

Dear Editor and Reviewers,

We would like to thank you for your kind words concerning our contribution. We have modified the article to reflect the reviewers' suggestions. The following is our explanation of the changes made, as well as how they address the comments raised by the reviewers. The bold text corresponds to the original comments. We are grateful to the reviewers for their constructive comments and suggestions in their revisions. They definitely have helped us improve the quality of the article.

We hope that the reviewers find the proposed revision of our manuscript satisfactory. We look forward to hearing from you soon.

## 1 Reviewer #2

*The authors did a great job addressing all the points raised. I have no further comments and would like to recommend the paper for publication. Thank you.*

We appreciate the positive feedback given towards this paper.

## 2 Reviewer #3

*The paper proposes a novel approach tackling multi-instance classification problems where bags or instances are labeled depending on whether they contain at least one positive instance or not. The topic is relevant for different fields of application and the publication is for sure relevant for this journal. The article is very clearly explained, very well written and thoroughly evaluated. The comparisons with other approaches all come with statistical significance tests, which makes the claims solid. I definitely argue for publication.*

We tried to be as comprehensive as possible with the experimental environment when testing our method and intend on continuing with this method of experimentation. We thank and appreciate your positive and constructive comment on this.

*Comment 1: It would be beneficial to add a short sentence in the introduction to explain with an example why concretely MIL is beneficial for training / deploying in real applications. The authors cite various fields (e.g. web mining, action recognition) and for at least one of these it would be nice to have a short statement about why MIL is useful for that application. I guess this can be found in the respective cited works, but for the purpose of self containment an example would be nice - e.g.: why is it useful to have a method to train with bags for action recognition problems?*

Adding an example of why MIL is useful, rather than using traditional learning, would definitely add to the quality of the paper by making it more self contained. We thank the reviewer for pointing this out and have added a sentence to the introduction with an example of how MIL better represents the real-world problem of drug activity prediction, rather than a traditional dataset. Specifically: “*In the case of drug activity prediction, MIL allows molecules to be represented as related instances contained in a bag rather than single instances with their own labels.*” Another example, for the case of action recognition, individual instances represent some activity data recorded over time, and the task at hand would be to determine whether a specific action of interest is being performed for a majority of the time. In this case, a bag would represent actions performed over time, and is labeled positively (e.g. for action “drinking from a cup”) if at least one instance in the bag indicates drinking from a cup.

*Comment 2: In the abstract (and even more in the conclusions) it is quite usual to have some concrete numbers defining the performance. Like that, someone reading the abstract already can get an idea of how much better this algorithm could get compared to other state-of-the-art approaches.*

We took your advice on this and added some more specifics on the performance of MIRSVM against the competing algorithms. Since we used 6 metrics to measure and evaluate performance, we decided to mention the accuracy in the abstract and conclusion, just for the sake of being comprehensive. We hope that in doing this we fulfilled your suggestion.

*Comment 3: The whole approach, along with other related ones, is suitable for problems involving exactly one positive and one negative class. It would be nice if the authors could briefly discuss about ideas (if there are any around) of how to extend such an approach for problems where there are negative bags and positive bags with different dominant classes. I'm thinking here about modelling null class as the negative class and then different (more than one) positive classes. But maybe it's not relevant, it just occurred to my mind while reading the paper.*

Thank you for the interesting question. The origin and core of MIL stems from a binary classification problem and the standard assumption by Dietterich et al., where bags obtain a negative label if all instances within it are negative, and otherwise the bag is labeled as positive. To the best of our knowledge, we have not encountered a multi-class MIL problem. However, one idea for an extension of MIRSVM for a multi-class application would be to use methods such as one-vs-one (pairwise classifier) or one-vs-all. These ensemble methods are known to be a little expensive in terms of run time but they merit investigation if the multi-class problem existed. We added a short discussion on this in Section 5 and appreciate you bringing our attention to this topic.

*Comment 4: Just a couple small clarifications / grammar issues in the text:*

- *Section 1 (page 2): instead of "assuming the bags' internal distributions" I'd write "making specific assumptions about the bags' internal distributions"*
- *Section 2.2 (page 5): I'd complete the sentence starting with "MIRSVM, is an iterative...", writing "Our proposed contribution, MIRSVM, is an iterative..."*
- *Section 4.7: "rather than use the instances" → "rather than using the instances"*

Thank you for pointing those clarifications and grammar issues. As you have suggested, we modified the text to reflect your suggestions.

*Within (or after) the conclusion, it would be nice to have some kind of outlook - or is it kind of "problem solved conclusively"? Are there still areas of improvement to be addressed?*

Thank you for looking ahead. One area to investigate would be to use a different SVM solver (instead of a quadratic programming, QP, solver) during the training procedure for the sake of speed. Another would be to investigate distributed implementations of the algorithm. A third idea would be to experiment with highly imbalanced datasets and observe how the algorithm performs. We have included these open ideas in the conclusion. We appreciate you mentioning this.