1.5em 0pt

Effective Unsupervised Author Disambiguation with Relative Frequencies

Group 6

Data Mining Project

Outline

Team Members

Abstract

Introduction

Method description

Evaluation

Graphical Data and Observations

Related Work and Acknowledgement

Team Members

The project was done by Group 6 under the guidance of Prof. Manish Singh and Mr. Rohan Banerjee.
Our group comprises of :

1. ADIL TANVEER - ES17BTECH11026
2. SOURADEEP CHATTERJEE - ES17BTECH11028
3. ANURAG PATIL - CS17BTECH11004
4. GAJANAN SHETKAR - CS17BTECH11016
5. DESU SURYA SAI TEJA - CS17BTECH11048

Abstract

The project aims to address the problem of author name homonymy in the Aminar DBLP Dataset.
We use an efficient, simple and straightforward solution to this problem using **agglomerative clustering** methods based on feature overlap.

Introduction

Documents have authors. Author name is nothing but a string of characters that is given with the documents. Using these strings gives rise to two problems:

- Name Synonymy: One author is referred to by two strings.
- Name Homonymy: More than one authors are referred to by the same string.

Disambiguation of both these problems have two entirely different approaches. Our project focuses on the problem of Homonymy Disambiguation.

This, in general is a clustering problem over author names. Each cluster is considered an author. More formally:

1. For each collection, there is a set N of names $name \in N$
2. For each name $name$, there is a set $\mathfrak{C}$ (also referred to as a clustering) of authors $C \in \mathfrak{C}$ (also referred to as a cluster) and a set of mentions $x \in X$
3. Each author $C \subset X$ is a set of mentions $x \in C$
4. For each mention x, there is a bag of features $f \in F(x)$, each with a frequency #(f,x) of occurrence with x

The task of author disambiguation is to suggest a system clustering C_{sys} that is as close to the correct clustering C_{cor} as possible. In the evaluation case, we have both C_{sys} and C_{cor} present and calculate some evaluation score $eval(C_{sys}, C_{cor})$. Our problem is clearly defined in the figure below.

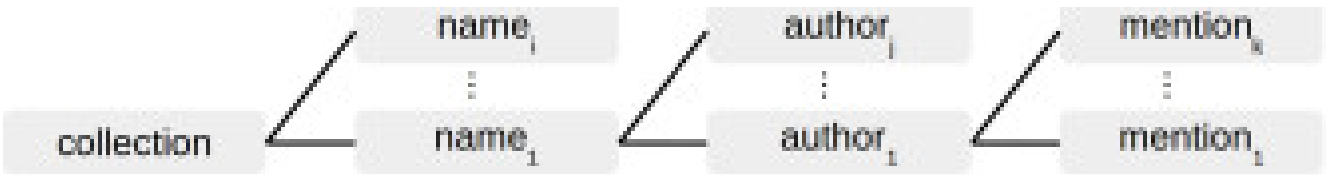


Figure 1: Author disambiguation problem structure

Method description

Our method disambiguates one name at a time. It clusters all mentions of that name based on features extracted from the collection.

Features

The focus of our project was not to investigate specific features but to develop a method that can provide satisfying results independent of the exact set of features and feature-types.
The following are the features considered:

1. Co-authors
2. Field of Study
3. Years
4. References
5. Reference Authors
6. Languages
7. Publishers
8. Keywords

Agglomerative clustering

We use agglomerative clustering technique. Which means that we start with the initial state where each mention x is in its own cluster $C = \{x\}$. Then, pairs $(C, \dot{C})$ of clusters are merged. If no stopping criterion is applied, this will ultimately result in a state where all mentions are in the same cluster $C = X$.
For this reason, we need to compute the score $score(C, \dot{C})$ of a pair $(C, \dot{C})$ of clusters to be merged.
Furthermore, we deploy a quality limit l, that tells us whether the score can be considered good or not. In our approach, $score(C, \dot{C})$ is not dependent on the score of any other pair of clusters. Neither is the quality limit.

This means that in each iteration of the clustering process, we merge all pairs (C, $\dot{C}$), such that

1. $\neg\exists\ddot{C} \in \mathfrak{C} : score(C,\ddot{C}) > score(C, \dot{C}) \bigwedge \neg\exists\dddot{C} \in \mathfrak{C} : score(\dddot{C}, C) > score(C,\dot{C})$ and at the same time
2. $score(C,\dot{C}) > 1$.

In other words, we evaluate all disjoint pairs $(C,\dot{C}) \in \mathfrak{C} \times \mathfrak{C}$ for each of these pairs, we check whether (1) and (2) holds. If yes, then the pair is saved for merging.
This process converges if there are no pairs saved for merging.

Algorithm

Algorithm 1: Agglomerative clustering

Input : *name* with X and C as well as N,#(f), l

while $|C| > 1$ **do**

  $score \leftarrow NULL$;

  foreach $(C,\dot{C}) \in \mathcal{C} \times \mathcal{C}$ **do**

    $score(C, \dot{C}) \leftarrow p(C|\dot{C})$

  end

  merges $= \{\}$

  foreach $(C,\dot{C}) \in \mathcal{C} \times \mathcal{C}$ **do**

    if $\neg\exists\ddot{C} \in \mathcal{C} : score(C, \ddot{C}) > score(C, \dot{C})$ & $\neg\exists\dddot{C} \in \mathcal{C} : score(\dddot{C}, \dot{C}) > score(C, \dot{C})$ & $score(C, \dot{C}) > l$ **then**

      merges $\leftarrow merges \cup (C, \dot{C})$

    end

  end

  if $|merges| = 0$ **then**

    break;

  end

  else

    foreach $(C,\dot{C}) \in merges$ **do**

      merge$(C, \dot{C})$

    end

  end

Variations

- We have considered the clustering to be fully enclosed in the current name block. While still treating each name as a separate clustering problem, we can say that X is not only the set of mentions for the current name, but for the entire collection.
- We only use $p(C|\dot{C})$ in $score(C, \dot{C})$.Intuitively, $p(C)$ favours large clusters for merging, which does not make sense as it introduces a tendency that reinforces itself.
- It does not matter that $p(C|\dot{C}) \neq p(\dot{C}|C)$, as merging is symmetric and the clustering procedure we described will simply use $\max(p(C|\dot{C}), p(\dot{C}, C))$ unless there is a third cluster that matches even better (or the limit is not met).

Evaluation

We used the DBLP Aminar dataset as a source of metadata. In the preprocessing step, we removed stopwords and punctuations. After that, we made objects for each paper with all features as attributes and stored the details of document there. For keyword extraction ,we prepared a Bag of Words using the abstracts of all documments combined. After that, we removed the most frequently and least frequently occurring words from our bag to prevent bias.

For each name in the clustering process, we record the following information:

1. Precision of the system clustering of that name
2. Recall of the system clustering of that name
3. Number of clusters in the correct system clustering of that name

Evaluation measures

$$pairs(\mathfrak{C}) = \cup_{C \in \mathfrak{C}} \{\{x, \dot{x}\} | x, \dot{x} \in C \wedge x \neq \dot{x}\}$$

$$\mathrm{P}_{pairF1} = \frac{pairs(\mathfrak{C}_{cor}) \cap pairs(\mathfrak{C}_{sys})}{pairs(\mathfrak{C}_{sys})}$$

$$\mathrm{R}_{pairF1} = \frac{pairs(\mathfrak{C}_{cor}) \cap pairs(\mathfrak{C}_{sys})}{pairs(\mathfrak{C}_{cor})}$$

$$\mathrm{C}_{sys}(x) = C \in \mathfrak{C}_{sys} \,:\, x \in C$$

$$\mathrm{P}_{bCube} = \frac{1}{|X|} \cdot \sum_{x \in X} \frac{\left|C_{sys}(x) \cap C_{cor}(x))\right|}{\left|C_{sys}(x)\right|}$$

$$\mathrm{R}_{bCube} = \frac{1}{|X|} \cdot \sum_{x \in X} \frac{\left|C_{sys}(x) \cap C_{cor}(x))\right|}{\left|C_{cor}(x)\right|}$$

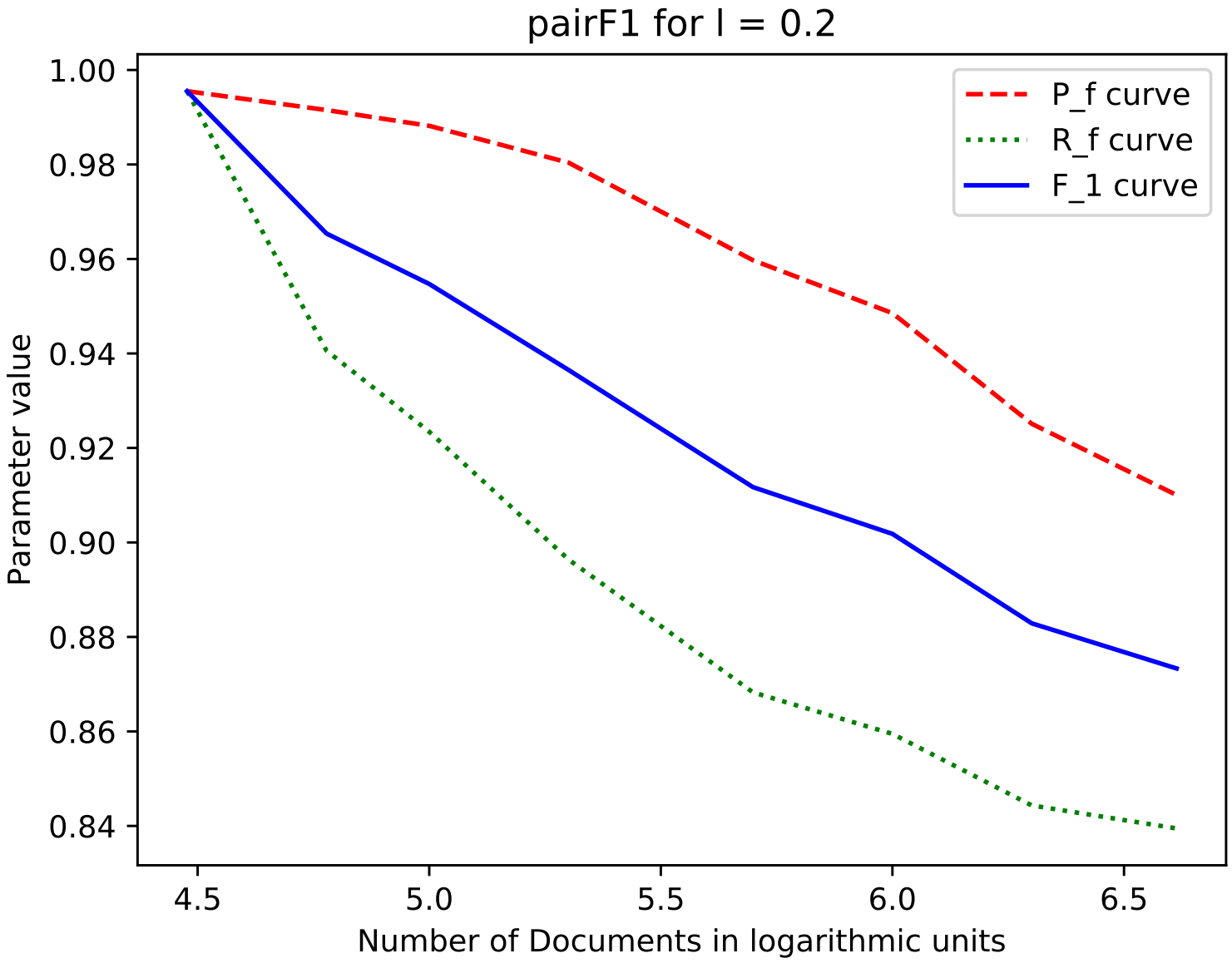
- It is understood from the above formula, that there is a distinct precision and recall value for each clustering problem, that is for each name.
- If we calculate precision and recall for each name separately, we have to average over the results that we get for each single name.
- We can then calculate F1 from the average precision and average recall over all names.
- We use this approach to obtain a final score for each correct number $\mathfrak{C}$ of clusters.

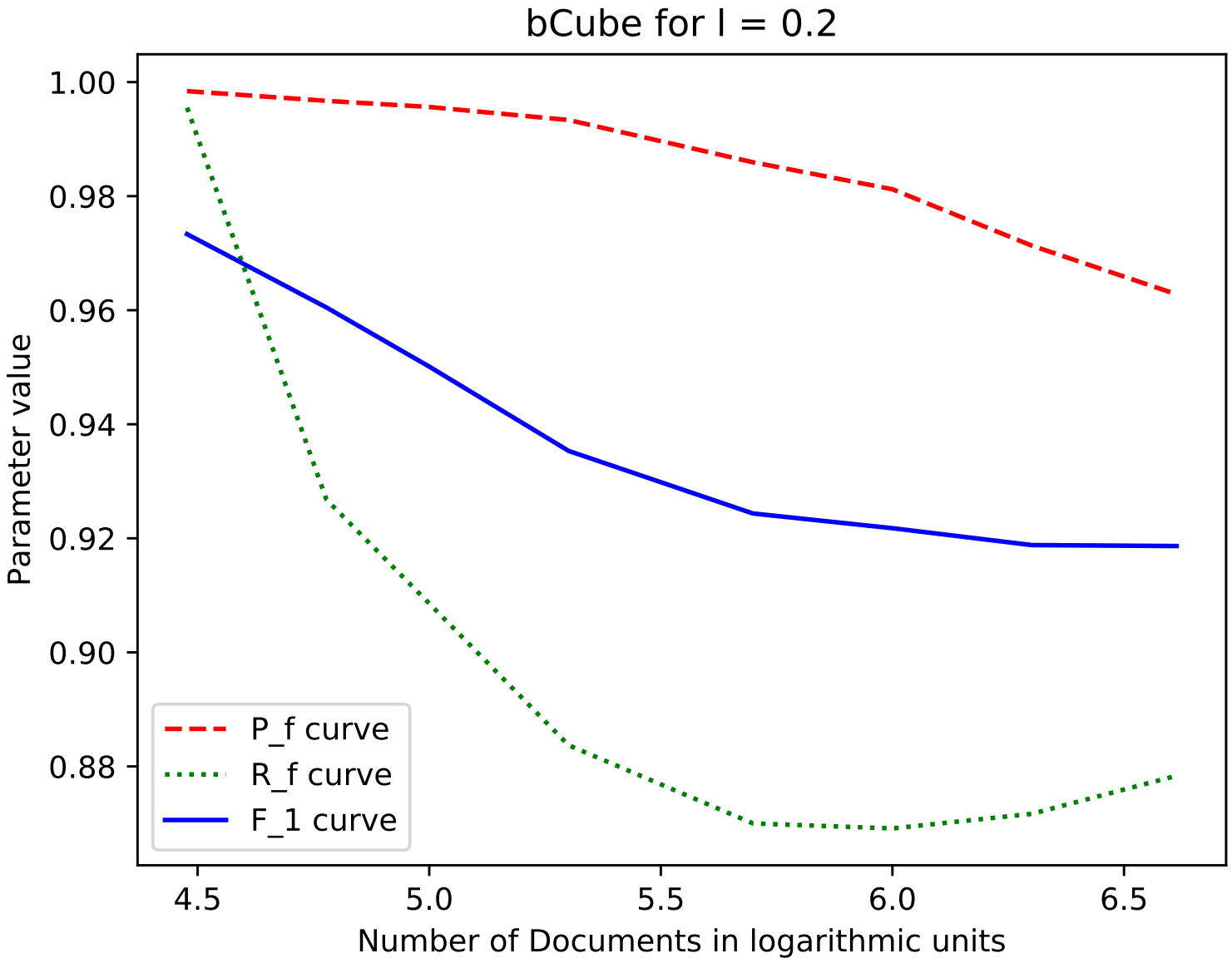
- We record a number of measurements during the clustering process in order to understand the behavior of our method and the effect of different versions and stopping limits.
- For a single block, clustering starts somewhere on the left of the plot with very low recall and maximal precision.As the process continues to merge clusters, recall increases and precision decreases.

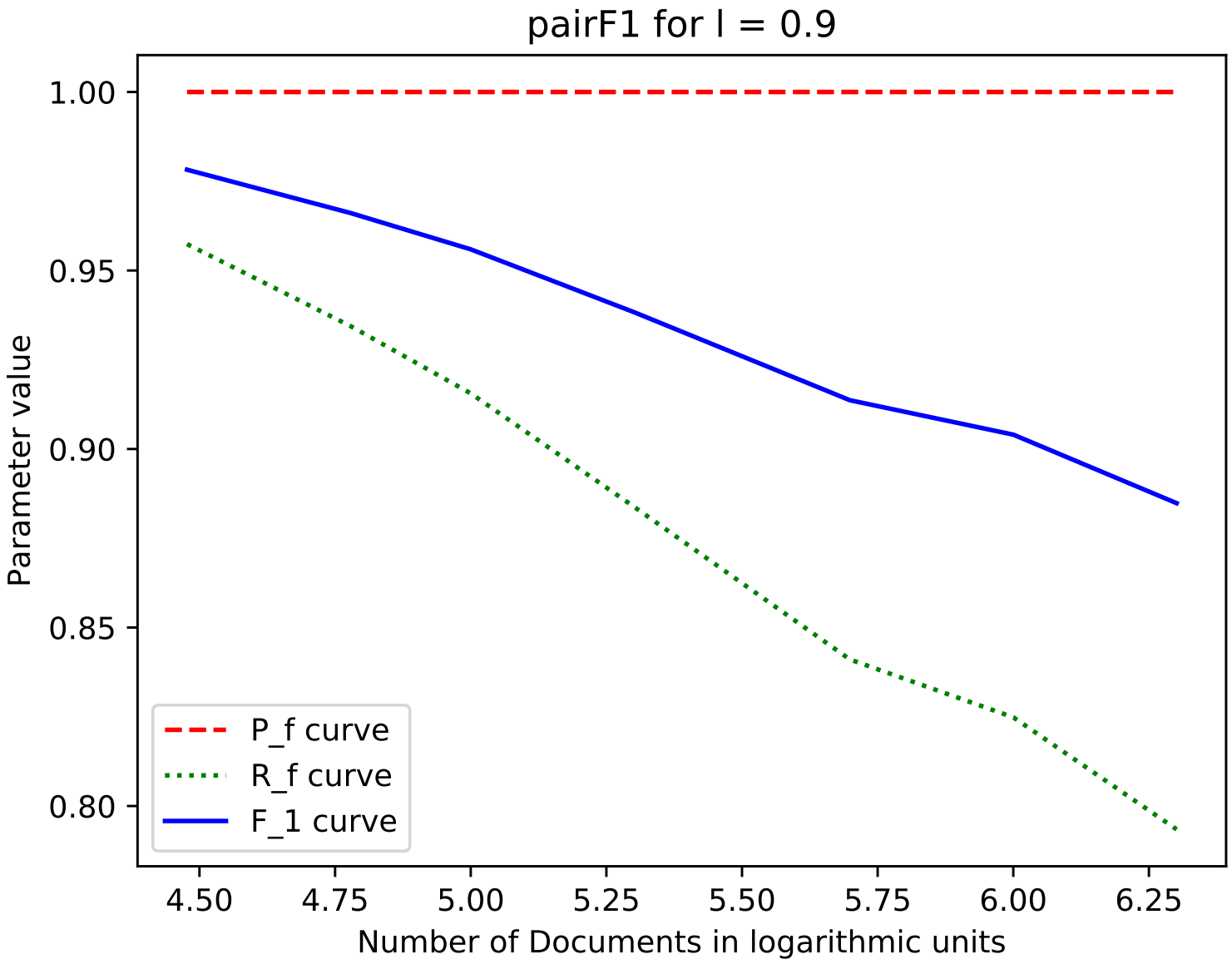
A direct comparison of our experimental results to those obtained by other researchers is not possible. In fact such a comparison would almost certainly not be fair as evaluation results depend heavily on so many different factors that reproducibility is close to impossible under normal circumstances:

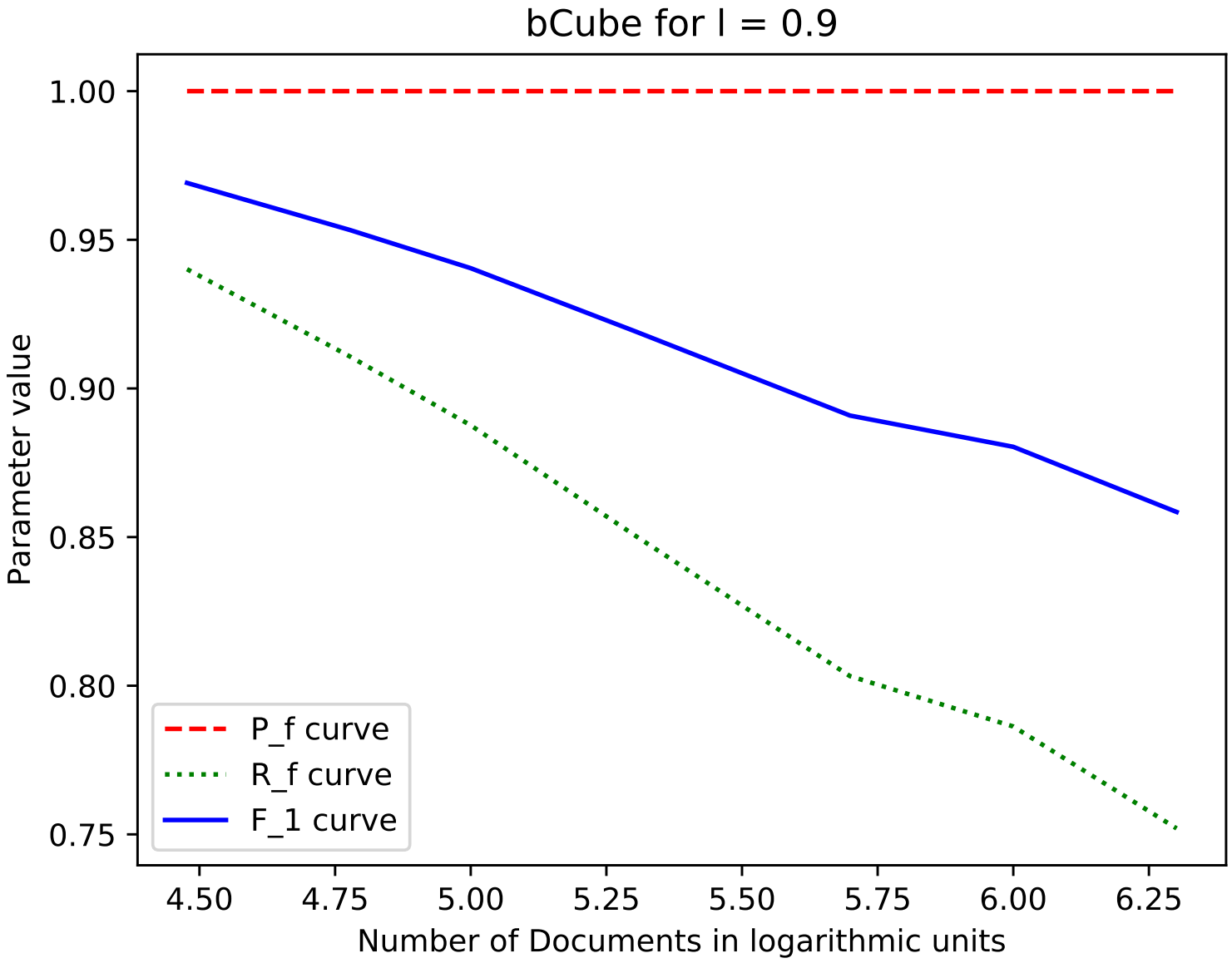
1. **Dataset**: size, distribution of authors, version of data set,domain, availability of features, completeness of author name, specification.
2. **Evaluation measure**: general choice of measure, micro/macro average, recall only inside block or over the entire dataset, pair- or element-wise comparison, counting pairs of equal mentions, evaluation for different problem sizes.

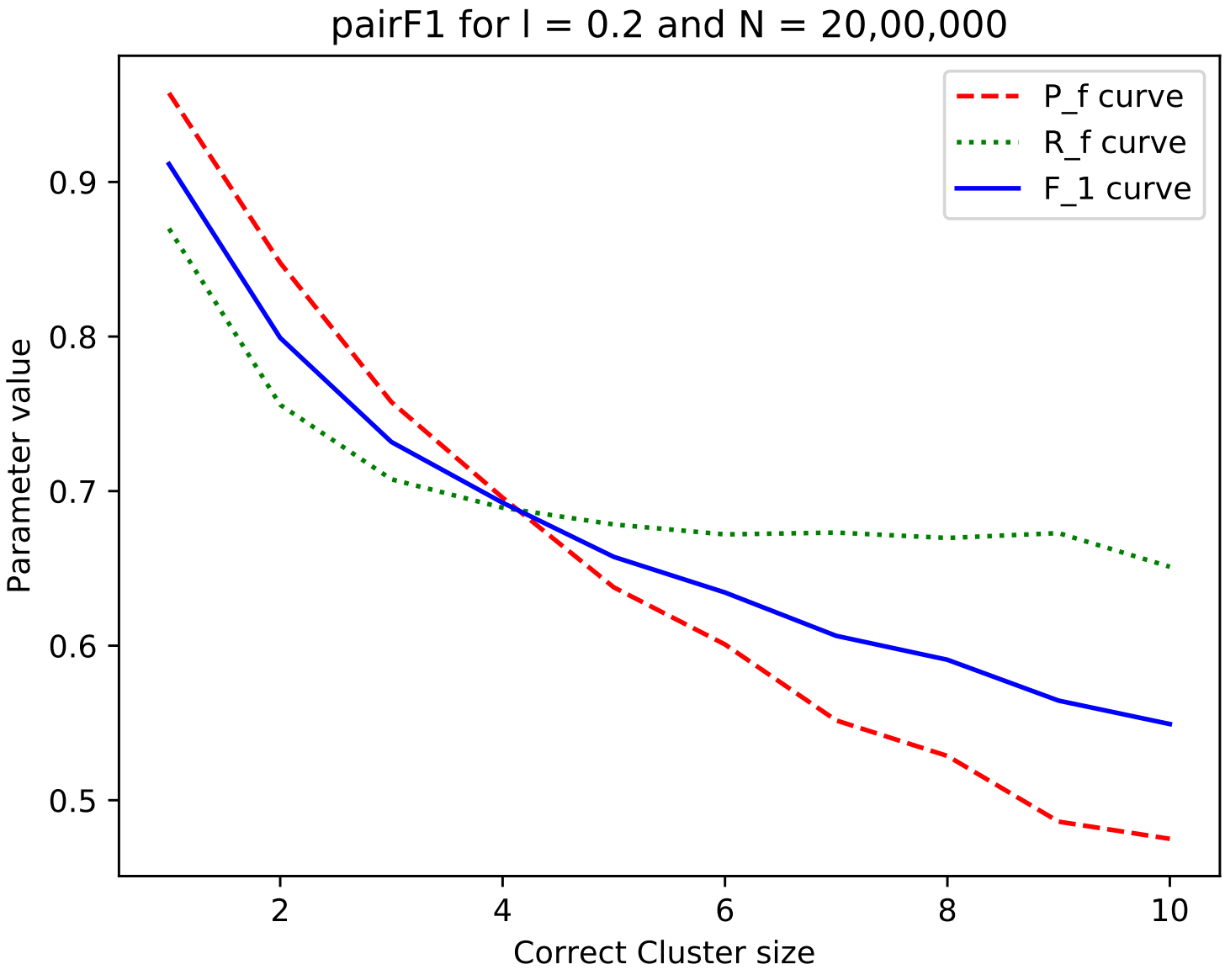
Graphical Data and Observations

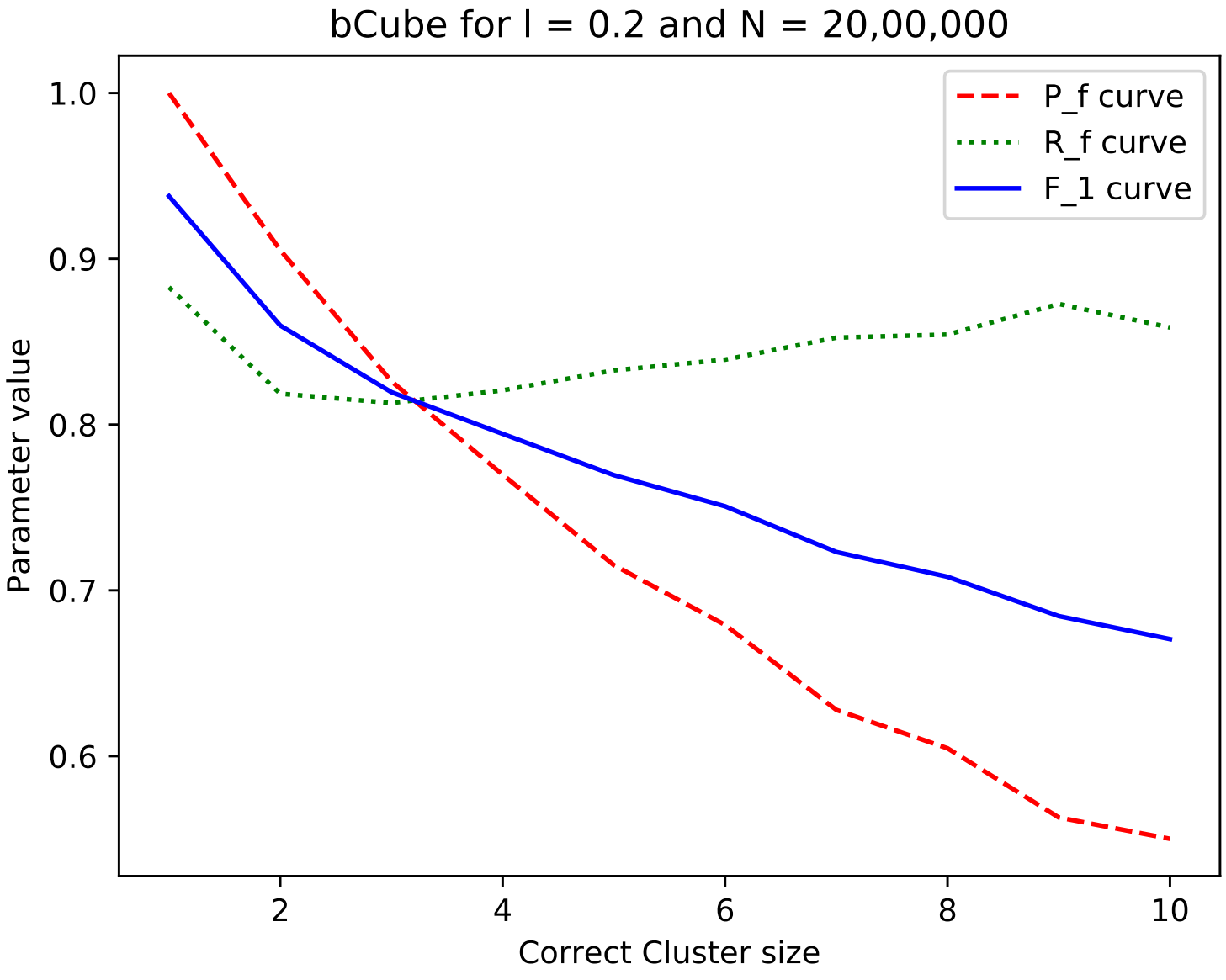

pairF1 for l = 0.2
P_f curve
R_f curve
F_1 curve
Parameter value
1.00
0.98
0.96
0.94
0.92
0.90
0.88
0.86
0.84
4.5
5.0
5.5
6.0
6.5
Number of Documents in logarithmic units

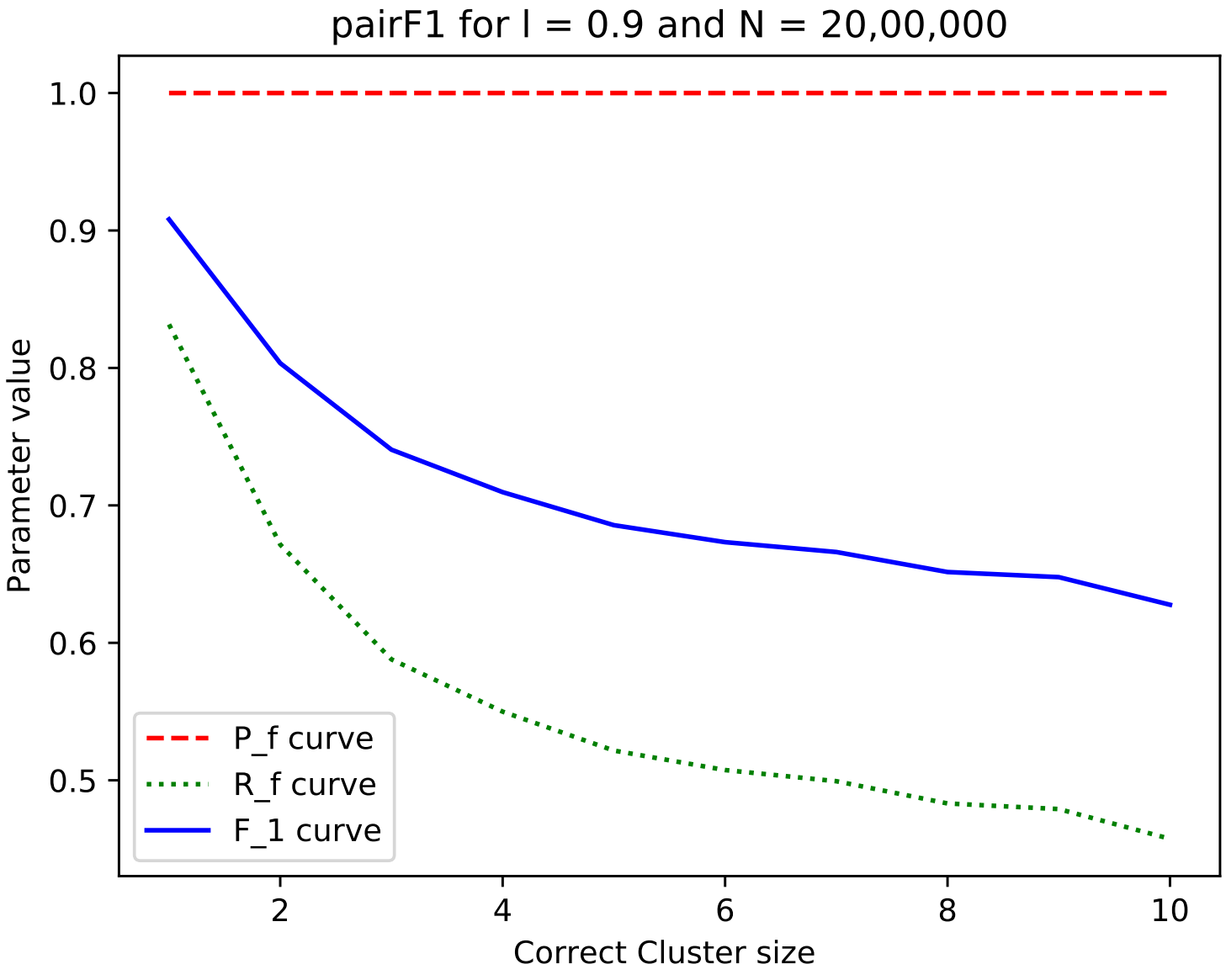

bCube for l = 0.2
1.00
0.98
0.96
0.94
0.92
0.90
0.88
Parameter value
P_f curve
R_f curve
F_1 curve
4.5
5.0
5.5
6.0
6.5
Number of Documents in logarithmic units

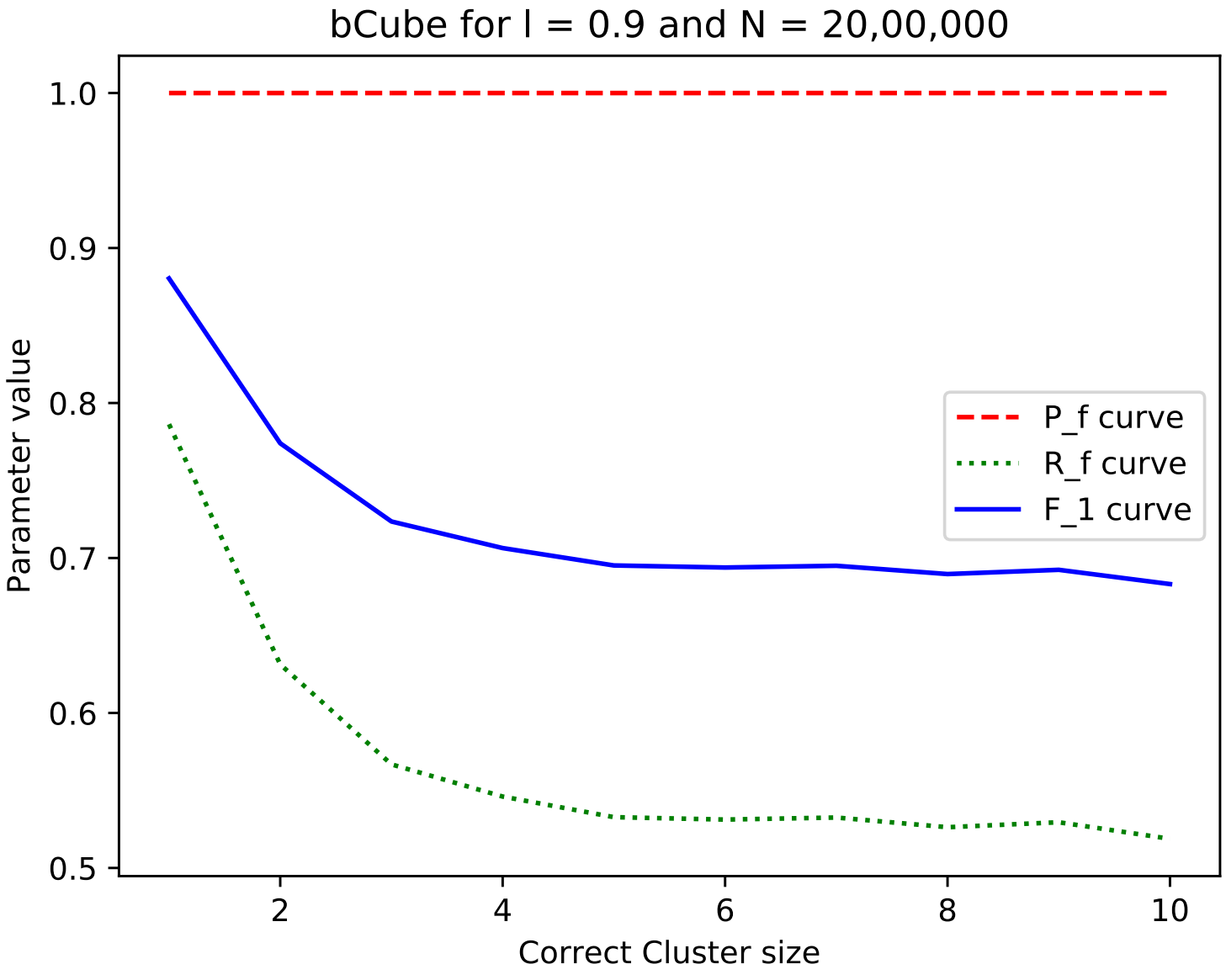

pairF1 for l = 0.9
1.00
0.95
0.90
0.85
0.80
Parameter value
4.50
4.75
5.00
5.25
5.50
5.75
6.00
6.25
Number of Documents in logarithmic units
P_f curve
R_f curve
F_1 curve

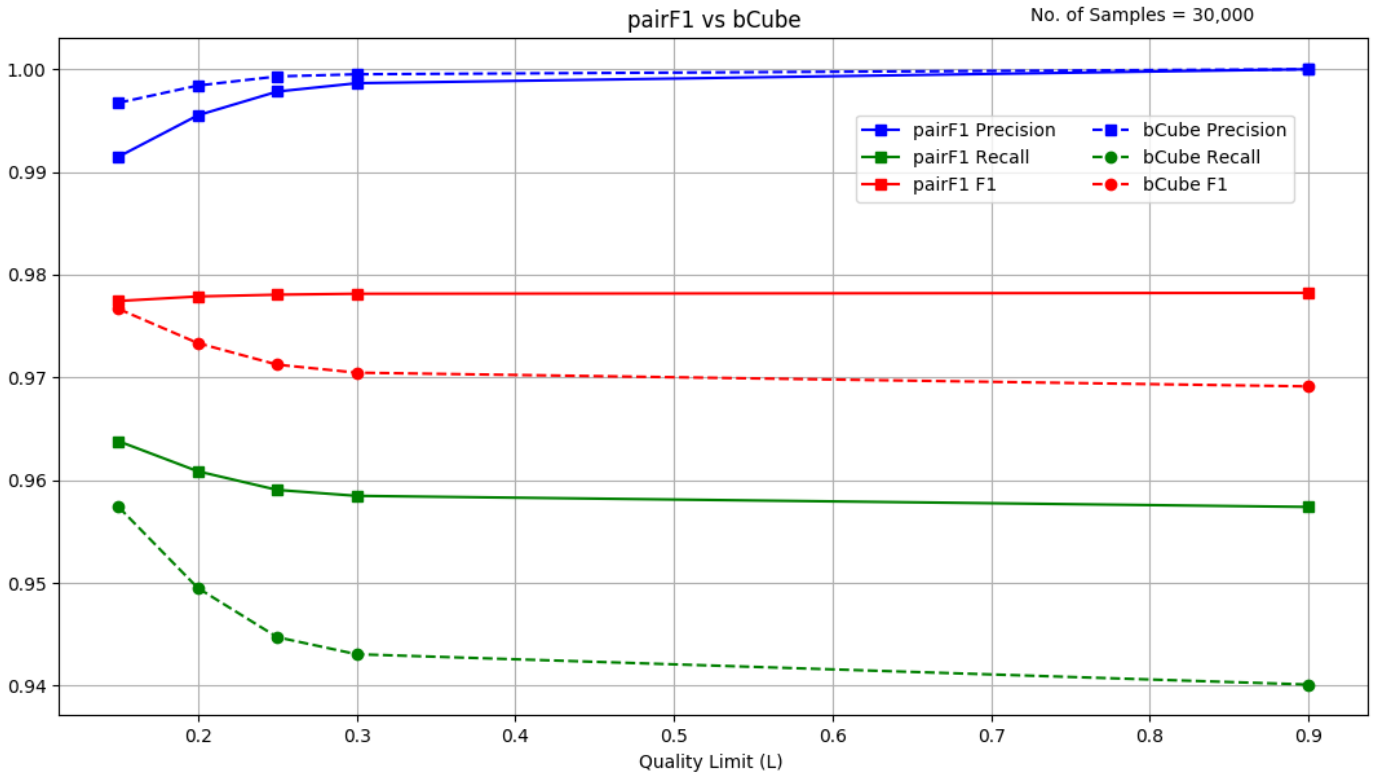

bCube for l = 0.9
1.00
0.95
0.90
0.85
0.80
0.75
Parameter value
P_f curve
R_f curve
F_1 curve
4.50
4.75
5.00
5.25
5.50
5.75
6.00
6.25
Number of Documents in logarithmic units

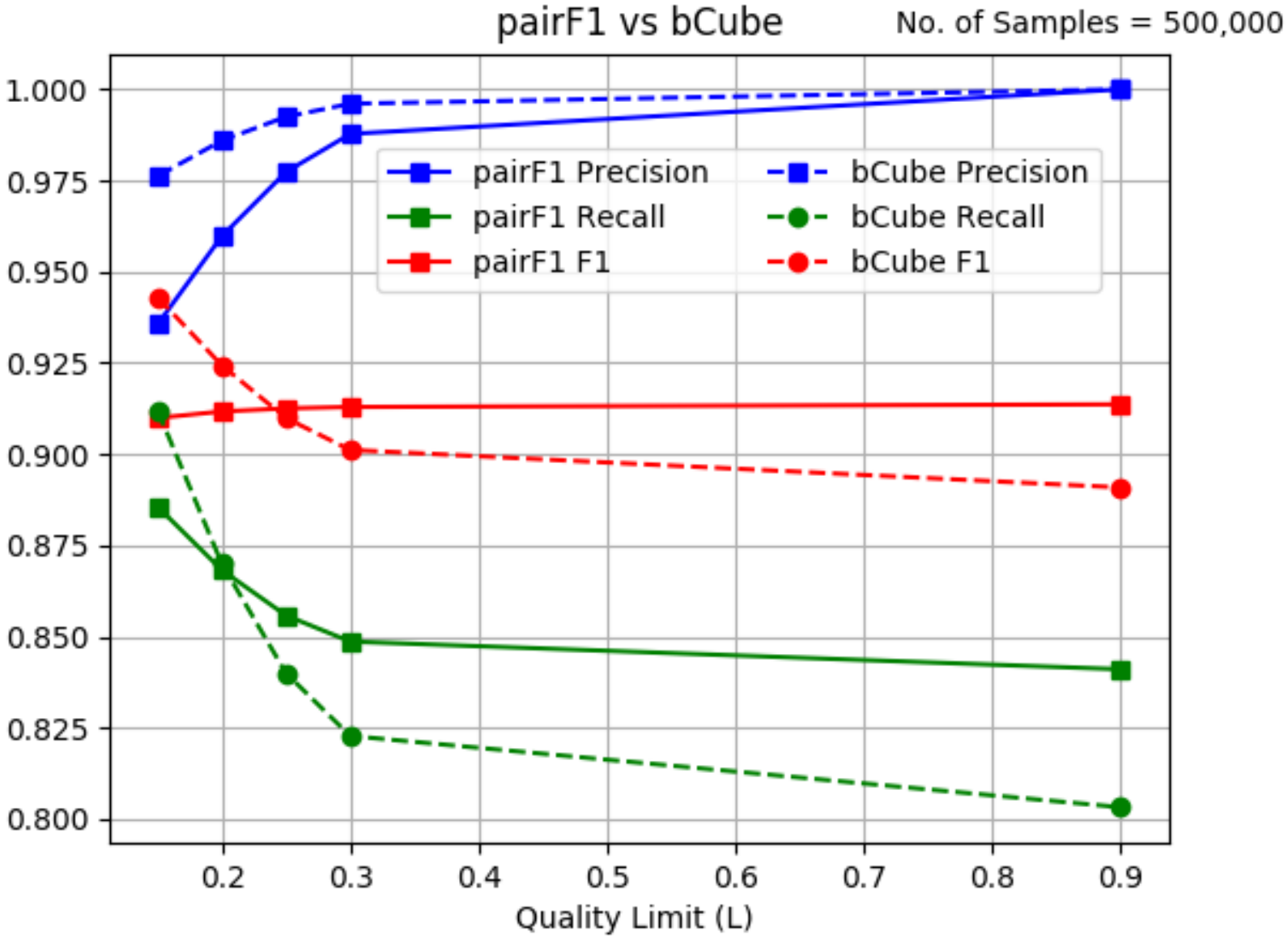

pairF1 for l = 0.2 and N = 20,00,000
P_f curve
R_f curve
F_1 curve
Parameter value
0.9
0.8
0.7
0.6
0.5
2
4
6
8
10
Correct Cluster size


bCube for l = 0.2 and N = 20,00,000
P_f curve
R_f curve
F_1 curve
Parameter value
1.0
0.9
0.8
0.7
0.6
2
4
6
8
10
Correct Cluster size


pairF1 for l = 0.9 and N = 20,00,000
1.0
0.9
0.8
0.7
0.6
0.5
Parameter value
2
4
6
8
10
Correct Cluster size
P_f curve
R_f curve
F_1 curve


bCube for l = 0.9 and N = 20,00,000
1.0
0.9
0.8
0.7
0.6
0.5
Parameter value
2
4
6
8
10
Correct Cluster size
P_f curve
R_f curve
F_1 curve


pairF1 vs bCube
No. of Samples = 30,000
pairF1 Precision
pairF1 Recall
pairF1 F1
bCube Precision
bCube Recall
bCube F1
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Quality Limit (L)


pairF1 vs bCube
No. of Samples = 500,000
pairF1 Precision
pairF1 Recall
pairF1 F1
bCube Precision
bCube Recall
bCube F1
1.000
0.975
0.950
0.925
0.900
0.875
0.850
0.825
0.800
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Quality Limit (L)

SampleSize --> ClusterSize	30,000	60,000	100,000	200,000	500,000	1,000,000	2,000,000
1	63,796	109,763	163,122	260,517	451,161	758,779	1,152,497
2	2,211	5,669	10,804	22,261	50,597	95,963	162,179
3	238	795	1,780	4,498	12,510	26,248	48,134
4	66	243	495	1,380	4,479	10,232	20,072
5	27	81	208	564	2,001	4,840	9,949
6	10	47	90	288	1,091	2,738	5,843
7	12	27	59	150	597	1,641	3,594
8	5	19	37	95	399	1,047	2,389
9	2	10	32	61	241	712	1,726
10	6	11	23	56	155	503	1,234

Related Work and Acknowledgement

In this section, we give a brief overview of the most cited literature on author disambiguation..

1. Strotmann and Zhao investigate the application of author disambiguation to citation networks.
   They conclude that the Web of Science researcher-ID used in this paper is a good source of author identity information, in contrast to other sources.
2. In another paper focusing on training aspects, Culotta et al.present a special similarity function and combine error-driven sampling with learning-to-rank.
   They find that their approach can be beneficial in terms of performance.

Observations

1. As we increase the number of documents, the accuracy tends to decrease. This is because on increasing the number of documents, the feature matrix gets sparse which leads to a decrease in accuracy.
2. bCube is in general, a stronger evaluation metric as compared to pairF1.
3. The value of the stopping limit l is changed to get different results. A very high value of 'l' eliminates all false positives but gives a lot of false negatives and vice versa for low values of 'l'. We have prioritised the false negatives and have chosen the value of 'l' accordingly.
4. There is scope of improvement in this code as the disambiguation is not very rigid. When the authors are easily distinguishable, we have a high accuracy which falls as the number of homonyms increases.

Acknowledgement

This project is just a reproduction of the original paper submitted by **Mr. Tobias Backes** of GESIS. We would like to thank him for making the paper available freely for our use.