DS4Biz - *Assignment 2* (20 Marks)

Text Scraping & Classification

Deadline: วันศุกร์ที่ 22 พฤศจิกายน 2562 เวลา 24.00 น.
Submission: Individual github repository – stage, commit with message, and push

Overview

จุดประสงค์ของ Assignment นี้ คือ ให้นศ. ทำการ scrape (รวบรวม) ข้อมูลบทความข่าวจากหน้าเว็บไซต์ (มีจำนวนมากกว่า 1 หน้าเว็บ), ประมวลผลจากข้อความ (text) ของข่าวทั้งหมดที่รวบรวมมา และทำการประเมินประสิทธิภาพของการทำ automated classification ของทุกบทความข่าวทั้งหมดที่อยู่บนเว็บไซต์ โดยใช้หลักการการวัดประสิทธิภาพของ classification problem ใน supervised learning context

Assignment นี้ จะต้องถูกทำลงใน Jupyter Notebook และใช้เพียง Notebook เดียวเท่านั้น (Not a script.) โค้ดที่นศ. เขียนจะ**ต้อง**มีการเขียนอธิบายที่ชัดเจนโดยใช้ Markdown cells เพื่ออธิบายแต่ละ Code cells ที่อยู่ด้านล่าง (ลำดับถัดไป) ของแต่ละ Markdown cell รวมทั้งใช้ inline (#) or block comments (""" … """) ประกอบการอธิบายโค้ดและผลลัพธ์ของการวิเคราะห์ข้อมูลของนศ.

โดยทุก ๆ Code cells จะต้องมีการเขียนอธิบายโดย Markdown cells ว่า แต่ละ Code cells นั้นทำอะไร และแปลความหมายผลลัพธ์ว่าอะไร

Part 1: Data Collection (30%)

เป้าหมายของส่วนที่ 1 นี้ คือ ให้นักศึกษาฝึกการ scrape ข้อมูลจากหน้าเว็บไซต์ โดยทำการรวบรวมข้อมูลข่าวที่ถูก labelled แล้ว**ทั้งหมด** ที่ถูกแสดงตามหมวดหมู่ของข่าว โดยในส่วนนี้นศ.จะต้องทำรายการงาน ทั้งหมดดังต่อไปนี้

1. ระบุหา URLs และ (news) category labels ทั้งหมด สำหรับบทความข่าวทั้งหมดที่มีอยู่ในเว็บไซต์ข้างล่างนี้ โดยโค้ดของนศ.จะต้องใช้ root URL เริ่มต้นข้างล่างนี้ในการหาข่าวที่เหลือทั้งหมด โดยเริ่มจาก URL ตั้งต้นที่ให้
   http://www.it.kmitl.ac.th/~teerapong/news_archive/index.html
2. ให้นศ.ค้นคืนหน้าเว็บไซต์ข่าวทั้งหมดซึ่งสอดคล้องกับข่าวแต่ละข่าวตาม URLs ที่ได้จากการสกัดจากหน้าเว็บไซต์ที่ได้จากตั้งแต่ root URL ที่ได้ข้างต้นและ crawl ตาม anchor links ที่ปรากฏในหน้าโดย scrape เฉพาะหน้าข่าวเท่านั้น และทำการสกัดส่วนที่เป็น body text ซึ่งมีเนื้อหาของบทความของข่าวแต่ละเรื่อง และทำการ save ส่วนของเนื้อหาข่าวเหล่านั้นเป็น plain text โดยใช้เพียง 1 ไฟล์ ซึ่งเนื้อข่าวของแต่ละบทความจะถูกบันทึกไว้ในแต่ละบรรทัด (1 บรรทัด : 1 บทความ) โดยให้ทำการ save เนื้อหาข่าวไว้ที่ folder ชื่อ ../datastore โดยตั้งชื่อไฟล์ให้สื่อความหมาย และใช้นามสกุล *****.txt
3. (optional) title พาดหัวข่าว ของแต่ละบทความ สามารถนำมารวมกับส่วนของ body ได้ แต่ถ้านศ.เลือกวิธีนี้ร่วมด้วย ต้องมีกระบวนการในการใช้งานคำที่ปรากฏใน title แตกต่างจาก คำที่ปรากฏอยู่ใน body และมีการทำการทดลองเปรียบเทียบว่า การใช้ title ประกอบในการทำ Text Classification นั้น ช่วยในเพิ่มประสิทธิภาพของการทำ Text Classification โดยใช้คำจาก body เพียงอย่างเดียวหรือไม่ และเพิ่มหรือลดประสิทธิภาพอย่างไร

4. ให้นศ.ทำการบันทึก category labels ของข่าวทั้งหมด ในไฟล์ที่แยกจาก body เพื่อใช้ในการทำ target variable

Part 2: Text Classification (50%)

เป้าหมายของส่วนที่ 2 นี้ คือ ให้นักศึกษาทำการวิเคราะห์จำแนกหมวดหมู่ของข้อความ (text classification) จากข้อมูลจากที่ได้มาใน part 1 โดยในส่วนนี้นศ.จะต้องทำรายการงาน ทั้งหมดดังต่อไปนี้

1. โหลดชุดของไฟล์ที่เราสร้างขึ้นใน part 1 ลงใน Jupyter Notebook ของตัวเอง (ในแต่ละไฟล์ต้องมี class label โดยยึดตาม category label ที่นศ. ได้การ save ไฟล์แยกเอาไว้จากบทความข่าว โดย class label นั้นจะต้องสัมพันธ์กับข่าวที่ระบุเอาไว้)
2. จากข้อมูลข่าว (raw documents) ที่โหลดมาข้างต้น ให้นศ.สร้าง document-term matrix โดยใช้วิธีที่เหมาะสมในแต่ละขั้นตอน ในการประมวลข้อความเบื้องต้น (text pre-processing) และการถ่วงน้ำหนักของคำ (term weighting) ซึ่งนำไปสู้ประสิทธิภาพของการทำ classification ในลำดับถัดไป พร้อมอธิบายเทคนิคที่นศ.เลือกใช้ในการประมวลผลข้อความเบื้องต้นและการถ่วงน้ำหนักคำ
3. ให้นศ. สร้าง multi-class classification models อย่างน้อย 2 โมเดล หรือมากกว่า โดยใช้ classifiers ต่างประเภทกัน อย่างน้อย 2 ประเภท หรือมากกว่า และทำการ Tune โมเดลให้ได้ประสิทธิภาพสูงที่สุด และอธิบายเหตุผลของการเลือกประเภทโมเดลมาในการทดลอง พร้อมวิธีการ Tune โมเดลที่คิดว่าจะทำให้ได้ประสิทธิภาพสูงที่สุด
4. ให้นศ. ทำการเปรียบเทียบผลลัพธ์ที่ได้จากแต่ละ models ที่นศ. ได้เลือกไว้ ในข้อ 3 โดยนศ .จะต้องเลือกวิธี กลยุทธ์ หรือการนำเสนอที่เหมาะสมในการประเมินประสิทธิภาพและเปรียบเทียบระหว่างโมเดล นศ. จะต้องรายงานและอภิปรายผลลัพธ์การประเมินที่ได้ลงใน Markdown cells ของ Jupyter Notebook ของตนเอง

Code quality and explanation text (20%)

- นศ. ต้องใช้ Markdown cells ในการอธิบายแต่ละขั้นตอนของกระบวนการ โดยนศ. ควรแยกส่วนของ Part 1 – data collection และ Part 2 – classifier evaluation ให้เด่นชัด แต่ยังอยู่ใน Notebook เดียวกัน
- โค้ดที่นศ. เขียนจะต้องอ่านและเข้าใจได้โดยง่าย ชัดเจน และไม่คลุมเครือ โดยควรมี comment อธิบายที่เพียงพอในการทำให้เข้าใจโค้ดได้โดยง่าย แต่ไม่มากจนเกินไป
- ความซับซ้อนของโค้ดอยู่ในระดับเท่าที่จำเป็น และมีการใช้ Package ต่าง ๆ ที่เหมาะสม โดยเกณฑ์หลัก ๆ จะดูจากวิธีการเซ็ตการทดลองเชิงเปรียบเทียบว่าสมเหตุสมผลหรือไม่ สอดคล้องตามเป้าหมายตามที่นศ. มีเจตนารมณ์ ในการทำหรือไม่ ซึ่งการให้คะแนนจะพิจารณาประสิทธิภาพของ

classifier เป็นเรื่องรอง และความเร็วในการประมวลผลเป็นเรื่องรอง (หากการทดลองไม่เวิ่นเว้อจนเกินจุดประสงค์)

Guidelines:

- สำหรับ assignment นี้ อนุญาตให้นศ. ใช้เฉพาะ third-party packages เหล่านี้เท่านั้นในการทำ assignment ได้แก่: **NumPy, Pandas, Scikit-learn, NLTK, SciPy, Requests, BeautifulSoup, Matplotlib, Seaborn** หากใครใช้มากกว่านี้ ต้องขออนุญาตก่อน มิฉะนั้นจะหักคะแนน package ละ 10% และหากมีการอนุญาต ก็จะประชาสัมพันธ์ให้นศ. คนอื่นใช้ได้ด้วยเช่นเดียวกัน
- ให้นศ. ทำการส่ง Assignment ซึ่งคือ Jupyter Notebook ของนศ. พร้อมข้อมูลที่รวบรวมมา ใน Github repository ของนศ. แต่ละคน โดยในแต่ละ Jupyter Notebook ของนศ. นศ.จะต้องเขียนชื่อ นามสกุล รหัสนศ. ลงใน Markdown cell แรกของ Notebook
- Assignment นี้เป็นงานเดี่ยว ของนศ. แต่ละคน หากมีการตรวจสอบพบการคัดลอก (Plagiarism) จะได้ 0 คะแนนในของส่วน Assignment นี้ หากมีข้อสงสัย และหากมีหลักฐานชัดเจนว่ามีการคัดลอกงานจากแหล่งใด ๆ ก็ตาม นศ. จะได้เกรด F ในวิชานี้ และส่งเรื่องต่อให้กับทางคณะฯ และสถาบันฯ ต่อไป
- Hard deadline: วันศุกร์ที่ 22 พฤศจิกายน 2562 เวลา 24.00 น.
  - ส่งช้า 1-5 วัน: ลด 20% จากคะแนนตรวจที่ได้ (ขอเปิด Github ให้ส่งช้า)
  - ส่งช้า 6-10 วัน: ลด 40% จากคะแนนตรวจที่ได้ (ขอเปิด Github ให้ส่งช้า)
  - จะไม่มีการรับตรวจ Assignment หากส่งช้าเกิน 10 วัน โดยปราศจากหลักฐานชี้แจงเหตุผลในการส่งงานช้า ได้แก่ หลักฐานด้านการแพทย์ว่าเข้านอนโรงพยาบาลเพื่อรับการรักษา