

A Bayesian Hierarchical Model for Comparing Average F1 Scores

Dell Zhang¹, Jun Wang², Xiaoxue Zhao², **Xiaoling Wang**³

¹Birkbeck, University of London, UK

²University College London, UK

³East China Normal University, China

17 Nov 2015

Outline

- 1 Background
 - Introduction
 - Problem Statement
 - Related Work
- 2 Our Approach
 - Models
 - Experiments
- 3 Summary

Outline

- 1 Background
 - Introduction
 - Problem Statement
 - Related Work
- 2 Our Approach
 - Models
 - Experiments
- 3 Summary

Introduction - Text Classification

- Definition:
 - Automatic text classification is a fundamental technique in information retrieval
- Applications:
 - Topic categorisation, spam filtering, sentiment analysis, message routing...
- Performance measure:
 - F_1 Score

Introduction - F_1 Score

- Definition:
 - The harmonic mean of **precision**(P) and **recall**(R).
- Two methods:
 - **Micro-averaged F_1 score** (MiF_1):
Gives equal weight to each classification decision
 - **Macro-averaged F_1 score** (MaF_1):
Gives equal weight to each class
- Limitations:
 - Does not tell us how reliable it is on unseen data.

Outline

- 1 Background
 - Introduction
 - **Problem Statement**
 - Related Work
- 2 Our Approach
 - Models
 - Experiments
- 3 Summary

Problem Statement

- Goal:
 - Assess the uncertainty of a classifier's performance as measured by miF_1 and maF_1

Outline

- 1 Background
 - Introduction
 - Problem Statement
 - Related Work
- 2 Our Approach
 - Models
 - Experiments
- 3 Summary

Related Work - Frequentist Performance Comparison

- NHST
 - Y. Yang and X. Liu, “A re-examination of text categorization methods”, in *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*
- use s -test to compare two classifiers' accuracy scores
- use t -test to compare two classifiers' performance measures in the form of proportions

Related Work - Frequentist Performance Comparison

- Deficiencies of NHST
 - Can only reject the null hypothesis, can never accept the null hypothesis.
 - Will reject the null hypothesis even the performance difference is very close to zero.
 - Can only be compared on the category level but not on the document level for complex performance measures

Related Work - Bayes Factor

1 Bayes Factor

- D. Barber, “Are two classifiers performing equally? a treatment using Bayesian hypothesis testing,” IDIAP, Tech. Rep., 2004.
- —, [Bayesian Reasoning and Machine Learning](#). Cambridge University Press, 2012.

2 Deficiencies of Bayes Factor

- Sensitive to the choice of prior distribution in the alternative model.
- The null hypothesis can be strongly preferred even with very few data and very large uncertainty in the estimate of the performance difference

Related Work - Bayesian Estimation

- 1 Bayesian Estimation
 - C. Goutte and E. Gaussier, “A probabilistic interpretation of precision, recall and F-score, with implication for evaluation,” in *Proceedings of the 27th European Conference on IR Research (ECIR)*,
- 2 It is restricted to a single F_1 score for binary classification with two classes only.
- 3 In contrast, our proposed approach opens up many possibilities for adaptation or extension.

Outline

- 1 Background
 - Introduction
 - Problem Statement
 - Related Work
- 2 Our Approach
 - Models
 - Experiments
- 3 Summary

Models - True Classification

- Multi-class single-label classification
 - M different classes
 - N labelled test documents
- Documents' true class labels y_i are i.i.d.
 - $\mu = (\mu_1, \dots, \mu_M)$: the probabilities that a test document truly belongs to each class
 - $\mathbf{n} = (n_1, \dots, n_M)$: the true size of each class
- n follows a multinomial distribution with parameter μ , where $\sum_{j=1}^M n_j = N$.

Models - True Classification

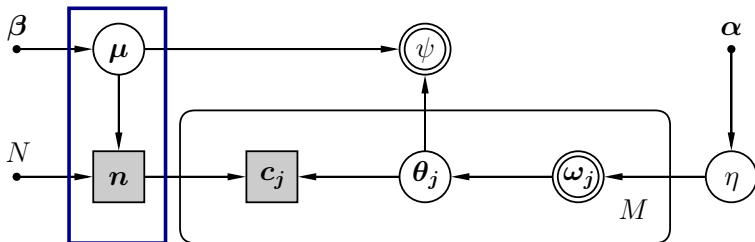


Figure: The probabilistic graphical model for estimating the uncertainty of average F_1 scores.

Models - Predicted Classification

- Class level
 - $\theta_j = (\theta_1, \dots, \theta_M)$: the probabilities that a document of true class label j is classified into different classes.
 - $\omega_j = (\omega_1, \dots, \omega_M)$: the parameters of the θ_j 's Dirichlet prior.
- Model level
 - η : the overall tendency of making correct predictions
- $w_{jk} = \begin{cases} \eta & \text{if } k = j \\ (1 - \eta)/(M - 1) & \text{if } k \neq j \text{ for } k = 1, \dots, M \end{cases}$

Models - Predicted Classification

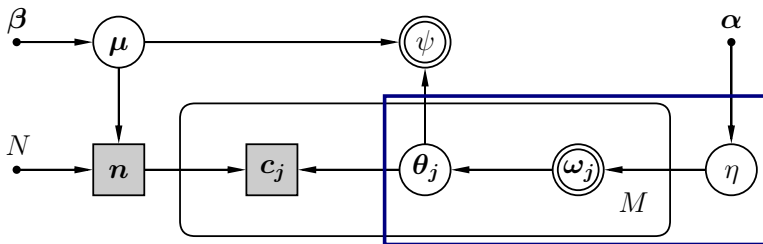
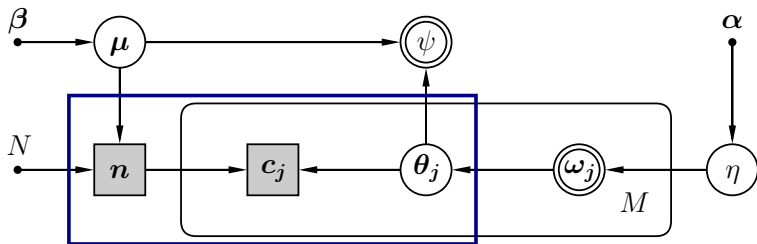


Figure: The probabilistic graphical model for estimating the uncertainty of average F_1 scores.

Models - Performance

- Confusion matrix C presents the classification results.
 - C is a $M \times M$ matrix.
 - c_{jk} represents the number of documents with true class label j but predicted class label k .
 - c_j follows a multinomial distribution with parameter θ_j , where $\sum_{k=1}^M c_{jk} = n_j$.



Models - Performance

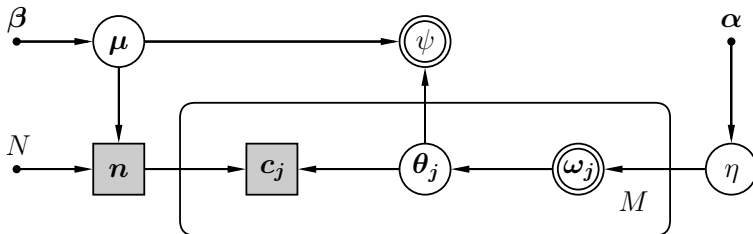
- μ presents the true classification of documents.
- ω presents the predicted classification.
- Treat the performance measure (either miF_1 or maF_1) as a random variable ψ , which is a function of μ and ω . For example, in miF_1

$$\text{Precision} = \frac{\sum_{j=1}^M tp_j}{\sum_{j=1}^M tp_j + fp_j} = \sum_{j=1}^M \mu_j \theta_{jj}$$

$$\text{Recall} = \frac{\sum_{j=1}^M tp_j}{\sum_{j=1}^M tp_j + fn_j} = \sum_{j=1}^M \mu_j \theta_{jj}$$

- In multi-class single-label, $miF1 = \text{Precision} = \text{Recall}$.

Models - Performance



- For two models A and B, the difference of the overall performance is represented by δ , where $\delta = \psi_A - \psi_B$.
- Estimate the uncertainty difference of two models by examining the posterior probability distribution of δ .

Outline

- 1 Background
 - Introduction
 - Problem Statement
 - Related Work
- 2 Our Approach
 - Models
 - Experiments
- 3 Summary

Experiments - Dataset

- A standard benchmark dataset for text classification, 20newsgroups¹.
- 60% subset for training
- 40% subset for testing
- Filtered by stripping newsgroup-related metadata

¹<http://qwone.com/~jason/20Newsgroups/> 

Experiments - Classifiers

Classification algorithms:

- Naive Bayse (NB)
 - Bernoulli event model (NB_{Bern})
 - Multinomial event model (NB_{Mult})
- linear Support Vector Machine (SVM)
 - $L1$ penalty (SVM_{L1})
 - $L2$ penalty (SVM_{L2})

Implementation of these algorithms:

- Python library `scikit-learn`

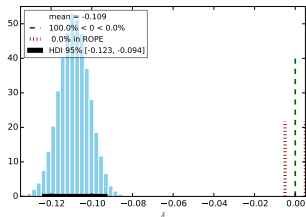
Experiments - Results

True class label i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	129	4	0	2	16	2	3	6	9	4	4	4	2	6	9	64	7	14	17	25
1	1	12	1	15	34	30	5	1	5	1	10	21	9	9	13	1	0	1	0	0
2	5	67	38	93	54	48	11	3	7	0	1	23	10	14	14	2	1	0	3	0
3	0	22	10	13	60	10	18	1	2	0	2	9	23	3	1	0	0	0	0	0
4	0	25	1	34	15	7	17	6	7	3	1	5	20	7	17	0	0	0	0	0
5	0	80	2	11	21	15	8	2	3	0	0	11	3	8	5	0	1	0	0	0
6	1	10	0	30	34	5	15	10	10	2	3	2	7	9	9	1	1	2	0	0
7	1	3	0	1	29	0	14	14	48	3	0	3	14	3	15	2	3	2	9	2
8	7	1	0	0	22	0	12	30	15	0	5	4	12	4	7	1	14	6	5	1
9	4	1	0	1	21	0	5	4	9	16	16	4	3	16	6	3	10	1	12	3
10	4	1	1	0	11	0	7	5	9	11	12	3	2	4	4	2	3	4	5	1
11	10	7	0	6	33	4	3	3	14	3	2	15	11	5	14	1	8	7	8	3
12	2	19	0	23	32	7	17	14	11	2	1	26	10	22	20	0	3	1	0	0
13	5	6	0	2	20	0	8	12	12	1	1	1	6	10	9	12	6	6	4	3
14	5	9	0	1	21	4	8	9	13	2	2	2	6	11	12	3	7	7	12	0
15	28	4	0	0	16	2	0	0	5	1	1	2	0	5	2	1	2	9	5	35
16	15	1	0	2	19	1	3	9	19	1	4	15	4	7	8	6	10	15	23	23
17	15	4	0	4	9	2	2	3	12	6	1	7	3	1	5	13	5	14	7	0
18	23	2	0	0	8	2	2	8	15	2	1	8	10	10	63	30	10	9	0	0
19	42	2	0	0	11	2	1	4	21	5	1	4	2	8	7	58	19	14	8	42
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
predicted class label j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	138	0	1	2	0	3	0	3	4	2	10	4	1	3	11	75	10	16	13	23
1	3	10	8	16	18	29	3	0	6	0	5	12	0	19	3	0	1	0	0	0
2	4	34	10	64	13	28	3	1	5	0	16	13	2	4	8	3	0	1	4	3
3	0	14	23	17	31	3	8	4	0	0	8	5	19	0	1	0	0	0	0	0
4	0	13	8	39	15	3	8	7	1	0	15	7	17	2	2	0	0	0	0	0
5	0	46	11	8	6	15	3	0	1	1	5	10	4	2	2	1	1	1	0	0
6	0	2	1	27	21	0	16	14	6	2	10	1	8	1	7	4	1	1	2	0
7	3	1	2	1	1	0	8	10	10	125	5	11	3	7	3	4	3	7	1	0
8	7	3	1	0	1	2	2	29	16	2	13	1	9	6	4	5	11	3	6	1
9	7	3	1	0	0	4	2	5	11	29	4	2	4	2	4	3	7	6	1	7
10	5	0	0	0	0	1	0	1	2	5	12	0	1	2	4	2	2	2	0	0
11	3	11	4	3	4	1	0	0	3	18	16	4	1	6	4	21	6	7	1	0
12	1	12	7	27	12	1	8	7	11	112	37	19	14	13	2	2	2	2	0	0
13	4	5	1	2	1	0	0	5	4	0	15	0	5	10	17	9	4	6	1	0
14	3	8	1	1	0	1	0	6	3	1	18	4	6	3	10	6	1	7	6	3
15	8	3	0	0	1	1	0	1	1	14	1	1	1	1	1	1	1	1	1	1
16	7	0	0	0	0	1	1	7	2	1	11	10	0	4	9	15	10	0	14	11
17	13	2	0	0	1	0	1	0	3	2	8	3	0	0	2	21	7	10	9	1
18	19	2	0	0	0	0	1	5	3	1	7	5	2	11	8	11	9	11	13	3
19	31	3	1	1	0	0	0	1	3	2	7	4	2	8	6	10	23	9	7	43
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

Comparing maF_1 between NB_{Bern} and NB_{Mult} .

Conclusion:

NB_{Bern} is significantly outperformed by NB_{Mult}

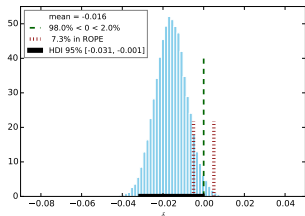


Experiments - Results

19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
14	4	3	2	2	1	5	15	5	4	1	3	5	8	11	47	13	8	4	37
5	23	23	11	3	26	4	9	6	3	2	8	11	1	6	5	1	4	6	3
5	23	37	37	16	11	4	19	1	2	2	2	3	5	9	2	4	1	9	5
2	13	36	25	26	5	16	10	0	1	1	4	33	2	1	0	1	0	5	1
6	9	14	33	25	4	8	18	6	1	2	2	2	5	5	3	2	0	2	1
5	46	33	9	5	25	5	6	3	0	1	3	4	2	7	4	1	3	2	2
0	6	3	19	15	1	25	18	8	4	1	1	12	2	5	3	4	1	1	4
4	8	3	2	6	2	14	25	19	4	1	4	15	2	4	3	7	1	10	7
4	5	2	3	2	0	6	39	25	6	2	0	8	9	6	3	2	4	10	6
4	1	5	3	3	1	6	22	7	25	24	0	5	5	2	9	3	2	6	2
2	3	2	2	4	2	1	12	2	20	25	4	1	3	4	4	0	1	2	2
3	5	6	4	6	4	8	19	2	1	4	27	10	3	8	4	14	3	11	5
6	13	8	25	16	7	16	24	13	4	4	13	20	18	6	6	4	0	5	4
9	6	2	4	0	0	8	23	7	1	5	1	11	25	7	8	6	4	10	4
5	13	3	4	5	1	5	24	4	3	3	1	18	9	25	7	5	2	15	1
25	1	2	2	0	1	3	16	1	2	1	3	2	10	4	25	1	5	3	28
7	3	4	3	2	2	4	22	3	0	11	4	4	9	9	21	9	34	13	3
26	0	3	1	2	0	0	7	5	6	0	6	2	3	3	16	15	25	13	3
10	1	0	2	0	0	2	12	5	3	2	8	3	7	5	4	24	5	13	18
34	4	1	4	1	2	2	12	3	2	2	4	3	8	3	55	21	3	12	75
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
predicted class label \hat{y}																			
145	1	2	1	0	2	2	3	6	11	1	2	10	11	10	47	12	10	12	31
5	22	9	6	25	8	3	2	10	3	6	12	2	8	2	1	4	2	3	1
5	24	33	36	19	14	2	2	19	2	3	4	6	9	1	3	1	5	3	5
1	18	32	24	25	8	13	2	0	7	1	2	33	0	3	1	0	1	2	0
1	5	9	38	25	1	8	7	15	3	4	19	2	6	2	1	0	2	1	2
0	48	33	9	5	25	2	0	1	6	0	6	4	2	6	3	1	2	2	0
0	5	7	15	13	2	20	8	4	10	1	2	7	0	5	2	2	3	0	2
6	2	2	4	5	2	10	25	15	30	2	2	16	5	7	1	8	4	7	3
2	3	2	2	4	0	4	23	20	18	1	3	11	7	8	8	5	6	7	4
3	1	0	5	2	2	5	5	4	31	26	2	2	2	5	3	6	1	1	8
1	3	3	1	1	1	2	4	3	24	25	1	1	2	1	3	2	2	2	4
5	3	6	2	5	4	7	3	4	20	3	27	11	4	5	4	14	7	9	3
7	15	7	21	17	9	17	11	7	15	1	14	20	15	12	4	2	3	2	0
5	9	3	2	0	1	4	7	7	15	5	1	8	25	5	14	5	4	9	4
4	10	5	2	4	3	3	6	4	18	4	2	14	25	11	11	3	7	3	3
22	2	3	3	0	1	1	0	18	1	1	1	5	2	31	0	1	2	18	8
4	3	5	2	1	1	3	4	6	17	0	10	3	19	0	11	25	10	22	12
24	1	1	4	2	0	0	4	7	12	1	4	3	2	2	13	8	25	14	5
14	1	0	1	0	1	1	6	3	12	3	7	2	4	9	4	25	9	13	9
31	4	3	3	3	2	2	3	2	9	4	2	0	11	5	27	17	8	9	26
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
predicted class label \hat{y}																			

Comparing maF_1 between SVM_{L1} and SVM_{L2} .

Conclusion:
 SVM_{L1} is only slightly outperformed by SVM_{L2}



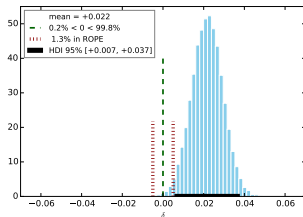
Experiments - Results

predicted class label \hat{y}	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
true class label y	139	0	1	2	0	3	0	3	4	2	10	4	1	3	11	75	10	16	13	23
3	27	8	16	18	29	3	0	6	0	5	12	0	1	9	3	0	1	0	0	
4	34	18	64	13	28	3	1	5	0	16	13	2	4	8	3	0	1	4	3	
0	14	23	27	31	3	8	4	0	0	8	5	19	0	1	0	0	0	0	0	
0	13	8	39	25	3	8	7	1	0	15	7	17	2	2	0	0	0	0	0	
0	46	11	8	6	25	3	0	1	1	5	10	4	2	2	1	1	1	0	0	
0	2	1	27	21	0	25	14	6	2	10	1	8	1	7	4	1	1	2	0	
3	1	2	1	1	0	8	25	30	1	25	5	11	3	7	3	4	3	7	1	
7	3	1	1	2	2	3	29	25	2	13	1	9	6	4	5	11	3	6	1	
7	3	1	0	0	0	4	2	5	11	29	4	2	4	3	7	6	1	7	0	
5	0	0	0	0	1	0	1	2	5	27	2	0	1	2	4	2	2	2	0	
3	11	4	3	4	1	0	0	3	3	16	25	4	1	6	4	21	6	7	1	
1	12	7	27	12	1	8	7	11	1	12	37	29	14	13	2	0	2	2	0	
4	5	1	2	1	0	0	5	4	0	15	0	5	20	8	17	9	4	6	1	
3	8	1	1	0	1	0	6	3	1	18	4	6	5	20	6	4	7	8	3	
8	3	0	0	1	1	0	1	1	1	14	1	1	1	2	25	2	0	3	5	
7	0	0	0	0	1	1	7	2	1	11	10	0	4	9	15	25	9	14	11	
13	2	0	1	0	1	0	3	3	2	8	3	0	0	2	21	7	20	9	1	
19	2	0	0	0	0	1	5	3	1	7	5	2	11	8	11	20	11	13	3	
31	3	1	1	0	0	0	1	3	2	7	4	2	8	5	10	23	9	7	43	
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
predicted class label \hat{y}	145	1	2	1	0	2	2	3	6	11	1	2	10	11	10	47	12	10	12	31
5	22	9	6	25	8	3	2	10	3	6	12	2	8	2	1	4	2	3	0	
5	24	23	36	19	14	2	2	19	2	3	4	6	9	1	3	1	5	3	0	
1	18	32	14	25	8	13	2	0	7	1	2	33	0	3	1	0	1	2	0	
1	5	9	38	24	1	8	7	15	3	4	19	2	6	2	1	0	2	1	0	
0	48	33	9	5	25	2	0	1	6	0	6	4	2	6	3	1	2	0	2	
0	5	7	15	13	2	20	8	4	10	1	2	7	0	5	2	2	3	0	2	
6	2	2	4	5	2	10	25	15	30	2	2	16	5	7	1	8	4	7	3	
2	3	2	2	4	0	4	23	20	18	1	3	11	7	8	8	5	6	7	4	
3	1	0	5	2	2	5	5	4	11	26	2	2	5	3	6	1	1	8	0	
1	3	3	1	1	1	2	4	3	24	25	1	1	2	1	3	2	2	2	4	
5	3	6	2	5	4	7	3	4	20	3	27	11	4	5	4	14	7	9	3	
7	15	7	21	17	9	17	11	7	15	1	14	29	15	12	4	2	3	2	0	
5	9	3	2	0	1	4	7	7	15	5	1	8	25	5	14	5	4	9	4	
4	10	5	2	4	3	3	6	4	18	4	2	14	5	2	2	11	5	3	7	
2	2	3	3	0	1	1	0	18	1	1	1	5	2	2	3	0	1	2	18	
4	3	5	2	1	1	3	4	6	17	0	10	3	10	0	1	12	10	22	12	
24	1	1	4	2	0	0	4	7	12	1	4	3	2	2	13	8	25	14	5	
14	1	0	1	0	1	1	6	3	12	3	7	2	4	9	4	9	9	9	9	
31	4	3	3	3	2	2	3	2	9	4	2	0	11	5	27	17	8	9	26	
predicted class label \hat{y}	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

Comparing maF_1 between NB_{Mult} and SVM_{L2} .

Conclusion:

NB_{Mult} works a lot better than SVM_{L2}



Summary

- The main contribution of this paper is a **Bayesian estimation approach to assessing the uncertainty of average F_1 scores** in multi-class text classification.
- We make *interval estimation* instead of simplistic *point estimation* of a text classifier's future performance on unseen data.
- Extension
 - To be used in the multi-class multi-label classification.
 - To compare classifiers on any type of data, e.g., images.