + - Algorithm Portfolio

ข้อดี : ทำนายตำแหน่งที่จอดรถได้ดี และยังทำนายค่าความน่าเชื่อถือได้ดีด้วยเนื่องจากใ]ช้วิธี Beta Reputation เช่นกัน

ข้อเสีย : ในกรณีที่มีข้อมูลเข้ามาน้อย จะตัดข้อมูลทิ้งไปเมื่อเกินเวลาที่กำหนด