HSL748 Natural Language Understanding

Dawid-Skene Algorithm for Unreliable Data

Presented by: Tanvi Gandhi (HCS237023)

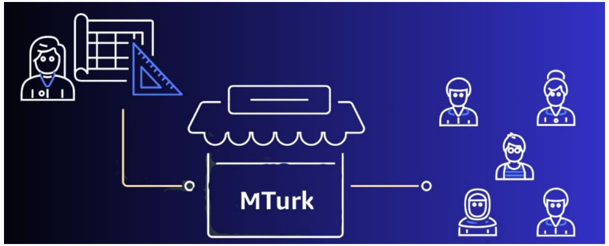
Introduction - Crowdsourcing

“Models are only as good as the data they learn from”

- In today’s data driven world, data collection is as as important as the modelling process.
- **Annotators** are used for tasks such as creating part-of-speech tagged corpora, sentiment analysis datasets, and more, which are foundational for training NLP models.
- Traditional annotation methods face challenges such as time limitations, budget constraints, and resource availability.
- **Crowdsourcing** emerged as a vital solution by facilitating quicker and more cost-effective data labeling by tapping into a global pool of annotators.
- There should be at least more than 10 annotators for each task

Introduction - Crowdsourcing

- Platforms like Amazon Mechanical Turk (MTurk) can be used to gather/label data by paying some monetary compensation.
  - diverse pool of "**turkers**"
  - Human Intelligence Tasks (HITs)



- Essential to ensure accuracy, quality and consistency.
- **Gold standard** - benchmark or "ground truth".
- To improve reliability - trap questions, pre-tests (e.g., language proficiency), time tracking, training.
- Event then, there can be individual biases or errors that can affect the data.
- How to get the best of crowdsourcing?

Problem Statement

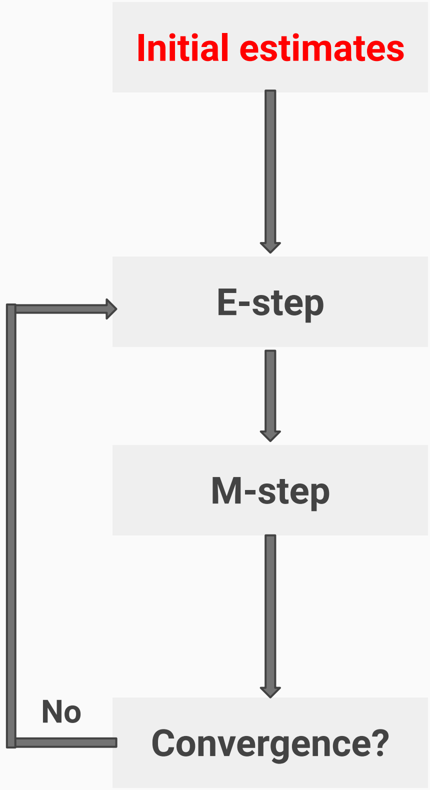
- To reduce unreliability in the collected data, it is essential to achieve crowd consensus.
- The impact of any single erroneous judgment is diluted by the majority, thus reducing the overall error rate.
- Helps in repeated confirmation of data points through multiple independent sources.
- One of the easiest way it to take majority voting.
- Others are weighted voting, iterative refinement.
- How to cater the unreliability of the annotators and minimize its impact?
- **How to determine the most likely true class from a mix of reliable and unreliable POS tagged data?**

Dawid-Skene algorithm

- Developed by Alexander Dawid and Allan Skene in 1979.
- Principle of **expectation maximization (EM)**.
- Operates under the assumption that not all annotators are equally reliable, and their errors are not randomly distributed but can be statistically modeled.
- Annotators make errors independently of each other and have different error rates, which the algorithm tries to estimate.
- **Dynamic** - It estimates the reliability of each annotator based on their agreement with consensus and uses these reliability scores to weight their annotations in determining the true labels.
- Thus, each annotator influences the final decision is weighted by their demonstrated reliability.

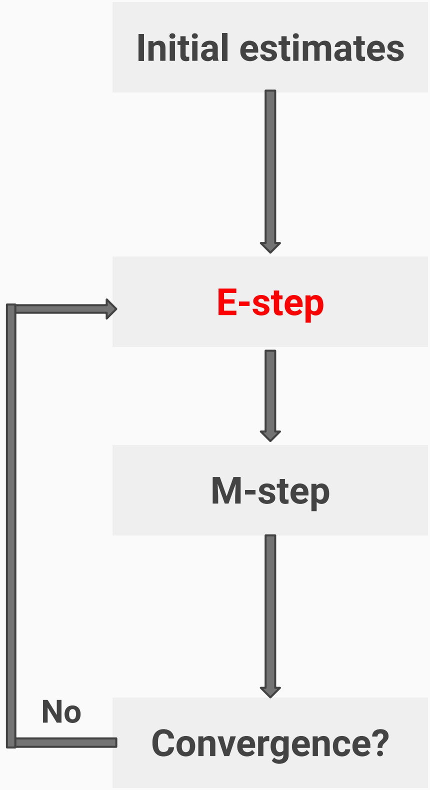
Dawid-Skene algorithm

- **Input:** Each item in a set labeled by several annotators.
- **Initial Estimates**:
  - **Annotator Reliability**: Initialize the reliability (or error rates) of each annotator. This can be uniform or based on prior knowledge if available.
  - **True Labels**: Initialize the probabilities of each possible true label for each item. Usually by assuming equal likelihood for all labels.



Dawid-Skene algorithm

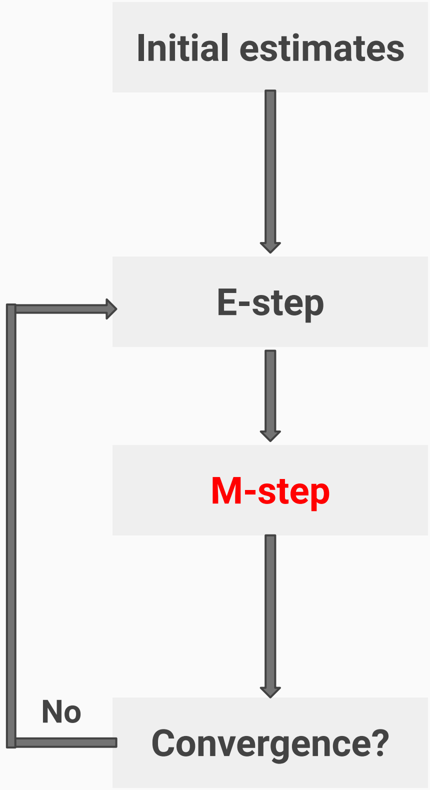
- **Expectation step** - Given the current estimates of annotators' reliability (**error rates**), the algorithm calculates the posterior probabilities of the true labels of each item.
- This means estimating the probabilities of the true responses given the observed data and current parameter estimates.
- This involves computing a weighted average of the annotations, where the weights are derived from the estimated reliability of each annotator.
- Thus, this step helps in estimating the most probable true label for each item.



Dawid-Skene algorithm

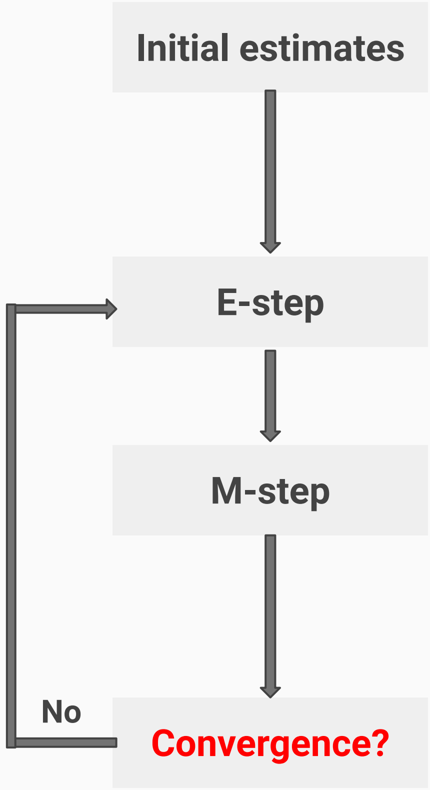
- **Maximization step** - Based on the estimated true label probabilities from the E-step, it updates the estimates of the error rates for each annotator. This step maximizes the likelihood of the observed annotations given the estimated true labels.

- This step updates the error rates for each annotator based on how well their annotations match the estimated true labels from the E-step.

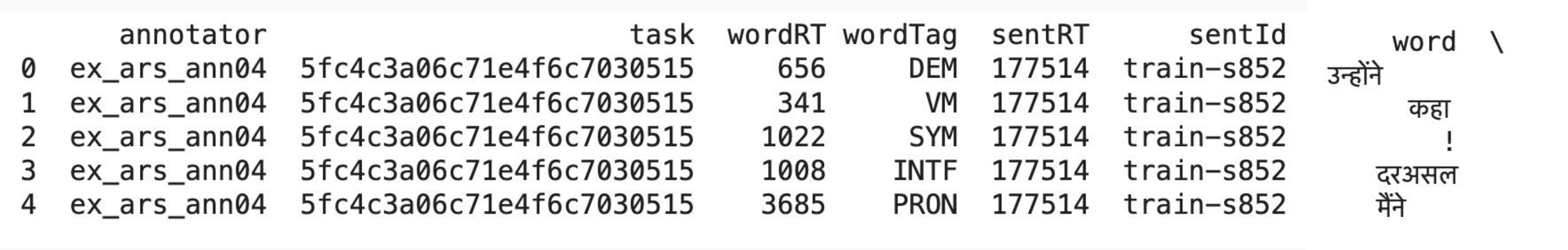


Dawid-Skene algorithm

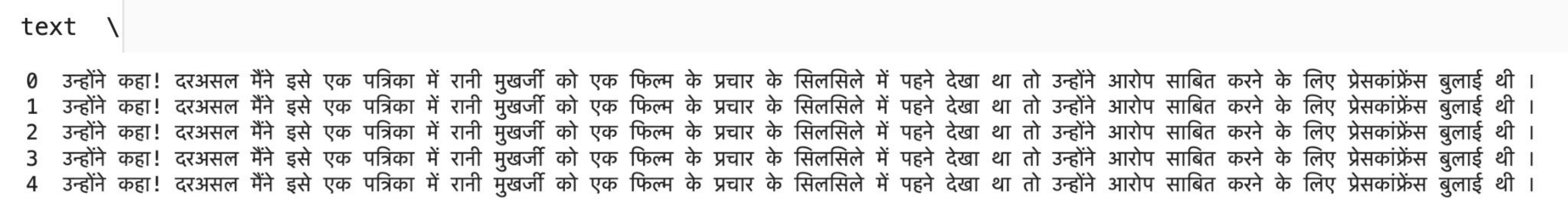
- Repeat the E-step and M-step until the changes in parameter estimates are sufficiently small, indicating convergence.
- The model estimates two main probabilities:
    - The probability that an item belongs to a certain class (the **true label distribution**).
    - The **error rates** of each annotator.
- Sometimes, convergence is determined by simply setting a maximum number of iterations.



Data and Resources

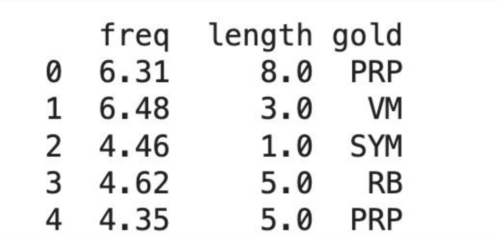


```
      annotator                      task  wordRT wordTag  sentRT      sentId      word  \
0  ex_ars_ann04  5fc4c3a06c71e4f6c7030515     656     DEM  177514  train-s852   उन्होंने
1  ex_ars_ann04  5fc4c3a06c71e4f6c7030515     341      VM  177514  train-s852       कहा
2  ex_ars_ann04  5fc4c3a06c71e4f6c7030515    1022     SYM  177514  train-s852         !
3  ex_ars_ann04  5fc4c3a06c71e4f6c7030515    1008    INTF  177514  train-s852    दरअसल
4  ex_ars_ann04  5fc4c3a06c71e4f6c7030515    3685    PRON  177514  train-s852      मैंने
```



```
text  \

0  उन्होंने कहा! दरअसल मैंने इसे एक पत्रिका में रानी मुखर्जी को एक फिल्म के प्रचार के सिलसिले में पहने देखा था तो उन्होंने आरोप साबित करने के लिए प्रेसकांफ्रेंस बुलाई थी ।
1  उन्होंने कहा! दरअसल मैंने इसे एक पत्रिका में रानी मुखर्जी को एक फिल्म के प्रचार के सिलसिले में पहने देखा था तो उन्होंने आरोप साबित करने के लिए प्रेसकांफ्रेंस बुलाई थी ।
2  उन्होंने कहा! दरअसल मैंने इसे एक पत्रिका में रानी मुखर्जी को एक फिल्म के प्रचार के सिलसिले में पहने देखा था तो उन्होंने आरोप साबित करने के लिए प्रेसकांफ्रेंस बुलाई थी ।
3  उन्होंने कहा! दरअसल मैंने इसे एक पत्रिका में रानी मुखर्जी को एक फिल्म के प्रचार के सिलसिले में पहने देखा था तो उन्होंने आरोप साबित करने के लिए प्रेसकांफ्रेंस बुलाई थी ।
4  उन्होंने कहा! दरअसल मैंने इसे एक पत्रिका में रानी मुखर्जी को एक फिल्म के प्रचार के सिलसिले में पहने देखा था तो उन्होंने आरोप साबित करने के लिए प्रेसकांफ्रेंस बुलाई थी ।
```

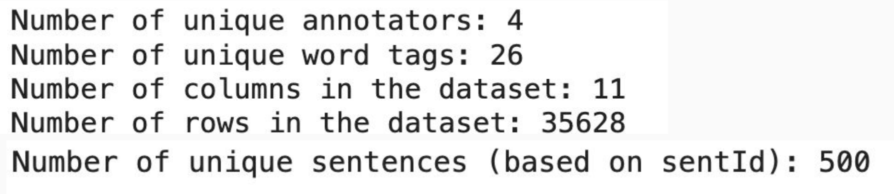


```
   freq  length gold
0  6.31     8.0  PRP
1  6.48     3.0   VM
2  4.46     1.0  SYM
3  4.62     5.0   RB
4  4.35     5.0  PRP
```

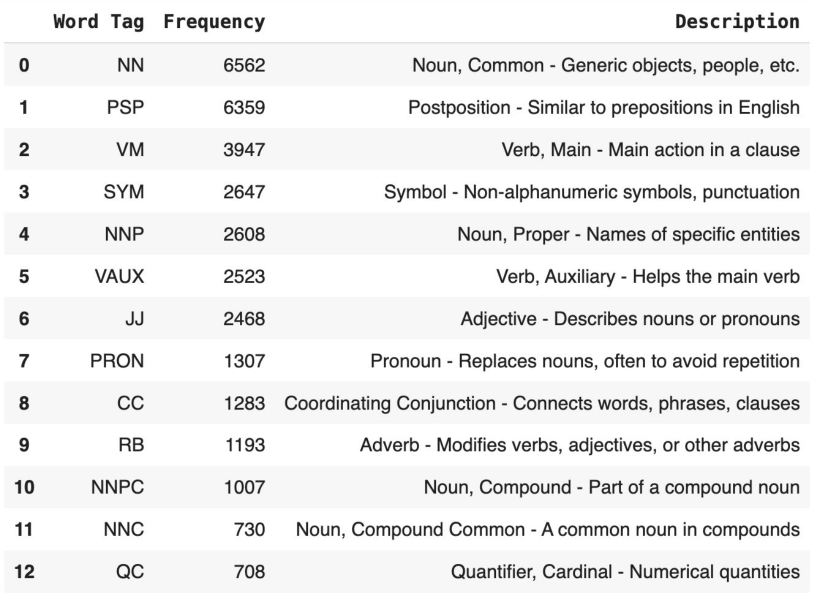
Data and Resources

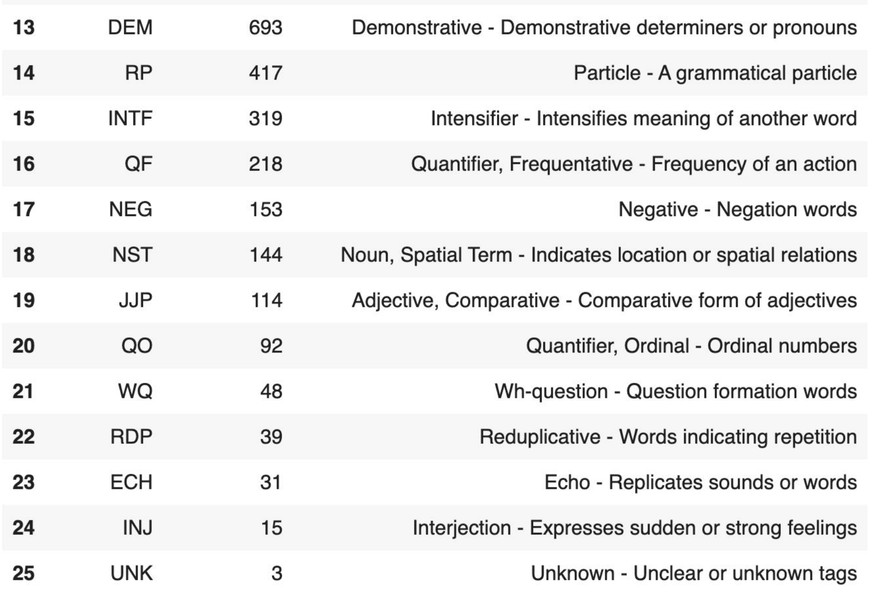
- **annotator -** code that uniquely represents an annotator.
- **task -** ID linking to a specific set of data items or annotation instructions.
- **wordRT -** time taken by the annotator to label the specific word.
- **wordTag -** the label assigned to the word by the annotator.
- **sentRT -** time taken by the annotator to label the entire sentence.
- **sentid -** unique identifier for the sentence passage that is being annotated.
- **word -** specific word that was annotated.
- **text -**the full sentence in which the annotated word appears.
- **freq -** frequency of the word's occurrence within the dataset.
- **length -** length of the word in characters.
- **gold -** the gold standard label

Data and Resources



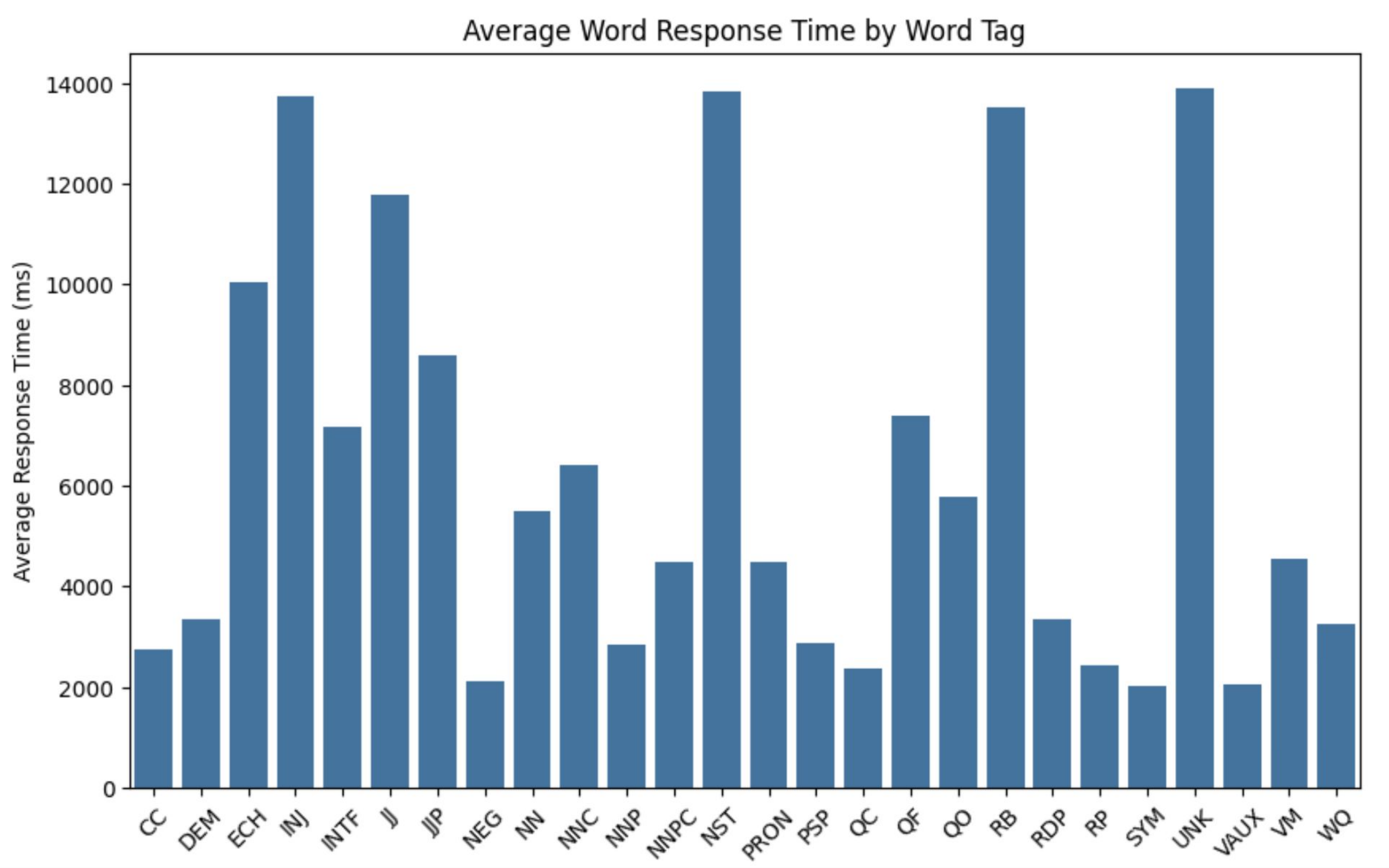
```
Number of unique annotators: 4
Number of unique word tags: 26
Number of columns in the dataset: 11
Number of rows in the dataset: 35628
Number of unique sentences (based on sentId): 500
```



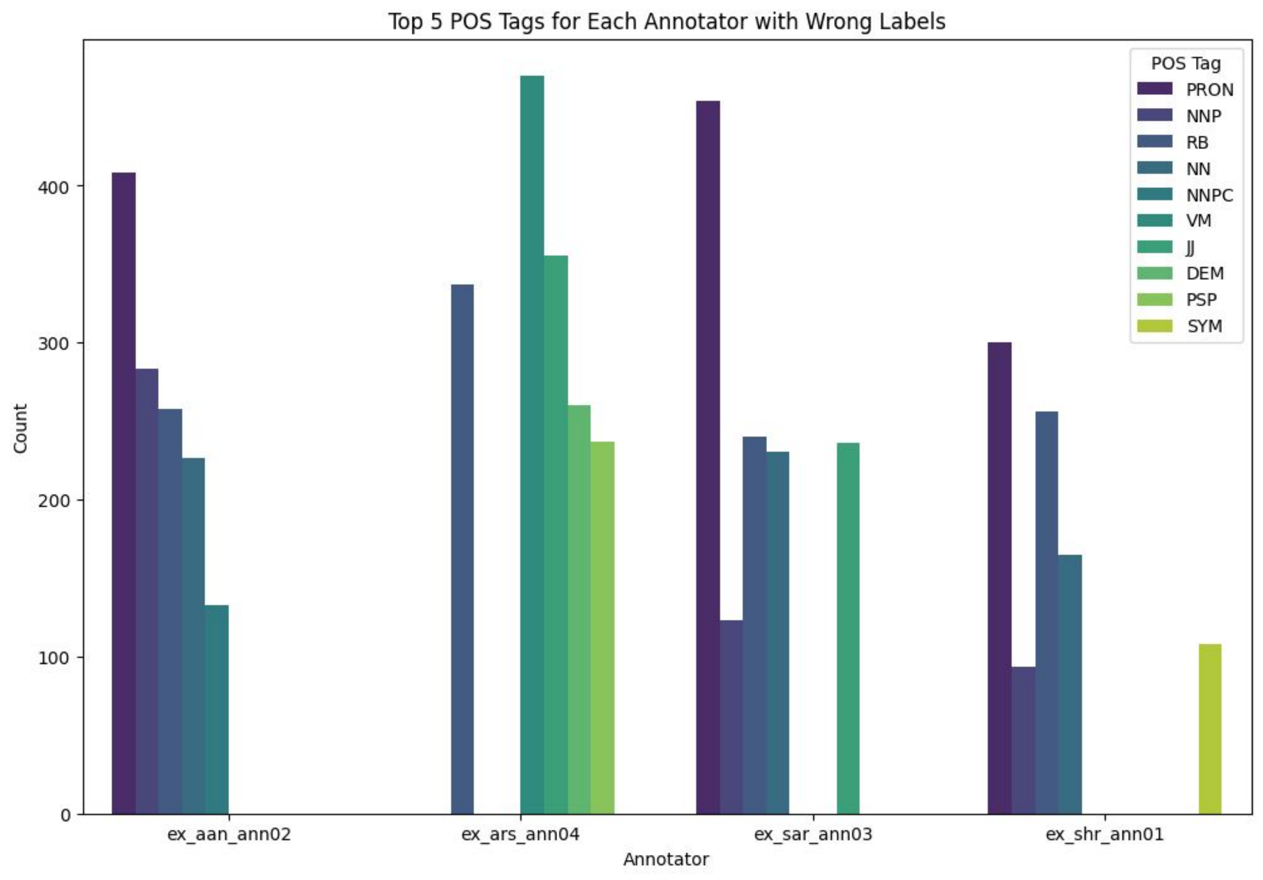


	Word Tag	Frequency	Description
0	NN	6562	Noun, Common - Generic objects, people, etc.
1	PSP	6359	Postposition - Similar to prepositions in English
2	VM	3947	Verb, Main - Main action in a clause
3	SYM	2647	Symbol - Non-alphanumeric symbols, punctuation
4	NNP	2608	Noun, Proper - Names of specific entities
5	VAUX	2523	Verb, Auxiliary - Helps the main verb
6	JJ	2468	Adjective - Describes nouns or pronouns
7	PRON	1307	Pronoun - Replaces nouns, often to avoid repetition
8	CC	1283	Coordinating Conjunction - Connects words, phrases, clauses
9	RB	1193	Adverb - Modifies verbs, adjectives, or other adverbs
10	NNPC	1007	Noun, Compound - Part of a compound noun
11	NNC	730	Noun, Compound Common - A common noun in compounds
12	QC	708	Quantifier, Cardinal - Numerical quantities
13	DEM	693	Demonstrative - Demonstrative determiners or pronouns
14	RP	417	Particle - A grammatical particle
15	INTF	319	Intensifier - Intensifies meaning of another word
16	QF	218	Quantifier, Frequentative - Frequency of an action
17	NEG	153	Negative - Negation words
18	NST	144	Noun, Spatial Term - Indicates location or spatial relations
19	JJP	114	Adjective, Comparative - Comparative form of adjectives
20	QO	92	Quantifier, Ordinal - Ordinal numbers
21	WQ	48	Wh-question - Question formation words
22	RDP	39	Reduplicative - Words indicating repetition
23	ECH	31	Echo - Replicates sounds or words
24	INJ	15	Interjection - Expresses sudden or strong feelings
25	UNK	3	Unknown - Unclear or unknown tags

Data Visualization



Data Visualization



Experiment

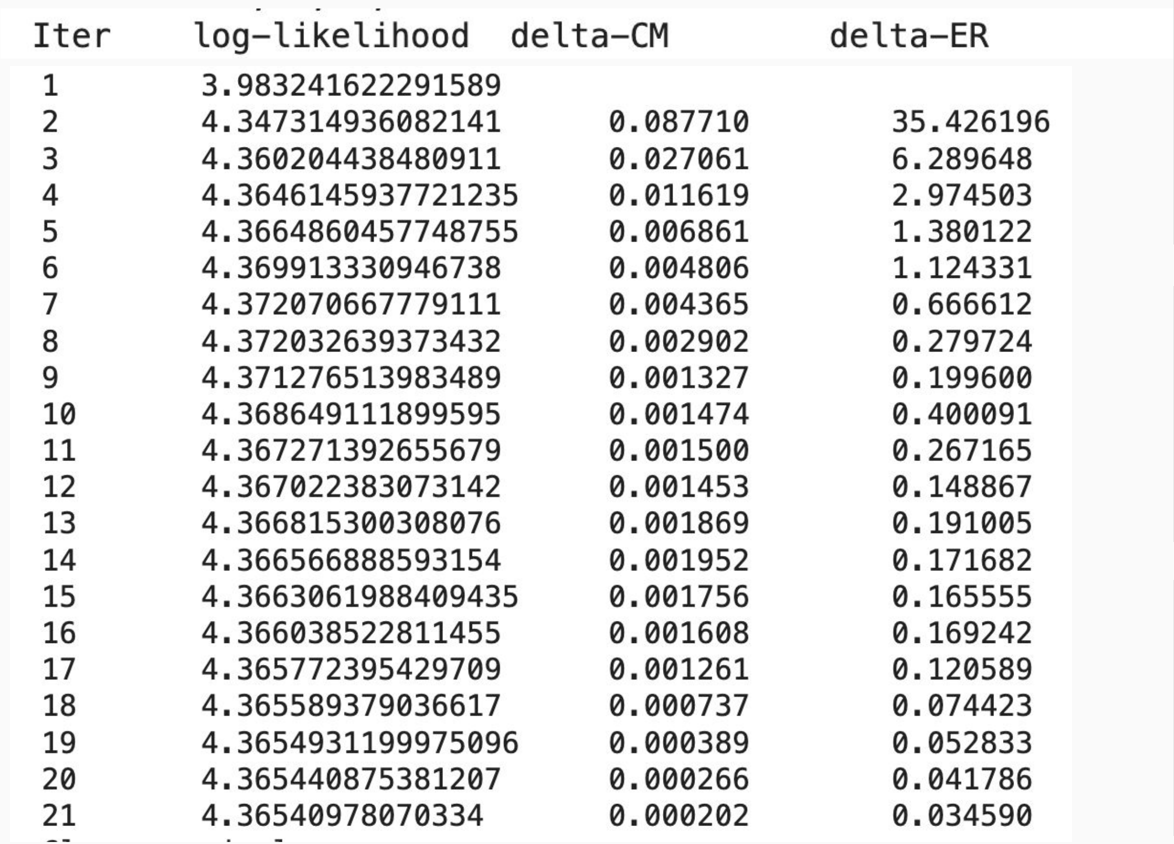
Number of items: 35628

Number of observers: 4

Number of classes: 26

- **Trial 1**
  - The initial parameters were taken as uniform.
  - The error rate is 1/4 for each annotator.
  - The estimated true labels each have a probability of 1/26.
- **Trial 2**
  - As discussed during the presentation, I also tried to set the initial marginal probabilities (i.e. true labels) based on the relative frequencies of tags per word.
  - The initial error rates are based on how the annotators' labels compare to the initial estimated true classes of each word.

Results for Trial 2 (20 iterations)



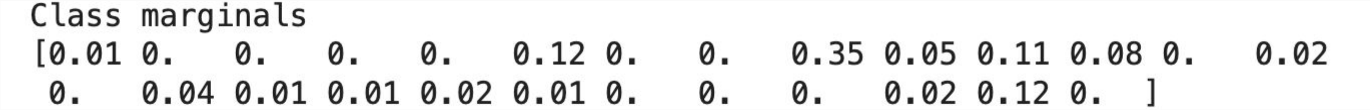
```
 Iter     log-likelihood   delta-CM          delta-ER
  1        3.983241622291589
  2        4.347314936082141      0.087710        35.426196
  3        4.360204438480911      0.027061        6.289648
  4        4.3646145937721235     0.011619        2.974503
  5        4.3664860457748755     0.006861        1.380122
  6        4.369913330946738      0.004806        1.124331
  7        4.372070667779111      0.004365        0.666612
  8        4.372032639373432      0.002902        0.279724
  9        4.371276513983489      0.001327        0.199600
  10       4.368649111899595      0.001474        0.400091
  11       4.367271392655679      0.001500        0.267165
  12       4.367022383073142      0.001453        0.148867
  13       4.366815300308076      0.001869        0.191005
  14       4.366566888593154      0.001952        0.171682
  15       4.3663061988409435     0.001756        0.165555
  16       4.366038522811455      0.001608        0.169242
  17       4.365772395429709      0.001261        0.120589
  18       4.365589379036617      0.000737        0.074423
  19       4.3654931199975096     0.000389        0.052833
  20       4.365440875381207      0.000266        0.041786
  21       4.36540978070334       0.000202        0.034590
```

The log-likelihood is a measure of how well the model parameters fit the observed data. As the EM algorithm progresses, this value should ideally increase (indicating better fit) or stabilize.

delta-CM shows the change in the class marginal probabilities from the previous iteration. These values should ideally decrease over iterations as the algorithm converges.

delta-ER shows the change in error rates from one iteration to the next. It should decrease or stabilize as the model parameters converge to their optimal values.

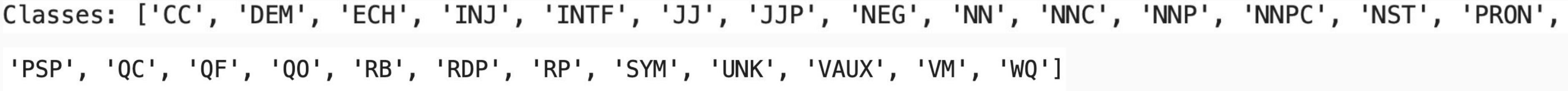
Results for Trial 2 (20 iterations)



```
Class marginals
[0.01 0.   0.   0.   0.   0.12 0.   0.   0.35 0.05 0.11 0.08 0.   0.02
 0.   0.04 0.01 0.01 0.02 0.01 0.   0.   0.   0.02 0.12 0.  ]
```

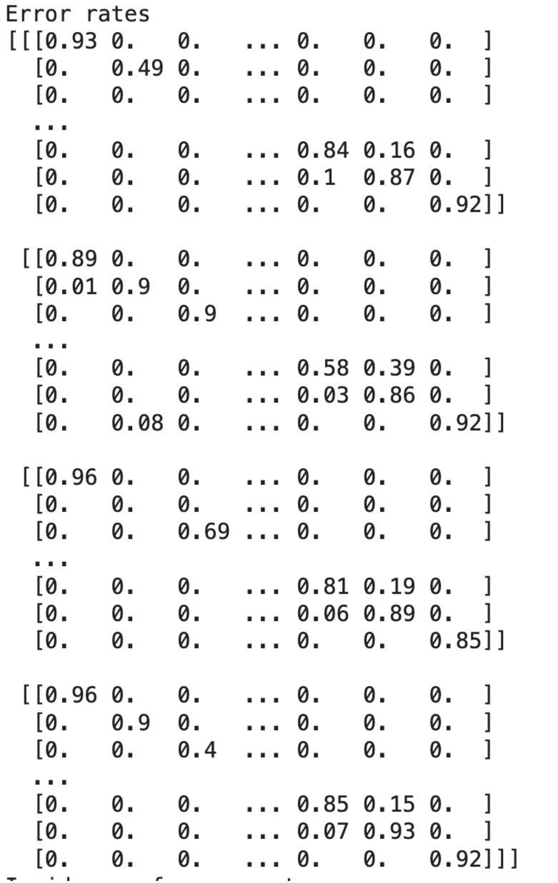
Each value in the array corresponds to the probability that a randomly selected data point (e.g., a tag in our context) belongs to a specific class. These probabilities are normalized so that they sum up to 1 across all classes.

Each index in the array corresponds to a different class (tag type as shown below) that was identified in the data. The value at each index represents the estimated proportion of the total data that belongs to that class.



```
Classes: ['CC', 'DEM', 'ECH', 'INJ', 'INTF', 'JJ', 'JJP', 'NEG', 'NN', 'NNC', 'NNP', 'NNPC', 'NST', 'PRON',
'PSP', 'QC', 'QF', 'QO', 'RB', 'RDP', 'RP', 'SYM', 'UNK', 'VAUX', 'VM', 'WQ']
```

Results for Trial 2 (20 iterations)



```
Error rates
[[[0.93 0.   0.   ... 0.   0.   0.  ]
  [0.   0.49 0.   ... 0.   0.   0.  ]
  [0.   0.   0.   ... 0.   0.   0.  ]
  ...
  [0.   0.   0.   ... 0.84 0.16 0.  ]
  [0.   0.   0.   ... 0.1  0.87 0.  ]
  [0.   0.   0.   ... 0.   0.   0.92]]

 [[0.89 0.   0.   ... 0.   0.   0.  ]
  [0.01 0.9  0.   ... 0.   0.   0.  ]
  [0.   0.   0.9  ... 0.   0.   0.  ]
  ...
  [0.   0.   0.   ... 0.58 0.39 0.  ]
  [0.   0.   0.   ... 0.03 0.86 0.  ]
  [0.   0.08 0.   ... 0.   0.   0.92]]

 [[0.96 0.   0.   ... 0.   0.   0.  ]
  [0.   0.   0.   ... 0.   0.   0.  ]
  [0.   0.   0.69 ... 0.   0.   0.  ]
  ...
  [0.   0.   0.   ... 0.81 0.19 0.  ]
  [0.   0.   0.   ... 0.06 0.89 0.  ]
  [0.   0.   0.   ... 0.   0.   0.85]]

 [[0.96 0.   0.   ... 0.   0.   0.  ]
  [0.   0.9  0.   ... 0.   0.   0.  ]
  [0.   0.   0.4  ... 0.   0.   0.  ]
  ...
  [0.   0.   0.   ... 0.85 0.15 0.  ]
  [0.   0.   0.   ... 0.07 0.93 0.  ]
  [0.   0.   0.   ... 0.   0.   0.92]]]
```

The Error rates matrix gives a detailed view of the estimated error rates for each observer (annotator) when assigning classes (tags) to the patients (words). This matrix is a 3-dimensional array where the dimensions correspond to:

- Observers (Annotators): Each "slice" or 2D matrix within the 3D matrix corresponds to one annotator.
- True labels: Each row in these 2D matrices represents the true class of the item being tagged.
- Assigned classes: Each column in these 2D matrices represents the class assigned by the observer.

The values along the diagonal of each 2D slice indicate the probability that the observer correctly identifies the true class. Higher values signify better accuracy. Values off the diagonal represent misclassification rates.

Overall, it shows how well each annotator performs across different classes and where their weaknesses lie in terms of misclassifications.

Discussion

- David-Skene algorithm provides a way to estimate the reliability of each annotator, which can be invaluable in contexts where annotations are subjective and annotators vary in expertise.
- Initial estimates and convergence of the EM algorithm are sensitive to starting values and model specifics.
- The convergence of Dawid-Skene would often be slow since it is computationally intensive, especially with a large number of annotators and classes, which can make it less scalable.
- While it's excellent for categorical data, its adaptation to continuous or highly dimensional data requires modifications that can complicate its application.
- The algorithm could be adapted to select the most informative items for manual annotation in an active learning setup, where the goal is to maximize model performance with minimal labeled data.

References

[1] Dawid, A. P., & Skene, A. M. (1979). Maximum likelihood estimation of observer error-rates using the EM algorithm. Journal of the Royal Statistical Society: Series C (Applied Statistics), 28(1), 20-28.

[2] Sabou, M., Bontcheva, K., Derczynski, L., & Scharl, A. (2014, May). Corpus annotation through crowdsourcing: Towards best practice guidelines. In LREC (pp. 859-866).

[3] https://github.com/dallascard/dawid_skene/tree/master

[4] Li, H., Yu, B., & Zhou, D. (2013, June). Error rate analysis of labeling by crowdsourcing. In ICML Workshop: Machine Learning Meets Crowdsourcing. Atalanta, Georgia, USA.