第9回最先端NLP勉強会（2017.9.15）

# Detecting annotation noise in automatically labelled data

Ines Rehbein & Josef Ruppenhofer

紹介者：笹野（名大）

# Summary

- ACL'17のOutstanding Paperの1つ
- タスクは自動アノテーションのエラー検出

Given: 複数の事例に対する複数の自動解析器の出力
+ 各事例の出力決定方法

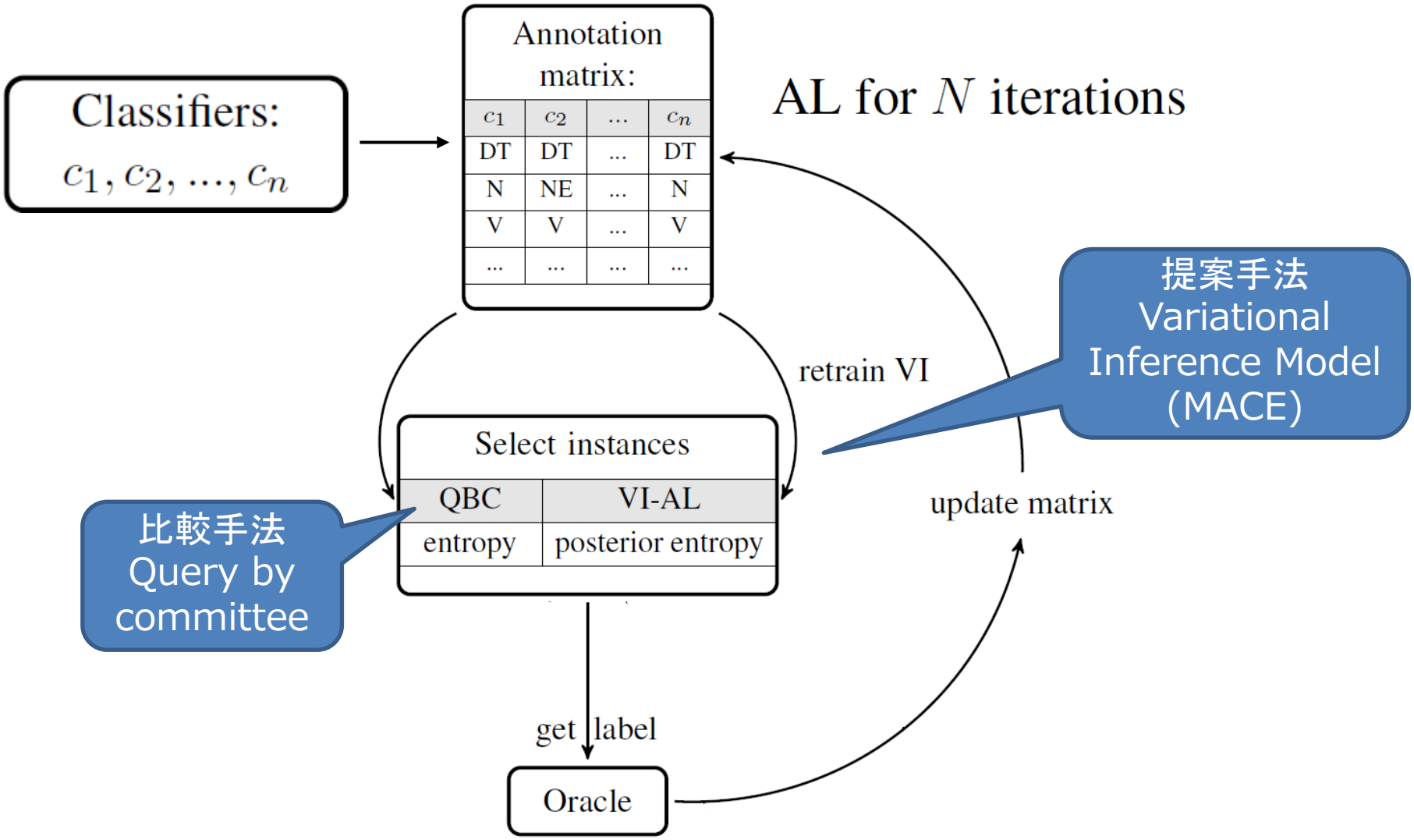
Goal: 以下を交互に実行し高品質の言語資源を作成

1. 誤り可能性が高い出力を検出
2. 人手で正しいラベルを付与

Annotation matrix:

| $c_1$ | $c_2$ | … | $c_n$ |
|---|---|---|---|
| DT | DT | … | DT |
| N | NE | … | N |
| V | V | … | V |
| … | … | … | … |

# Error detection procedure



# The difference from Active Learning (AL)

| | Typical AL | This work |
|---|---|---|
| Goal | Improving the accuracy of a machine learning system | Improving the quality of an existing language resource |
| Retrain | Need to retrain the baseline classifiers | Do not retrain the baseline classifiers |

# Example of Experimental Results (POS tagging for 5000 tokens)

比較手法

提案手法

検出数

| $N$ | QBC # tp | QBC ED prec | QBC rec | VI-AL # tp | VI-AL ED prec | VI-AL rec |
|---|---|---|---|---|---|---|
| 100 | 85 | **85.0** | **13.5** | 75 | 75.0 | 11.9 |
| 200 | 148 | **74.0** | **23.5** | 146 | 73.0 | 23.2 |
| 300 | 198 | 66.0 | 31.4 | 212 | **70.7** | **33.6** |
| 400 | 239 | 59.7 | 37.9 | 278 | **69.5** | **44.1** |
| 500 | 282 | 56.4 | 44.8 | 323 | **64.6** | **51.3** |
| 600 | 313 | 52.2 | 49.7 | 374 | **62.3** | **59.4** |
| 700 | 331 | 47.3 | 52.5 | 412 | **58.9** | **65.4** |
| 800 | 355 | 44.4 | 56.3 | 441 | **55.1** | **70.0** |
| 900 | 365 | 40.6 | 57.9 | 465 | **51.7** | **73.8** |
| 1000 | 371 | 37.1 | 58.9 | 484 | **48.4** | **76.8** |

# Query by committee [Seung+'92]

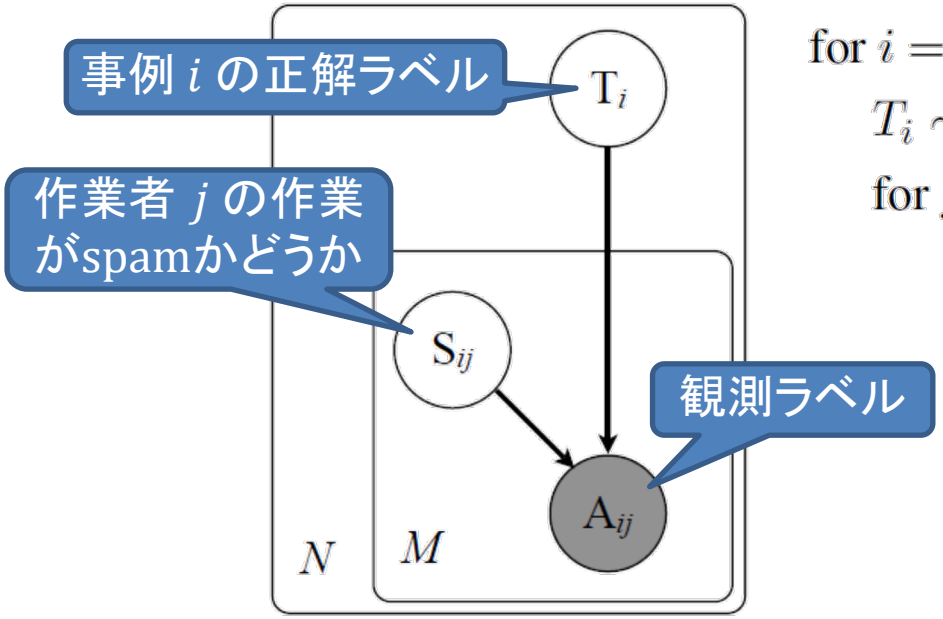
- QCB: One of the selection strategies for Active Learning (AL)
- Uses a classifier ensemble (committee)
- Selects the instances that show maximal disagreement

i.e. the instance with the highest entropy

$$H = -\sum_{m=1}^{M} P(y_i = m) \log P(y_i = m)$$

# MACE: Multi-Annotator Competence Estimation [Hovy+'13] (1/2)

- Generative model of the annotation process
  - The Correct label is treated as latent variables
  - Assuming an annotator always produces the correct label when s/he tries to



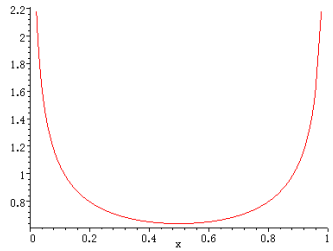

for $i = 1 \ldots N$ :
  $T_i \sim$ Uniform
  for $j = 1 \ldots M$ :
    $S_{ij} \sim$ Bernoulli$(1 - \theta_j)$
    if $S_{ij} = 0$ :
      $A_{ij} = T_i$
    else :
      $A_{ij} \sim$ Multinomial$(\xi_j)$

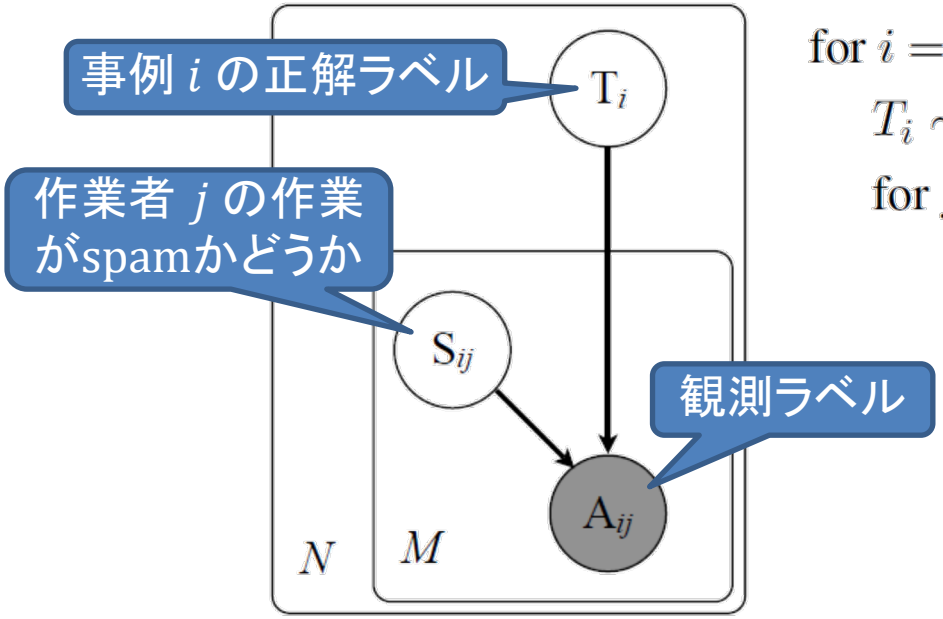
Model parameters

# MACE: Multi-Annotator Competence Estimation [Hovy+'13] (2/2)

- パラメータに事前分布を導入(⇔EM版)
  - 変分法で事後分布を導出(Variational Inference; VI)
  - EM版だと$\theta_j$はlinearな分布

either an annotator tried to get the right answer, or simply did not care, but (almost) nobody tried "a little"





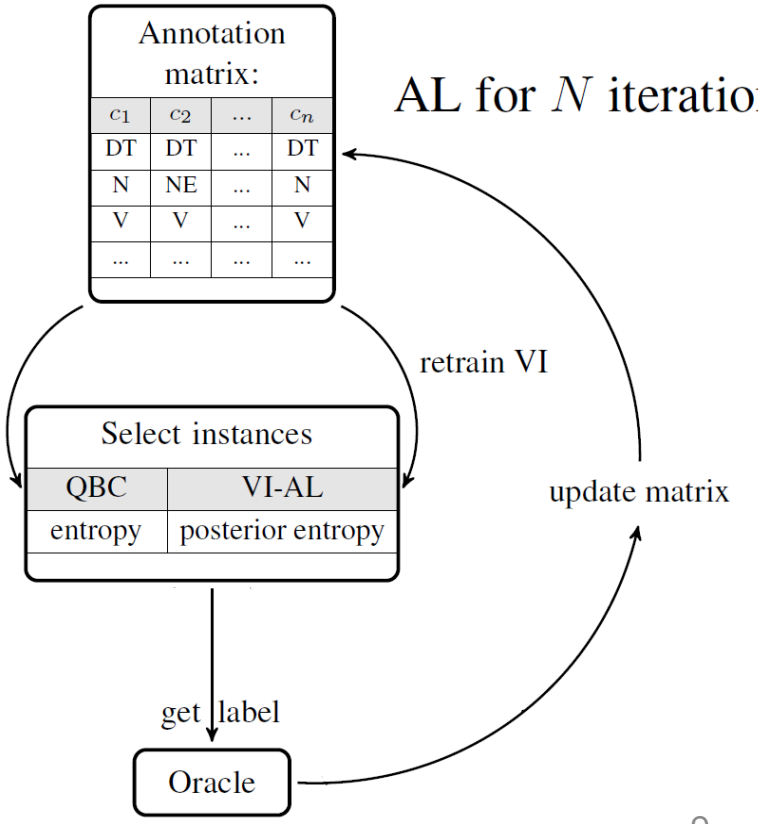

$$
\begin{aligned}
&\text{for } i = 1 \ldots N: \\
&\quad T_i \sim \text{Uniform} \\
&\quad \text{for } j = 1 \ldots M: \\
&\qquad S_{ij} \sim \text{Bernoulli}(1 - \theta_j) \\
&\qquad \text{if } S_{ij} = 0: \\
&\qquad\quad A_{ij} = T_i \\
&\qquad \text{else}: \\
&\qquad\quad A_{ij} \sim \text{Multinomial}(\xi_j)
\end{aligned}
$$

symmetric Beta priors (α=β=0.5)

Symmetric Dirichlet priors (hyper-parameters are clamped to 10.0)

# 提案手法: VI-AL

- Guiding variational inference with AL
  - 以下を繰り返す
    1. 変分推論 by MACE
    2. (おそらく$T_j$の)事後エントロピーが最大となる事例を選択
    3. 人手で正解を付与(oracle)
       - 付与した正解を次の反復時に事前知識として使用($T_i$の?)
       - ランダムに選んだ1つの分類器の予測結果をoracleに置換



# Experiments

- Simulation studies
  1. POS tagging (in-domain setting)
     - Train & Test: Penn Treebank
  2. POS tagging (out of domain setting)
     - Train: Penn Treebank, Test: Web treebank
  3. German NER (out of domain setting)
     - Train: HGC or DeWaC corpus, Test: Europarl corpus
- Real AL scenario with a human annotator
  4. POS tagging (out of domain setting) (=2)
     - Train: Penn Treebank, Test: Web treebank
     - 2より少しだけ低い精度（詳細省略）

# Tools for POS tagging & in-domain accuracies

- 7 tools
  - bi-LSTM-aux [Plank+'16]
  - HunPos [Halácsy+'07] (=HMM)
  - Stanford [Toutanova+'03] (=ME)
  - SVMTool [Jiménez+'04]
  - TreeTagger [Schmid'99]
  - TWeb [Ma+'14] (=Easy-first+NN)
  - Wapiti [Lavergne+'10] (=CRF)

WSJ with 20,000 tokens

| Tagger | Acc. |
|---|---|
| bilstm | 97.00 |
| hunpos | 96.18 |
| stanford | 96.93 |
| svmtool | 95.86 |
| treetagger | 94.35 |
| tweb | 95.99 |
| wapiti | 94.52 |
| avg. | 95.83 |
| majority vote | **97.28** |
| MACE | 97.27 |

ほとんど差なし

# 1. POS tagging (in-domain setting)

| | QBC | | VI-AL | |
|---|---|---|---|---|
| $N$ | **label acc** | **ED prec** | **label acc** | **ED prec** |
| 0 | 97.58 | - | 97.56 | - |
| 100 | 97.84 | 13.0 | 98.42 | **41.0** |
| 200 | 97.86 | 7.0 | 98.90 | **33.0** |
| 300 | 97.90 | 5.3 | 99.16 | **26.3** |
| 400 | 97.82 | 3.0 | 99.26 | **21.0** |
| 500 | 97.92 | 3.4 | 99.34 | **17.6** |

Label accuracies on 5,000 tokens after *N* iterations

# Tagger accuracies on different genres in Web treebank [Bies+'12]

| | answer | email | newsg. | review | weblog |
|---|---|---|---|---|---|
| bilstm | 85.5 | 84.2 | 86.5 | 86.9 | 89.6 |
| hun | 88.5 | 87.4 | 89.2 | 89.7 | 92.2 |
| stan | **89.0** | 88.1 | **89.9** | 90.7 | 93.0 |
| svm | 87.4 | 86.1 | 88.2 | 88.8 | 91.3 |
| tree | 86.8 | 85.6 | 87.1 | 88.7 | 87.4 |
| tweb | 88.2 | 87.1 | 88.5 | 89.3 | 92.0 |
| wapiti | 85.2 | 82.4 | 84.6 | 86.5 | 87.3 |
| avg. | 87.2 | 85.8 | 87.7 | 88.7 | 90.4 |
| major. | 87.4 | **88.8** | 89.1 | 90.9 | 93.8 |
| MACE | 87.4 | 88.6 | 89.1 | **91.0** | **93.9** |

ほとんど差なし

# 2. POS tagging (out of domain setting)

| | QBC | | | VI-AL | | |
|---|---|---|---|---|---|---|
| | **# tp** | **ED prec** | **rec** | **# tp** | **ED prec** | **rec** |
| answer | 282 | 56.4 | 44.8 | 323 | **64.6** | **51.3** |
| email | 264 | **52.8** | **47.1** | 261 | 52.2 | 46.6 |
| newsg. | 195 | 39.0 | 36.0 | 214 | **42.8** | **39.6** |
| review | 227 | 45.4 | 49.7 | 255 | **51.0** | **55.8** |
| weblog | 166 | 33.2 | 54.6 | 196 | **39.2** | **64.5** |

No. of true positives (# tp), precision (ED prec) and recall for error detection on 5,000 tokens after 500 iterations on all web genres

| | QBC | | | VI-AL | | |
|---|---|---|---|---|---|---|
| $N$ | **# tp** | **ED prec** | **rec** | **# tp** | **ED prec** | **rec** |
| 100 | 85 | **85.0** | **13.5** | 75 | 75.0 | 11.9 |
| 200 | 148 | **74.0** | **23.5** | 146 | 73.0 | 23.2 |
| 300 | 198 | 66.0 | 31.4 | 212 | **70.7** | **33.6** |
| 400 | 239 | 59.7 | 37.9 | 278 | **69.5** | **44.1** |
| 500 | 282 | 56.4 | 44.8 | 323 | **64.6** | **51.3** |
| 600 | 313 | 52.2 | 49.7 | 374 | **62.3** | **59.4** |
| 700 | 331 | 47.3 | 52.5 | 412 | **58.9** | **65.4** |
| 800 | 355 | 44.4 | 56.3 | 441 | **55.1** | **70.0** |
| 900 | 365 | 40.6 | 57.9 | 465 | **51.7** | **73.8** |
| 1000 | 371 | 37.1 | 58.9 | 484 | **48.4** | **76.8** |

No. of true positives (# tp), precision (ED prec) and recall for error detection on 5,000 tokens from the answers set

# 3. German NER

- Baseline classifierには、HGCとDeWacでそれぞれ学習したGermaNER[Benikova+'15]とStanfordNER[Finkel+'09]を使用(2×2=4)
- そもそものエラー数がMajority vote (1756 errors)の方がMACE (1628 errors)より多いので注意 (⇔ POS Tagging)

| | QBC | | | VI-AL | | |
|---|---|---|---|---|---|---|
| $N$ | **# tp** | **ED prec** | **rec** | **# tp** | **ED prec** | **rec** |
| 100 | 54 | 54.0 | 3.1 | 76 | **76.0** | **4.7** |
| 200 | 113 | 56.5 | 6.4 | 155 | **77.5** | **9.6** |
| 300 | 162 | 54.0 | 9.2 | 217 | **72.3** | **13.4** |
| 400 | 209 | 52.2 | 11.9 | 297 | **74.2** | **18.2** |
| 500 | 274 | 54.8 | 15.6 | 352 | **70.4** | **22.3** |
| 600 | 341 | 56.8 | 19.4 | 409 | **68.2** | **25.5** |
| 700 | 406 | 58.0 | 23.1 | 452 | **64.6** | **27.8** |
| 800 | 480 | 60.0 | 27.3 | 483 | **60.4** | **29.8** |
| 900 | 551 | **61.2** | 31.4 | 512 | 56.9 | **31.9** |
| 1000 | 617 | **61.7** | 35.1 | 585 | 58.5 | **35.8** |
| 1000 | remaining errors:1,139 | | | remaining errors:1,043 | | |

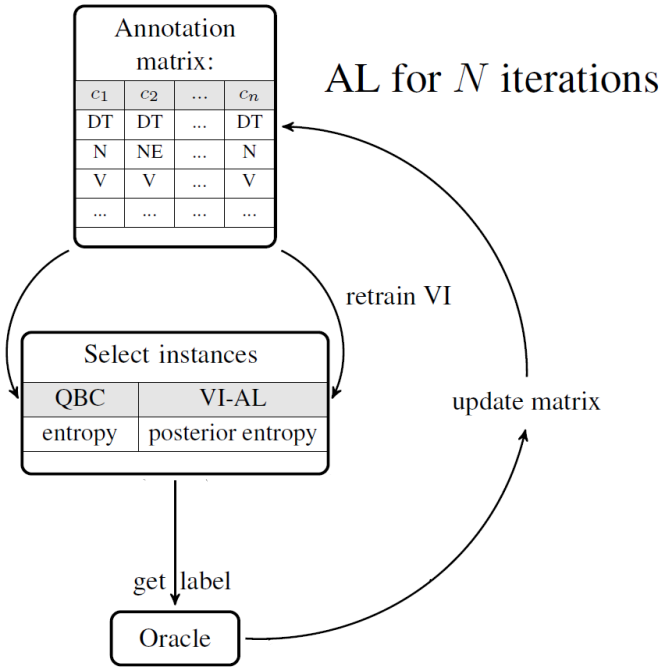
# Summary

- 自動アノテーションのエラー検出法を提案
  - 教師なし生成モデル (MACE/VI)
    + Active Learning (AL)

Given: 複数の自動解析器の出力
+ 各事例の出力決定方法

Goal: 以下を交互に実行し高品質の言語資源を作成

1. 誤り可能性が高い出力を検出
2. 人手で正しいラベルを付与



# 感想

- タスクの設定が面白い
  - 論文中でも主張(severely understudied problem)されている通りあまり見たことがないタスク
  - Active Learningとの違い(目的が高品質データの構築&再学習が不要)がポイント
- 実際に(そこそこの品質の)大規模データの構築に使えそう
  - ソースは公開: https://github.com/julmaxi/MACE-AL
    (MACE: https://www.isi.edu/publications/licensed-sw/mace/)
  - ただし、性質の異なる複数の解析器の存在が前提
  - かつ、一定以上の精度が必要かもしれない？
    (論文中のタスクはいずれもベースの解析器が85%以上の精度)

# よく分からないor自信がない点

- なぜQBCと比べて大幅に精度が良いのか
  - ALの結果が反復時において反映されるのが主な違いだが、事後確率は観測ラベルと類似しそう
  - 各分類器はspamではないはずなので、$S_{ij}$に事前分布を設定する必然性がMACEより弱い
- VIにおいてハイパーパラメターを更新するのかどうか
  - 先行研究（MACE）では固定している
- なぜ1つの分類器の予測結果をoracleに置換するか
  - 人手で付与したoracleを、次の反復時において事前知識（$\mathrm{T}_i$の?）として使用するだけでは十分でないのか