

Computational Linguistics

Contest 1: Aspect-based Sentiment Analysis

Contest นี้มีจุดประสงค์ คือ ให้สร้าง Aspect-based sentiment analyzer ที่มีความแม่นยำมากที่สุด Aspect-based sentiment analyzer จะต้องบอกว่าข้อความนั้นพูดถึงเกี่ยวกับอะไร (aspect) และพูดถึงในแง่ไหน (sentiment) เพราะฉะนั้นระบบจะประกอบไปด้วยสองส่วน ส่วนแรกต้อง classify aspect อีกส่วนต้อง classify sentiment เมื่อนำการทำนายจากทั้งสองส่วนมารวมกัน คำตอบจะถูกตัดสินว่าถูกต้องถ้าทั้ง aspect และ sentiment ถูกต้อง คะแนนที่จะได้ขึ้นอยู่กับประสิทธิภาพของระบบโดยดูจาก Overall F1 score (50%) และกระบวนการพัฒนาโมเดล (50%) ซึ่งต้องพูดถึงในรายงานที่ต้องส่งด้วย

นิสิตสามารถเลือกใช้โมเดลใดก็ได้ที่เรียนมา แต่บังคับว่าต้องทำ Logistic Regression กับ bag-of-word features เป็นอย่างน้อยที่สุด และต้องรายงานผลไว้ในรายงานการทดลองด้วย ถึงแม้ว่าจะไม่ใช่โมเดลที่ดีที่สุดที่เราส่งในตอนท้าย

งานนี้สามารถทำเดี่ยวหรือทำเป็นคู่ก็ได้
งานนี้ไม่อนุญาตให้ส่งเลตได้เพราะต้องนำคะแนนความแม่นยำของทุกคนมาประกอบกัน

สิ่งที่ให้มา

1. Dataset ซึ่งเราจะแบ่งเป็น training set และ dev set อย่างไรก็ได้
2. Test set ซึ่งจะยังไม่มี label ให้
3. [evaluate.py](#)

ในการคำนวณ Overall F1 score – prediction จะจัดว่าถูกก็ต่อเมื่อทั้ง aspect และ sentiment ถูกต้อง ตัวอย่างการใช้ (ต้องใช้ terminal หรือ anaconda prompt)

```
>>> python evaluate.py contest1_train.csv (ชื่อไฟล์).csv
```

```
$ python3 evaluate.py lab4_train.csv pred1.csv
=== CLASSIFICATION : ASPECT ===
      class name  precision  recall  F1-score  support
0          food      0.801    0.754    0.777    203
1          price      0.852    0.383    0.529     60
2          service    0.847    0.587    0.693    104
3          ambience  0.700    0.394    0.505     71
4 anecdotes/miscellaneous 0.790    0.680    0.731    194
5          MACRO AVG    0.798    0.560    0.647    632
6          MICRO AVG    0.799    0.628    0.703    632

=== CLASSIFICATION : SENTIMENT ===
      class name  precision  recall  F1-score  support
0      positive    0.749    0.817    0.781    306
1      negative    0.511    0.376    0.433    125
2       neutral    0.422    0.257    0.319     74
3      conflict    0.308    0.333    0.320     24
4      MACRO AVG    0.497    0.446    0.463    529
5      MICRO AVG    0.652    0.612    0.632    529

=== CLASSIFICATION : OVERALL ===
      precision  recall  F1-score  support
0      MICRO AVG    0.551    0.434    0.485    632
```

สิ่งที่ต้องส่ง

1. ส่ง prediction csv ของ test set ได้ทุกวันก่อนเดดไลน์ ถ้าเราได้เก็บค่า prediction เอาไว้ใน DataFrame ([ตัวอย่างสร้างตาราง predict และวิธีใช้](#)) ที่มี column เหมือนกับ training set ('id','aspectCategory','polarity') และจากนั้นเก็บใส่ไฟล์ที่ชื่อว่า test_pred.csv

หมายเหตุ: ถ้า text ไหนมีมากกว่า 1 aspect ให้เพิ่ม row เข้าไปแต่ให้คง id ของ text เอาไว้ เพราะฉะนั้น csv ที่ส่งอาจจะมีจำนวนแถวมากกว่า test set csv ตัวต้น เช่น

id	aspectCategory	polarity
123	food	positive
368	service	negative
368	price	positive

(test set csv มีแค่ id กับ text เท่านั้น ซึ่งค่าของ id ไม่ซ้ำกันอย่าง train set เพราะไม่มีเฉลยที่เป็น gold standard label)

2. รายงานว่า model ที่ส่งนั้นมีที่มาที่ไปอย่างไร (ไม่เกิน 4 หน้า) ส่งบน gradescope
ในรายงานต้องพูดถึง
 - a. Experimental Setup: รายละเอียดการแบ่งข้อมูลสำหรับการ train และ dev รวมถึง label distribution
 - b. Model: โมเดลที่ได้ทดสอบมาทั้งหมดที่ได้ผลดี และได้ผลไม่ดี และต้องพูดถึงรายละเอียดของ Hyperparameter เช่น ใช้ feature อะไรบ้าง หรือใช้ hidden layer กี่ชั้น ชั้นละกี่ unit หรือ embedding ถูก pretrain มาจากไหนก่อนหรือไม่ เพราะอะไรจึงคิดว่าโมเดลเหล่านี้จึงจะมีประสิทธิภาพที่ดี
 - c. Results: ตารางเปรียบเทียบผลการทดลองรวมทั้งข้อสรุปที่ได้และเหตุผลประกอบ ในตารางเปรียบเทียบผลต้องมีผลของ Logistic Regression ที่ใช้ bag-of-word features
 - d. Conclusion: สรุปว่าโมเดลอะไรที่ใช้ในการเอาไปใช้กับ test set บน gradescope โมเดลที่ส่งไปสามารถเทียบกับ training set ทั้งชุดได้
3. โค้ด

เกณฑ์การให้คะแนนส่ง

- 50% Overall F1 score on test set
 - อันดับ Leader board top 1-30% = 50%
 - อันดับ Leader board top 31-60% = 45%
 - อันดับ Leader board top 61-100% = 40%

- 30% คุณภาพของกระบวนการพัฒนาโมเดล
 - การ set up การทดลองจะต้องมีการแบ่งข้อมูลอย่างเหมาะสม
 - ควรแสดงให้เห็นถึงความเข้าใจในหลักการพัฒนาโมเดลให้แม่นยำมากขึ้น
เข้าใจว่าปัจจัยอะไรบ้างที่ผลต่อความแม่นยำของโมเดล
- 20% คุณภาพของการเขียน ต้องอ่านง่าย ชัดเจน