Response to Reviewers

COATi: statistical pairwise alignment of protein-coding sequences

Juan J. García Mesa   Ziqi Zhu   Reed A. Cartwright

Dear Editor,

We are resubmitting our manuscript, "COATi: statistical pairwise alignment of protein coding sequences", for your consideration. This manuscript describes a new model for aligning protein coding sequences which produces more accurate alignments, while being robust to common artifacts found in genomic annotations. Our approach is implemented in the software package, COATi, which is released via an MIT license and available on GitHub. While COATi can generate multiple-sequence alignments, this paper is focused on our pairwise aligner which is has received more attention and validation. Additionally, to promote open science and reproducibility all data, procedures, and scripts used to validate COATi and compare it with alternative methods have been uploaded to GitHub.

We would like to thank the associate editor and two reviewers for their helpful comments and suggestions. As requested, we have broadened the scope of the manuscript from a software note to a larger research paper about our COATi software. Nearly all of the paper has been rewritten, expanded, and reconfigured to provide more details and background about our work. Importantly, we now use empirical results from the human-gorilla sequence pairs to initially compare aligners before we show results on semi-empirical benchmarks. If certain concepts need additional exposition, we will be happy to make those adjustments.

We have addressed all of the concerns brought to our attention by the associate editor and reviewers as described below.

Sincerely,
Reed A. Cartwright, on behalf of all authors

Associate Editor Comments

1. Comment: *The general consensus about this paper is that the question and method (codon-aware alignment) are both extremely important and suitable for MBE, but the manuscript, written essentially as a short software note, is simply too sparse to adequately present the solution, eliding over important issues and methodological details.*

   We thank the editor and reviews for the positive feedback. As mentioned above, we have now broadened the scope of the manuscript from a software note to a larger research paper about our COATi software. We have rewritten nearly the entire paper as we expanded to scope in response to feedback.

2. Comment: *In addition to the reviewers' comments, one more critical question strikes me. The method makes uses of a simplification procedure predicated on the concept of treating one sequence as if it is the ancestor of the other, which allows modeling the problem using FSTs rather than HMMs. This modeling choice introduces additional directionality into the model. If there are two sequences (A and B) the method arbitrarily chooses one for the ancestor and one for the descendant. What is not at all clear is whether this directionality is important, i.e., is the same result generated if the model goes A–>B as it would if it went B–>A. This is a fundamentally critical question that needs to be addressed as the stability of any alignment method should not be dependent on the order of the sequences within the input file.*

   The editor is correct that our model is directional. This is not a limitation of the model but rather a feature. We have added more details about our FSTs and our reasoning for using them. In particular, we now explain that our input sequence (i.e. "the ancestor" or "reference") is required to have a length that is a multiple of 3 and missing stop codons and ambiguous nucleotides (circa line 99). The output sequence (i.e. “the descendant” or “non-reference”) does not have these restrictions. This asymmetry is due to COATi needing one high-quality reference sequence to establish reading-frame contexts.

In order to test the impact of directionality on our results, we have included results when using gorilla as the reference sequence. The results are very similar to using human as the reference sequence.

We have thought further about how to reduce the dependency the directionality of our model, but we do not have any improvements to report at this time.

Reviewer 1 Comments

1. Comment: *It is surprising this work has not been extended to something larger*

   We appreciate the reviewer’s suggestion that our ideas may be suitable for longer sequences. We feel that COATi, with a bit of work, would be the correct tool for aligning long-read sequences generated from target genes against a reference. Our existing suite of FSTs can support both exonic and intronic regions, and via concatenation we can create a new FST that supports specific reference genes’ exon and intron segments. The challenge is to figure out the best user-interface for specifying such information as well as validating the result. We choose to tackle these challenges in a future research project.

   We now comment on this in the discussion.

Reviewer 2 Comments

1. Comment: *I would like to start this review with an apology. This review took me a bit longer than I had hoped.*

   We also apologize for taking longer than expected to submit this revision, as the first author needed to focus on defending his dissertation last semester. We are happy to report that he passed his defense without issue and is now Dr. García Mesa, joining Dr. Zhu who graduated the year before.

   We also discovered a bug in COATi’s scoring algorithm (used for defining “perfect alignments” in the analysis) as well as minor issues with how we were using MACSE and ClustalΩ. We opted to take the time to regenerate our results after fixing these issues. Tables and figures barely changed and no conclusions were affected.

2. Comment: *With that said, I would like to start by saying that Garcia Mesa and colleagues have taken on an important problem that has (in my opinion) received substantially less attention than it should – the problem that sequence alignment for coding regions ignores the potential for indels to occur out of phase with codon boundaries. I find the approach that the authors propose to be exciting and potentially very useful. I also believe that MBE is an appropriate venue for this work, and I think MBE readers will be excited.*

   Thank you for the encouraging words.

3. Comment: *However, I also feel that the manuscript is hard to read, and I think it requires substantial revision for clarity before it is acceptable.*

   We have taken the opportunity to do a substantial revision of the manuscript, including aiming for a larger research paper, instead of a brief note. He hope that it is now easier to read and answers the concerns of the reviewers. If further details are needed, we are happy to include them.

4. Comment: *I will start with the good part of the manuscript – the problem is stated clearly in the introduction. Figure 1 is excellent, and it illustrates the fact that codon-based aligners can make the error of forcing indels to lie between codons.*

   Thank you for the encouraging words.

5. Comment: *I am confused by the state space. Perhaps they could be enumerated in the supplementary information. Obviously, there are 61 possible codon states (assuming the standard code). It strikes me that there are 12 possible 1 nucleotide, 2 gap states (i.e., N–, -N-, and –N, where N is any nucleotide) and 48 possible 2 nucleotide, 1 gap*

states (i.e., NN-, N-N, and -NN). I would assume that -N- and N-N would be very rare, but I am fine with including them for the purpose of completeness. There is also the all-gap state, which brings the total up to 122 states.

Our FSTs do not have the same state space concept as in phylogenetic models. Consider the codon substitution FST (Fig. 3a). Its input and output sequences have state spaces of four symbols (A, C, G, and T). This FST has two internal states (M and S) and 61x61 arcs from S to M. Each of these 3721 arcs consume three nucleotides from the input sequence, emit three nucleotides to the output sequence, and represent one transition in a standard codon substitution model.

Similarly, consider the ambiguity FST (Fig 2b). Its input sequence has four symbols (A, C, G, and T) and its output sequence has five symbols (A, C, G, T, and N). It has two internal states (M and S), and 8 arcs from S to M. Four of these arc consume a nucleotide and emit the same nucleotide. The other four arcs consume a nucleotide and emit N. This is the FST that allows COATi to support ambiguous nucleotides in the non-reference/output sequence.

To handle indels, the indel FST (Fig 3b) has a state space of four symbols (A, C, G, and T) with a special symbol, epsilon, that represents an empty match. An arc that reads nothing (epsilon) and outputs a symbol represents an insertion, while an arc that reads a symbol and outputs nothing (epsilon) represents a deletion.

We don't specifically model these processes in a single FST as we use the composition algorithm to combine separate FSTs that represent codon substitution, indels, error, and sequencing ambiguity to produce one FST that handles all four processes (Fig 3c).

We hope the updated manuscript offers clarity about these points.

6. Comment: *I can see how one can forbid ungapped emission of stop codons. However, I could see problematic cases like ...TACAAG... (...Tyr-Lys...) where a deletion of three bases after the first T -or- after the first A would yield a stop codon. Is that forbidden? Or is it allowed in the "low quality sequence" in the pair?*

   Our Indel FST does not consider codon translations when deleting or inserting nucleotides. Indels can occur that create early stop codons. However, this will only happen if the output sequence is of low quality and contains an early stop codon. We consider this a feature of COATi as it will not fail to align output sequences that contain early stop codons.

7. Comment: *I don't see how they can use a codon model when indels can occur within codons. This is especially true for the Kosiol ECM. Since the ECM is purely empirical it can be viewed as a matrix of rates of change between 61 symbols. I think this should also be true for the Muse and Gaut codon model.*

   We are able to easily mix codon models with indels that occur within codons because we are modeling alignments using FSTs. The composition algorithm allows us to combine an FST representing a codon substitution model with an FST representing an indel model (Figure 3). Composition does produce the complex graphs that you would expect from combining a codon model with an indel model, and we show a short example in Figure S1. Composition is a complex but well established algorithm in the FST literature, and we have chosen to refer readers to Mohri et al. (2005) if they would like to know more.

8. Comment: *It might be possible to improve Fig. 2 to clarify this. It is very difficult to work through, and I am still having trouble despite staring at the figure quite a bit. I wonder if showing the example alignment for Fig. 1 might help. Perhaps something could be done in the supplement.*

   We have updated Figure 2 and added a new Figure 3 that we feel more adequately explains the finite-state transducers that we are using and the composition algorithm that allows us to combine them in a reasonable manner. We also include in the supplement Figures S1 and S2 which show examples of fully composed alignment graph between the codons "CTC" and "CTG" as well as the best path through the graph.

9. Comment: *Finally, how is the "low quality sequence" in the pair identified?*

   By default, COATi uses the first sequence in the input file as the input/reference sequence and the second sequence as the output/non-reference sequence. It also contains flags to customize this behavior. This gives users the power to place low-quality sequences as the output sequence when aligning a pair of sequences using COATi FST. We now mention this in the manuscript circa line 108.

10. Comment: *Perhaps one of the biggest problems with the methods is that the first two paragraphs in the results cover a lot of methods. However, they cover them in an abbreviated format that is almost impossible to follow. For example, I think that the way indels are simulated are by taking alignments of empirical human-gorilla sequence pairs and then mapping the gaps from those pairs onto simulated sequences, where those sequences were simulated without gaps. Is that correct? I think the actual methods need to be spelled out much more carefully and put in the methods section.*

    We have added details about how we created our semi-empirical benchmark alignments (circa line 227). The revision now contains information about our procedure, as well as an example of it being applied to data. We believe that this update answers the concerns of the reviewer.

11. Comment: *In this context, it would be good to state how Ka and Ks are calculated in the methods. The supplement has one sentence about how they are calculated and a lot about what Ka/Ks > 1 means. The material about interpretation is not a problem, but I think many MBE readers will not need it. I'd really like to see an explicit articulation of how the stats were collected in the main paper.*

    We used sequinr to calculate Ka and Ks and have expanded our methods to explain which method that sequinr uses to calculate these stats and how that method works (briefly).

12. Comment: *I would also like to authors to articulate something about equally optimal alignments (or slightly suboptimal alignments). The github page has information on "coati sample - align two sequences and sample alignments" and it seems to me that equally optimal alignments might be common. In fact, if one looks at the biological alignment in Fig. 1 the second indel:*

    ```
    …T-- -CG… (Ser)
    …TCC CAG… (Ser-Gln)
    ```

    Could just as easily be:

    ```
    …TC- --G… (Ser)
    …TCC CAG… (Ser-Gln)
    ```

    In fact, this alternative alignment seems better. Minimally, the first alignment requires a three-base pair indel and a C<->A change whereas the second would only require a three-base pair indel.

    Thank you for pointing out a typo in our Figure 1. It was supposed to be TCC CCG (Ser-Pro) in the bottom sequence there. (No C<->A change.) We've fixed it in the revision.

    Back to your question, `coati alignpair` breaks ties according to the implementation of the "shortest path" algorithm in the OpenFST library (when using an FST-based model like tri-mg). We currently use OpenFST to implement our FST-based algorithms. We now mention this in the manuscript (circa line 116).

    Furthermore, `coati sample` doesn't need to break ties as it randomly samples alignments from the posterior. This is useful for studies that need a better picture of alignment space than retrieving a single, best alignment. We've utilized it to study indels across the tree of life (e.g. Zhu 2022), and it will be described in greater detail in a future paper.

13. Comment: *Finally, I would like to see the impact of alternative alignments on distances – perhaps logdet or K2P or F84 distances. It might be informative to ask how often each alignment method results in the largest distance. It strikes me that methods that "overalign" (e.g., their Fig. 1c) would tend to have the largest distances.*

    It would also be nice to summarize Ka/Ks and evolutionary distances for empirical human-gorilla alignments. It strikes me that they already have the data, having used it to get parameters for simulation.

    We would like to thank the reviewer for the suggestion to analyze the empirical human-gorilla alignments. Using this suggestion, we have expanded our comparison of COATi and the other aligners. We now use both empirical human-gorilla sequence pairs and semi-empirical benchmarks. From empirical data, we can confirm that aligners that overalign sequences (fewer gaps, more mismatches) do produce higher K2P distances (Table 1 and Figure 4). For the benchmarks, aligners that had higher alignment errors had higher root-mean-squared errors for K2P distances and lower F1 scores for selection as well.

14. Comment: *Minor (but still important) issue: The values in the supplementary information tables are reported to five decimal points. It is a judgement call whether this is overkill for the proportions, but it is very distracting for integer counts (i.e., just use "5678" rather than "5678.00000" for the number of perfect alignments)*

    We have adjusted our tables to be less distracting.

15. Comment: *I hope the authors are given a chance to revise this. In my opinion, the approach has a lot of potential. But I also think it needs a lot of work to clarify. In fact, my confusion about how the simulations were done is such that I would really like to see methods presented more clearly before I make a final decision on validity. If they did the simulations the way I described I think their approach is good, but I am honestly not sure given what is written.*

    Thank you for the encouraging words.