Reviewer 1

I have read the revised manuscript. I am not still persuaded that the authors cannot use PMI as a weighting scheme, because if you can use frequency of "words" for each document, you can also compute PMI too.

Using tf.idf as a weighting scheme is basically a small improvement over the standard technique so far. Because tf.idf has some options, such as using a raw count c or log(c+1) as tf, or how to set the base of log, not to mention a comparison between PMI: as a journal paper, I think it is desirable to conduct such a systematic comparison since it is quite easy to do. When a researcher reads this paper, it is desirable for him to use this method including what kind of options he should adopt with confidence.

Since the improvements are basically slight, I would like to leave the editor whether this technical memorandum would be published in the current form (not satisfactory from NLP point of view) in BMC Bioinformatics.

**Response:** We thank the reviewer for their valuable comments and suggestions. Following your suggestions, we have implemented some other methods to compare with the one that we described in our paper. Specifically, we implemented the PMI method as well as some modifications of TF-IDF. If the reviewer thinks that it is important to include these methods in the main text, we are happy to do so.

Below, we provide a precise description of these methods.

First, we describe a procedure to apply the PMI method to our dataset. For a given word [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=w_1#0) in the corpus (in our context, a mass spectral feature in the set of all samples), we define [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=p(w_1)#0) to be the number of documents containing [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=w_1#0). For a pair of words [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=w_1%2C%20w_2#0), we define

[Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=p(w_1%2C%20w_2)%20%3D%20%5Cfrac%7B%5Ctext%7Bnumber%20of%20documents%20containing%20both%20%7D%20w_1%2C%20w_2%7D%7BN%7D%2C#0)

where [A picture containing shape

Description automatically generated](https://www.codecogs.com/eqnedit.php?latex=N#0) is the number of documents. Finally, we define the PMI score of the pair [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=w_1%2C%20w_2#0) is defined to be

[Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BPMI%7D(d_1%2C%20d_2)%20%3D%20%5Cfrac%7Bp(w_1%2C%20w_2)%7D%7Bp(w_1)p(w_2)%7D#0)

We remark that some authors use different normalizations for PMI. For a document [Shape

Description automatically generated with low confidence](https://www.codecogs.com/eqnedit.php?latex=d#0) we form a vector [Shape

Description automatically generated with low confidence](https://www.codecogs.com/eqnedit.php?latex=v_%7Bd%7D#0) indexed by words in the corpus. Specifically, the value of this vector at a word [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=w#0) is given by

[Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=v_d%5Bw%5D%20%3D%20%20%5Cfrac%7B%20%5Csum_%7Bw'%7D%201_%7Bd%7D(w)%20%5Ctext%7BPMI%7D(w%2C%20w')%7D%7BM%7D#0)

Here [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=1_d(w)%3D1#0) if [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=w%20%5Cin%20d#0) and [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=1_%7Bd%7D(w)%3D0#0) otherwise; and [Shape

Description automatically generated with low confidence](https://www.codecogs.com/eqnedit.php?latex=M#0) is the number of words in the corpus. Roughly speaking, [Shape

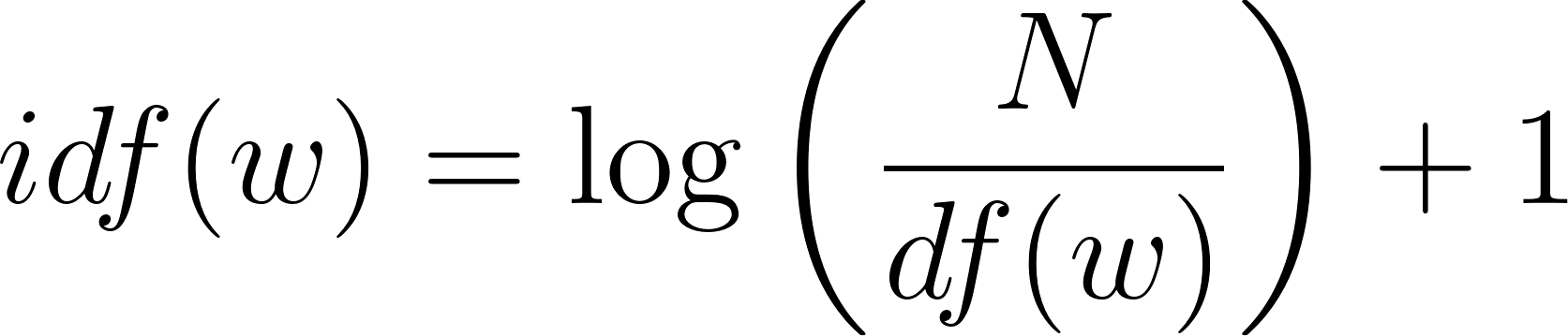
Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=v_d%5Bw%5D#0) is the average value of the PMI score for a document [](https://www.codecogs.com/eqnedit.php?latex=d#0) at [Shape

Description automatically generated with medium confidence](https://www.codecogs.com/eqnedit.php?latex=w#0).

Once the vector [Shape

Description automatically generated with low confidence](https://www.codecogs.com/eqnedit.php?latex=v_d#0) is calculated for each document, we can use the standard cosine similarity to compare them.

The reviewer also suggested some other weighting methods for TF-IDF, which we found very helpful. We are grateful to the reviewer for suggesting these methods. Using the standard library Sklearn, we have applied these alternative methods to our problem. We found that the standard weighting scheme remains the optimal choice for our dataset. We compared three weight schemes and the PMI method in the table below. If the reviewer would like for us to include this table in the main text, we can do that. We would also like to take this opportunity to modify the formula for IDF in our manuscript. The formula that we used should be as follows (see revision in the manuscript):

[](https://www.codecogs.com/eqnedit.php?latex=idf(w)%20%3D%20%5Clog%20%5Cleft(%5Cfrac%7BN%7D%7Bdf(w)%7D%20%5Cright)%2B1#0)

This is the formula that Sklearn uses instead of what is shown in textbooks (scikit-learn.org/stable/modules/generated/sklearn.feature\_extraction.text.TfidfTransformer).

Please note that the second method is where idf=1, and the third method is where we replace tf with 1+log(tf). We found that the second method and PMI method classified three out of eight (37.5%) of the samples correctly, while the method described in the manuscript classified five out of eight (62.5%) of the samples correctly. The third method also classified 62.5% of the samples correctly, but with scores lower than the method described in the manuscript. We found that the method described in the manuscript remains the most accurate approach for predicting which plants had been smoked in the clay pipes.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Blind clay pipe (BCP) | Key | Method described in manuscript | Second method | Third method | PMI method |
| BCP1 | *Nta* | *Nta* (0.0370) | *Nta* (0.0536) | *Nta* (0.0300) | *Nta* (0.0957) |
| BCP2 | *Nqu* | *Nqu* (0.1106) | *Nqu* (0.1405) | *Nqu* (0.1071) | *Nqu* (0.1322) |
| BCP3 | *Nob* | *Nob* (0.1145) | *Ngl* (0.1487) | *Nob* (0.1118) | *Ngl* (0.1194) |
| BCP4 | *Alu* | *Nta* (0.0934) | *Nta* (0.1334) | *Nta* (0.0881) | *Nta* (0.1373) |
| BCP5 | *Ngl* | *Ngl* (0.0884) | *Ngl* (0.1110) | *Ngl* (0.0809) | *Ngl* (0.1325) |
| BCP6 | *Lin* | *Ngl* (0.0844) | *Ngl* (0.1434) | *Ngl* (0.0747) | *Ngl* (0.1272) |
| BCP7 | *Auv* | *Auv* (0.0735) | *Cse* (0.1097) | *Auv* (0.0739) | *Cse* (0.1133) |
| BCP8 | *Auv* and *Nta* mixture | *Cse* (0.0813) and *Auv* (0.0617) | *Cse* (0.1317) and *Auv* (0.0959) | *Cse* (0.0727) and *Auv* (0.0592) | *Cse* (0.1106) and *Auv* (0.0978) |

Other Reviewer

GENERAL COMMENTS: The authors replied sufficiently to the critics of both reviewer and made changes/improvement accordingly. Doubts are clear now.

**Response:** We sincerely appreciate your feedback. Thank you so much!