Shouji Pre-Alignment Stress Testing

Algorithms and Data Structures in Bioinformatics

Andrew Millward

May 3, 2021

1 Introduction

Performing sequence alignments between a reference sequence and many samples is often an incredibly computationally expensive task that leads to significant bottlenecks in many applications. However, heuristic methods have been developed that help to bypass some of the computationally expensive alignment procedure by effectively vetting the sample sequences and eliminating those that are deemed unacceptable to the filtering heuristic. One such method published in November 2019 details a novel filtering heuristic called the Shouji Pre-Alignment Filter [1]. Compared to standard sequence alignments, Shouji can be computed in both linear time and space, allowing many sequences (sometimes as much as 90% of the input sequences) to be eliminated before the alignment actually begins. This process leads to massive improvements in the end-to-end execution time of a multi-sequence alignment. However, there do exist some potential downsides to the standard Shouji alignment algorithm that will be examined in greater detail in this report.

1.1 Mechanisms and Limitations

Firstly, when it comes to the Shouji alignment algorithm, the primary mechanism employed to reduce the time and spatial complexity is accomplished using the assumption that the best alignment will be present on or near the primary diagonal of the alignment matrix [1]. Researchers know that this assumption is not always valid as some poorly aligned sequences can incur large gap penalties in order to properly fit with the reference sequence. With that said, for sample data, and for the overwhelming majority of the time, the assumption holds. This is due to the fact that when people attempt to classify a reference to a large sample set to determine various characteristics such as biological abundance, often the associated sequences will be present and at least closely resemble the reference. Nonetheless, errors can still occur.

For instance, although sequencing technology has radically improved over the past few decades, it still is not perfect. According to Pfeiffer et al., the average error rate in sequencing is around $0.24 \pm 0.06\%$ with slightly mutated sequences accounting for $6.4 \pm 1.24\%$ of samples [2]. Combined, these error vectors could potentially lead to some complications for algorithms such as the Shouji pre-alignment filter due to Shouji's reliance on smaller neighborhood maps which only occupy the diagonal of the alignment matrix. Specifically, when the algorithm considers a neighborhood map of only 4 characters in width as is used frequently within the paper, a single mutation within that zone could impact the overall alignment score for up to 4 maps, potentially giving inaccurate concentrations of zeros in the resultant alignment bit-vector for use in checking the pre-alignment against the edit distance threshold. This issue opens the window to both potential false-positives and false-negatives in the alignment process. As a result, the experiment outlined in this paper will address these concerns across a variety of different possible error vectors and perform a comparative analysis of these sources and their relative impact.

2 Stress Analysis Methodology

When analyzing the resiliency of the Shouji alignment filter against various error vectors, this report will consider several different sources for error and run simulations of that error across a range of realistic error rates based on the aforementioned study by Pfeiffer et al. The error sources and reasoning for their inclusion in this study are as follows:

1. **Substitution Errors**: This is one of three very common type of error present in sequencing. To simulate a substitution error, this paper will take a valid input sequence of the same size as the sample sequence (100bp) and a given error threshold (0%-2.5%) and will proceed to iterate through each character and mutate those which yield a random number less than the error threshold. Once this occurs, a random character will be substituted in place of the existing one.

2. **Insertion Errors**: Insertion errors, like substitution errors, are also very common in sequencing. To simulate insertion errors without adjusting other factors, the simulator must maintain the same sequence length following the mutation. It will therefore take the sample sequence (100bp) and the error threshold (0%-2.5%) and will randomly insert characters while iterating through using a similar mechanism as before. Any character beyond 100bp will be cut from the sequence in order to control for sequence length in this case.

3. **Deletion Errors**: Like insertion errors, deletion errors will simply iterate over the input sequence of 100bp and delete when a random number is generated below the error threshold. To maintain 100bp sequence length, the deletion simulator will input 'X' characters at the end of the sequence to prevent alignment from occurring against those characters.

4. **Size Mismatch Errors**: Due to the fact that Shouji relies heavily on very small windows called neighborhood maps to perform alignment, adjusting the length and positioning of the alignment can pose a massive risk to the efficacy of the algorithm. As a result, this portion of the simulator will simply replace 0-25bp at the beginning or end of the sequence based on the error threshold with an 'X' character. Since this relies on higher difference thresholds than the others, it will only be compared in normalized data.

5. **Gap Character/Data Loss Errors**: These errors are very similar to substitution errors with the difference being that characters are substituted with 'X' instead of another base pair. This source is being tested since possibilities for data loss or uncertainty could hypothetically arise, so it will be tested for thoroughness purposes.

2.1 Simulator Assumptions

In order to assess the efficacy of the Shouji pre-alignment filter at various error thresholds, the simulator will make several assumptions about the Shouji alignment algorithm. Firstly, the simulator will assume that the results of the Shouji alignment filter with no error introduced are correct and accurate as a baseline measurement to test against. This is a reasonable assumption given that the original authors found the filter to have a 0% false reject rate [1]. This is very important because falsely rejecting a correct alignment is extremely bad for a filtering heuristic that needs to accurately narrow down sequences without throwing out the most optimal ones at the start. Shouji is noted, however, to have a non-zero false-accept rate, but this is not too much of an issue since false-accepts only lead to more time spent in the alignment phase rather than potential for a complete failure of the alignment process. Secondly, this simulator will make the assumption that error rates will typically fall in line with the $0.24 \pm 0.06\%$ error rate with $6.4 \pm 1.24\%$ of samples experiencing mutations from Pfeiffer et al. and will therefore only examine errors between 0% and 2.5% thresholds [2]. Finally, this simulator will consider only

one type of error at any given time in order to isolate causation rather than precisely simulating a more realistic situation where many sources of error are present at the same time. To account for this, however, the code provided as an addendum to this report will contain functionality to test up to 32 combinations of the aforementioned error sources.

2.2 Procedure

To start each configuration which consists of an edit distance threshold and error threshold, the simulator will establish a baseline measurement. This measurement will take 100,000 samples from a data set of over 30 million samples, provided by the original authors of the Shouji Pre-Alignment Filter, and will compute a bit-vector with each bit representing one sequence with a 1 implying that that sequence was accepted and a 0 implying it was rejected. The baseline measurement will be constructed free of error and will be used for comparison later.

Next, for each subsequent error threshold, the simulator will modify each sample sequence with the associated type of error from section 2.0, and will then test the Shouji alignment against the reference sequence. This process will generate a new bit-vector which will be called the sample bit-vector.

Now that the simulator has constructed a baseline bit-vector and a sample bit-vector for a specific edit distance and error threshold parameter, it will proceed to compare the indexes against each other. Whenever a given index in the baseline bit-vector and the sample bit-vector does not match, this is considered to be an error. If that error is an accept on the baseline and a reject on the sample, it is a false-reject error. If that error is a reject on the baseline and an accept on the sample, it is a false-accept error. The total number of errors, false-accepts, and false-rejects will be tallied along with the total number of accepts and rejects in the sample bit-vector. Finally, once complete, the accuracy of each will be computed as follows:

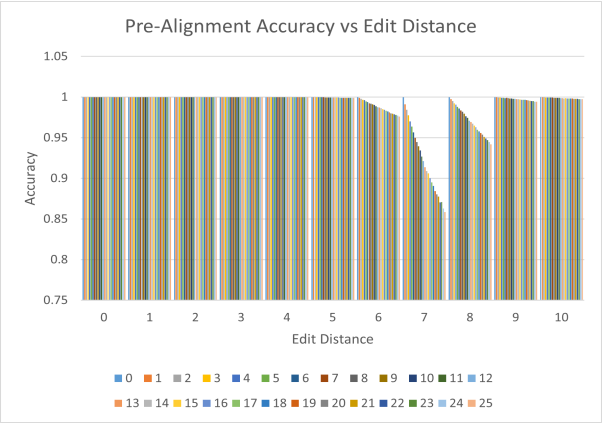
$$\begin{aligned} \text{Accuracy} &= \frac{\#\text{Errors}}{\text{Sequence Length}} \\ \text{False-Accept Rate} &= \frac{\#\text{False Accepts}}{\#\text{Accepts}} \\ \text{False-Reject Rate} &= \frac{\#\text{False Rejects}}{\#\text{Rejects}} \end{aligned}$$

Once these quantities are computed for every error threshold across every edit distance threshold (286 total iterations, each with 100,000 samples), the simulator will finish constructing a CSV file containing all of the information for use in Microsoft Excel to generate graphs.

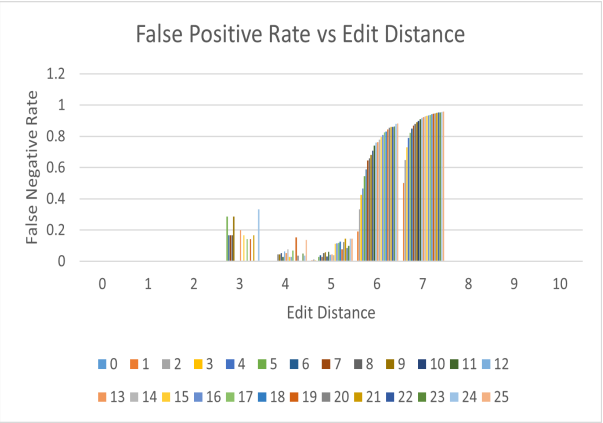
3 Results

The results of this research will be divided into categories related to each error source followed by a discussion of their associated results. At the end, there will be a comparative analysis between the different sources to assess which ones are the most destructive to the Shouji pre-alignment filter or which ones are not as large of an issue. The code used to generate these results can be found at this project's GitHub repository at
`https://github.com/andrew-j-millward/Shouji-Stress-Analysis` and the tests can be run using the command `./main 0 4 100 ../Datasets/ERR240727_1_E3_30million.txt 100000 3 -1` (replacing 3 with the desired test case 1-5).

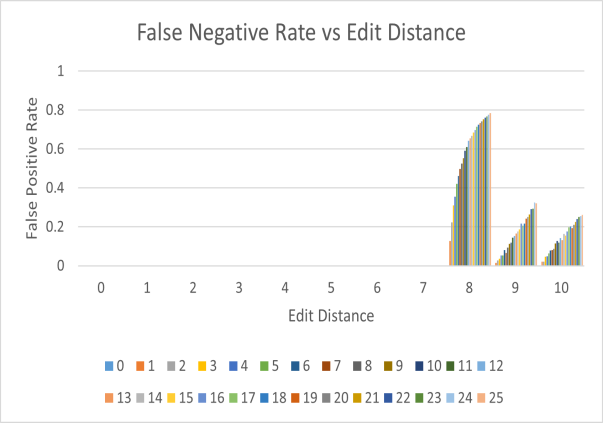
3.1 Substitution Errors



(a) Pre-Alignment accuracy at various thresholds



(b) False positive rate at various thresholds



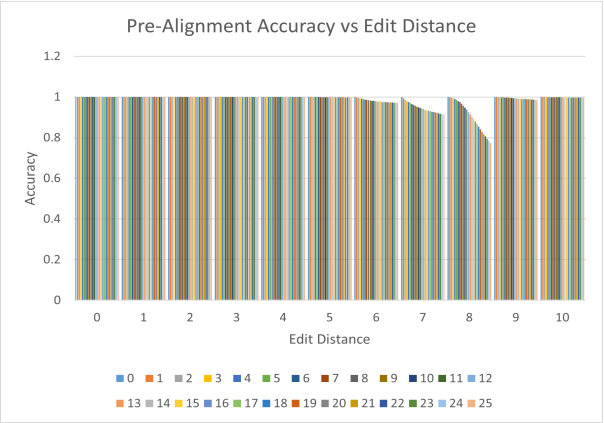
(c) False negative rate at various thresholds

(d) Alignments needed at various thresholds

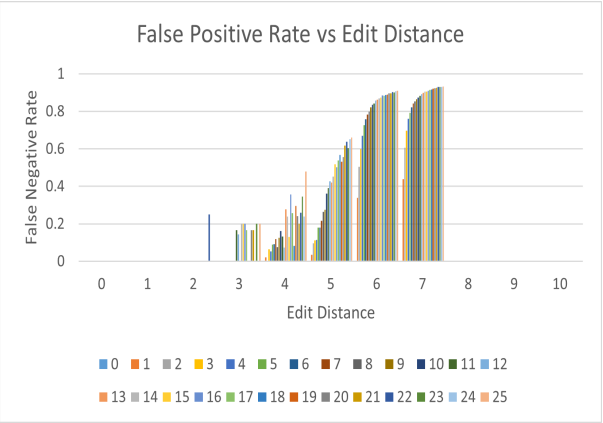
Figure 1: Substitution Error Graphs

Starting with Figure 1b, as the edit distance threshold is increased, the accuracy starts to dip. This becomes especially evident starting at edit distance 7 where accuracy drops as low as 86%. One can notice, however, that in edit distance 8, 9, and 10, the accuracy seemingly increases. The reason for this anomaly is actually entirely explainable with Figure 1d. Here, the total number of alignments needed rapidly increases starting at edit distance 8. As a result, when over 90% of the sequences are already being accepted anyway, the overall accuracy is harder to shift. As a case in point, Figure 1c illustrates the false negative rate of at different edit distances and error rates. None of the edit distances 0-7 experience any significant quantity (though some are possible) of false negatives prior to edit distance 8. Following this tipping point, however, a significant portion of those rejected strings actually become ones that should have been otherwise accepted. This ratio shift is actually due to the base Shouji pre-alignment filter which happens to begin accepting far more sequences starting with edit distance 8 on this particular data set. Finally, in Figure 1d, one can notice that at edit distance 7 in particular, the number of needed alignments increases as error rates go up. This is increase in alignments is due almost entirely to error rates increasing the false positive rates of distances 3-7 most severely in Figure 1b. Overall, substitution errors perform better than expected when evaluated using the Shouji algorithm. At lower edit distance thresholds, the errors remain largely manageable and have a low impact on performance.

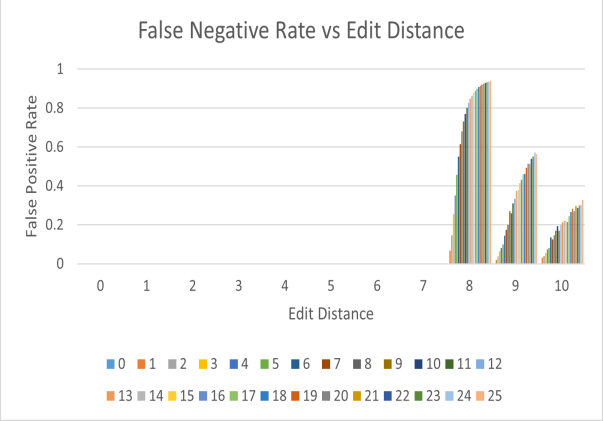
3.2 Insertion Errors



(a) Pre-Alignment accuracy at various thresholds



(b) False positive rate at various thresholds



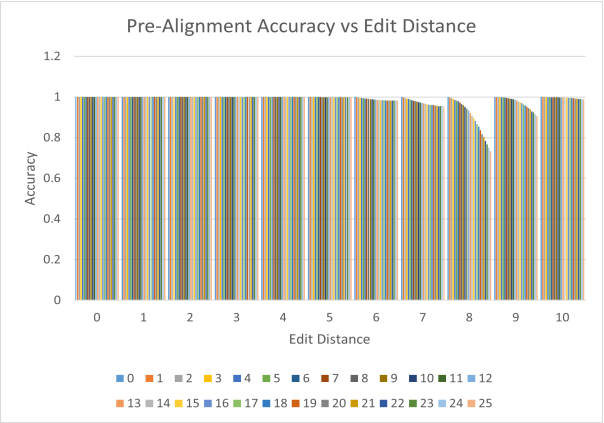
(c) False negative rate at various thresholds

(d) Alignments needed at various thresholds

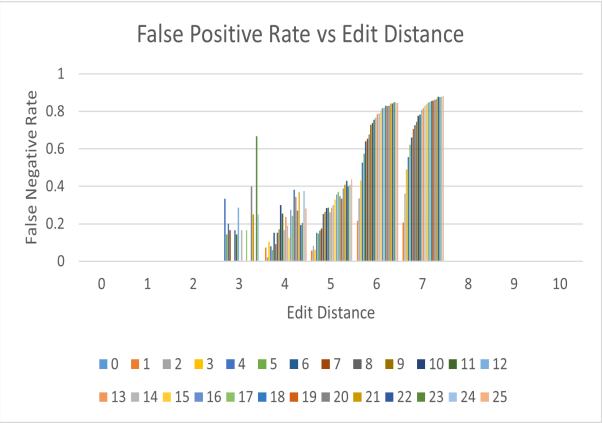
Figure 2: Insertion Error Graphs

Similar to substitution errors, insertion errors follow a similar pattern with some more extremes on some areas. In particular, the rate of false positives in Figure 2b follows a smoother curve for lower edit distances and error thresholds. This difference indicates that insertion errors are more likely to impact sequence pre-alignment using Shouji on lower edit distance thresholds than substitution errors. This particular point will be addressed more in section 3.6. Next, the same trend is visible in the false negatives in Figure 2c where an insignificant number of these values occur prior to edit distance 8 when again the needed alignments begin to approach the maximum as shown in Figure 2d. It is important to note, however, that the drop-off in needed alignments in Figure 2d is much more severe at an edit distance threshold of 8% than in the case of substitution errors. This increased drop-off rate is typically a good indicator of how much a particular error type impacts the effectiveness of the Shouji algorithm as a whole due to the fact that the needed alignments indicate how the percentage of a particular error impacts the number of accepted/rejected to the baseline which is the first column in every segment of the bar charts. Overall, however, insertion errors tend to hold up very well in lower edit distance levels with only minor impacts in efficacy at these locations. In higher error levels and higher edit distance thresholds, the differences become more severe, so intelligently adjusting the edit distance threshold to an optimal state could be used to mitigate the impact of read errors.

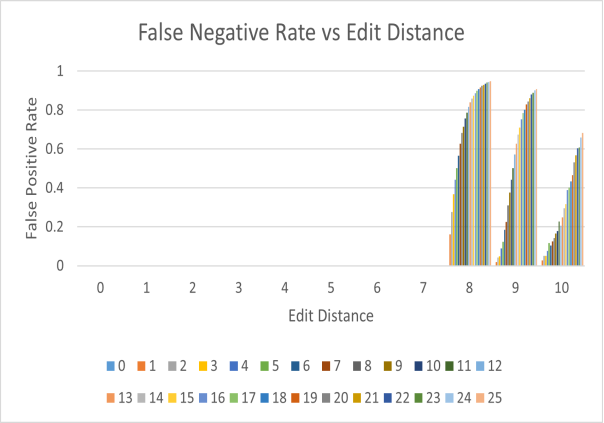
3.3 Deletion Errors



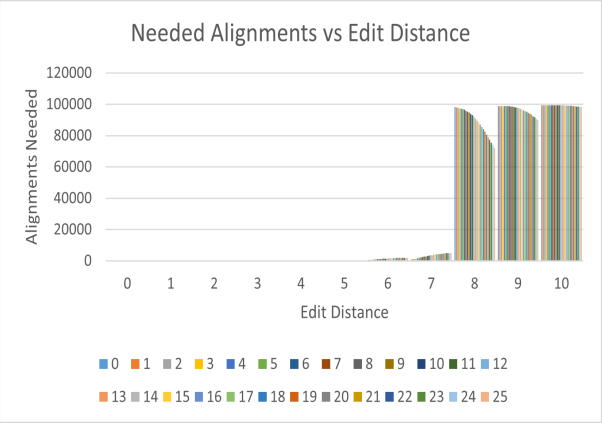
(a) Pre-Alignment accuracy at various thresholds



(b) False positive rate at various thresholds



(c) False negative rate at various thresholds

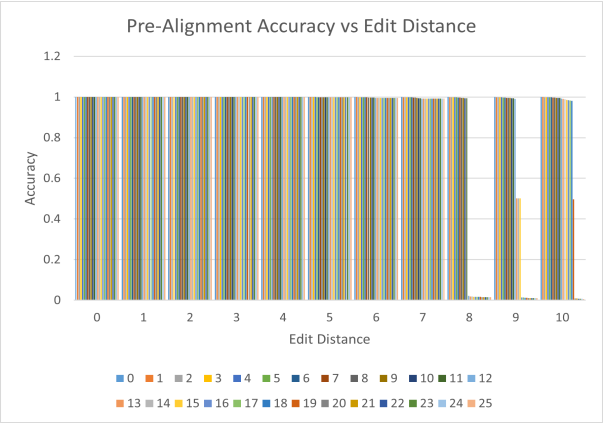


(d) Alignments needed at various thresholds

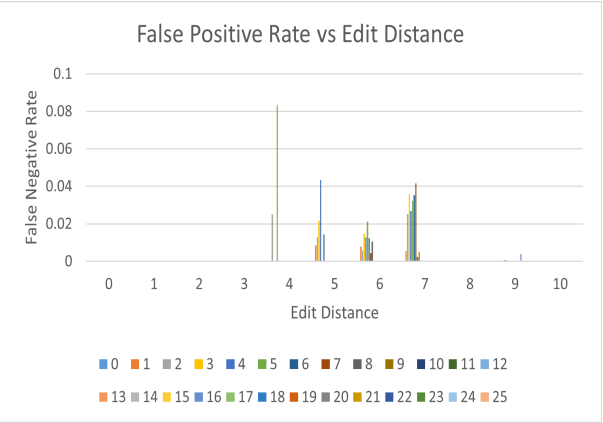
Figure 3: Deletion Error Graphs

Going into the experiment, deletion errors were predicted to be rather detrimental to the overall functionality of the algorithm due to the fact that not only do they shift a sub-sequence to the left, but they also substitute in gap characters at the end of the sequence which always yield a misaligned mark in the Shouji alignment grid. However, as with the last two, the performance impact is relatively low at smaller edit distance thresholds. With that said, some edit distances such as 3% and 4% appear to have some more noticeable spikes in their false positive rates as shown in Figure 3b. The false negative rates, on the other hand, land more in line with the experimental results obtained in section 3.2 related to insertion errors which is a good sign that false negative rates were not increased too substantially albeit appear possibly a little higher. Again, as a consequence of higher false positive and negative rates, the needed alignments appear to drop off slightly faster than in section 3.2 as can be seen in Figure 3d. The accuracy in Figure 3a is shown to remain relatively high throughout the execution with the exception of edit distance 8 dropping off the most similar to insertion errors but different from substitution errors which experienced the lowest accuracy at the edit distance of 7. As a whole, deletion errors appear to have the highest impact on accuracy out of the three primary read-based error sources, with much of the inaccuracy stemming from having the highest false negative rates of the three.

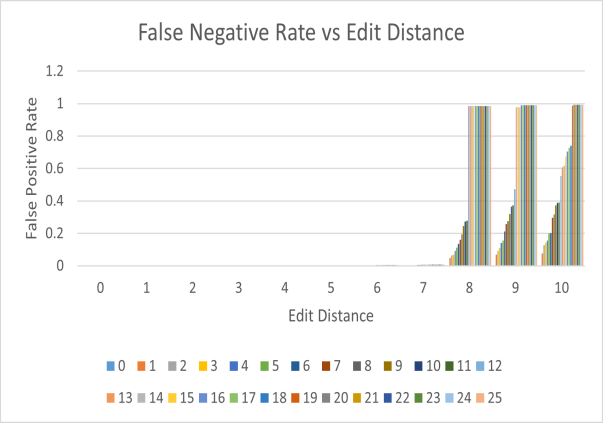
3.4 Size Mismatch Errors



(a) Pre-Alignment accuracy at various thresholds



(b) False positive rate at various thresholds



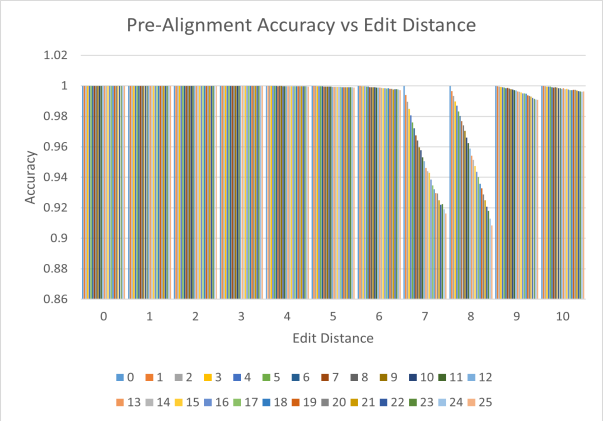
(c) False negative rate at various thresholds

(d) Alignments needed at various thresholds

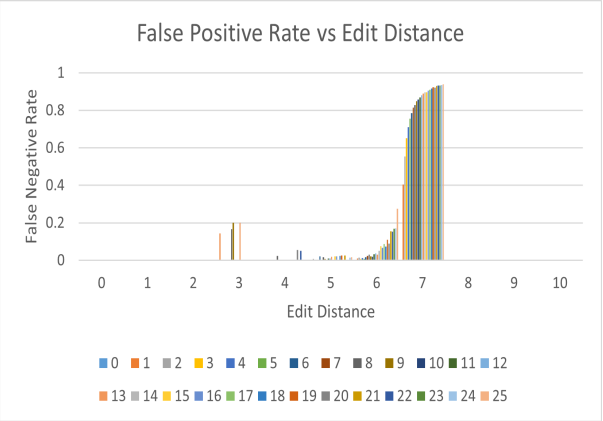
Figure 4: Size Mismatch Error Graphs

In the context of this experiment, size mismatch errors behave slightly differently from the other error sources. Most notably, this experiment varies sizes from 75bp to 100bp in size whereas the other error sources experience between a 0.0% and 2.5% error rate for any given base pair. As a result, some quantities such as needed alignments and alignment accuracy cannot be directly compared between size mismatch and the other error sources. With that said, the false positive and false negative rates are still possible to be compared as these quantities are normalized relative to the total number of each quantity, and the total size differential of size differentials falls within reasonable boundaries for real-world use as with the error boundaries on the other tests. As a result, one can see that unlike the other error sources, size differences in the sequences drastically reduces the total number of false positives that occur beyond the low difference levels. Unlike before, an increase in the size differential actually decreases the false positive rate in Figure 4b to near zero due to the fact that mismatched characters will more often disqualify strings from being aligned than allow them to be. As a result, the vast majority of the errors occur as false negatives shown in Figure 4c where the sequence with one of its ends chopped off will much more frequently reject otherwise correct sequences. As a result, there exists a massive drop-off in both accepted sequences and in accuracy in Figures 4d and 4a respectively that is almost entirely a consequence of an extremely high rate of false negatives. Therefore, one can conclude by looking at this data that the Shouji alignment algorithm almost always requires sequences to be of the same or a very similar size in order to function properly in order to minimize these much more dangerous false negative errors from occurring.

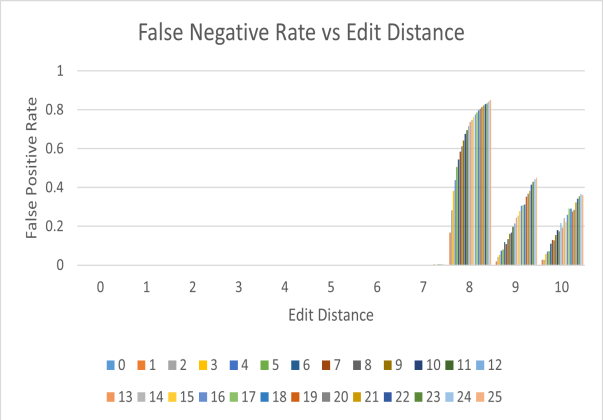
3.5 Gap Character Errors



(a) Pre-Alignment accuracy at various thresholds



(b) False positive rate at various thresholds



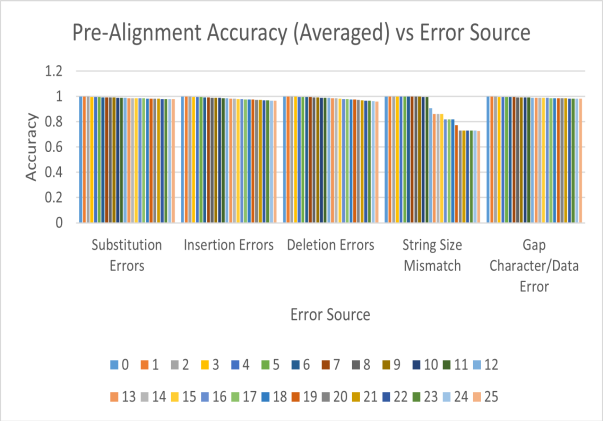
(c) False negative rate at various thresholds

(d) Alignments needed at various thresholds

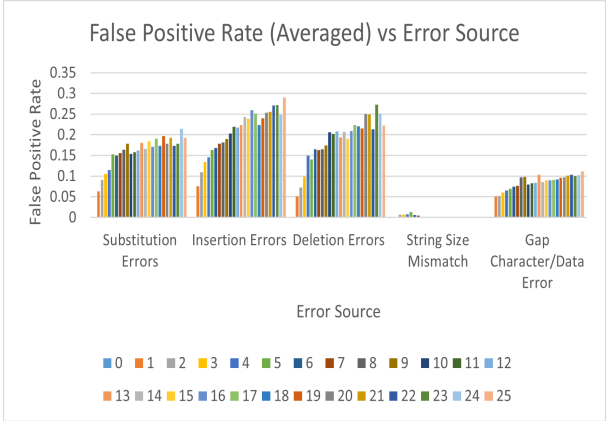
Figure 5: Gap Character Error Graphs

Finally, gap characters and data loss illustrate an unsurprisingly similar picture to that shown in section 3.1 with substitution errors. This point of similarity is mostly due to the fact that the data loss issue is defined almost identically to substitution errors with the exception that only gap characters 'X' are used instead of another random character. As a result, in diverse sequences, downstream alignment within the neighborhood map is possible with substitution errors which will yield potential alternative alignments, but this is not a possibility with gap character errors. Accordingly, the accuracy in Figure 5a does suffer a little at some higher thresholds such as edit distance 8 which experiences a higher rate of false negatives than its substitution error counterpart. It does, however, also result in relatively low false positive rates shown in Figure 5b across all edit distances except for 7 which is relatively unique to gap characters outsize of size mismatch. Altogether, data loss in the form of gap characters results in relatively low risk towards accuracy of the alignment algorithm at edit distance threshold 6 and lower, and, as a result, mitigates some of the risk if the algorithm is specified to use a lower edit distance threshold.

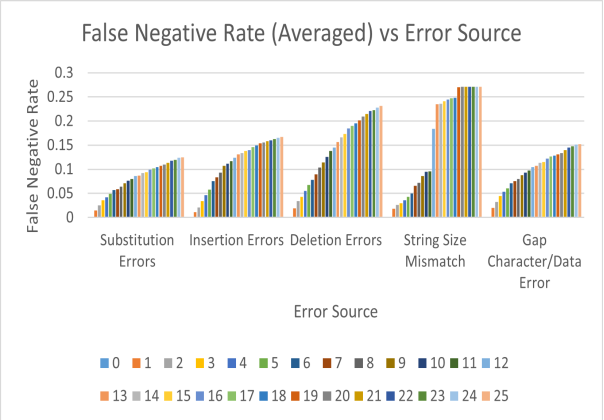
3.6 Comparing Error Sources



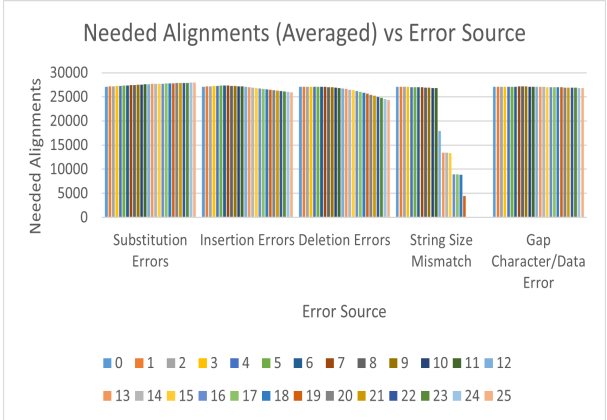
(a) Pre-Alignment accuracy at various thresholds



(b) False positive rate at various thresholds



(c) False negative rate at various thresholds



(d) Alignments needed at various thresholds

Figure 6: Comparing Sources of Error

Looking at a more holistic view of all of the data averaged across each edit distance threshold, one can begin to get a better picture of the relative impacts on accuracy, false positive/negative rates, and needed alignments. As mentioned earlier, the accuracy, shown in figure 6a, dips the most substantially with regards incorrectly matched sequence sizes where sequences of a smaller size than the reference sequence are utilized. As a result, one should not consider using Shouji if the base pair count of the sample and reference differ too greatly as with as little as 12bp difference, the accuracy of the algorithm will drop by 9% on average. Furthermore, examining Figure 6b in more detail, the rate of false positives impacts size mismatches the least and cascades up to impact insertion and deletion errors the most. Fortunately, false positives only increase the end-to-end execution time of the algorithm, so the increase may not necessarily cause major errors in the dataset that may impact the efficacy of the alignment if sufficient time is provided. On the other end of the spectrum, false negatives are the lowest in substitution errors, and increase in gaps, insertions, and deletions before reaching their highest level in string size mismatches in Figure 6c. As mentioned before, this is because any scheme that deletes some characters will automatically result in lower match rates as 'X' characters cannot be matched with any reference character. Finally, the most drastic difference can be seen in the needed alignment count for each method in Figure 6d. Due to the high level of false negatives in string size mismatches, some of the longest error source thresholds result in nearly all or sometimes all of the sequences being outright rejected. The other errors exhibit similar issues but not to the same extent with deletion errors only experiencing a minor dip in comparison.

4 Conclusions

From the results of this experiment, several major conclusions can be drawn about the nature of the error sources with regards to the efficacy of the Shouji pre-alignment filter algorithm. These conclusions and their explanations are as follows:

1. The ideal edit distance for a sequence lies between $4\% - 7\%$.
    a. As can be easily seen in figure 1a through 5a, the needed number of alignments rapidly increases after reaching the edit distance of 8%, and, for many of the error sources in 1b through 5d, the false positive rate is very high at 7% and sometimes 6%. This alone is not too large of an issue as the total number of alignments still remains very low, even with high levels of error. However, with that said, below the edit threshold of 4%, the needed alignments never exceed double digits. As a result, edit distances above 7% are said to be too inclusive and below 4% are said to be too restrictive. Therefore, this study has determined that an edit distance threshold of between $4\% - 7\%$ is sufficient to have adequate sequence diversity in case of potential read errors.
2. Strings should be of exactly the same size for ideal performance.
    a. This assumption was made by the original authors of the Shouji pre-alignment paper. However, in testing, this paper determined the extent to which this factor mattered. In the simulations, the size differential posed an extraordinary obstacle to successful alignments and high precision as they shift the neighborhood map entirely outside of the range of accessible values on the original diagonal. On the most extreme end of the spectrum, nearly every sequence was rejected as a result of mismatched sizing. Therefore, as a general rule, Shouji should only be used as the authors intended with strings of identical sizes.
3. Insertion errors generate the greatest false positive rate among similarly-sized reads.
    a. The false positive rate of insertion errors is roughly 50% greater than that of substitution errors, double that of gap character errors, and 20% greater than deletion errors.
4. Deletion errors generate the greatest false negative rate among similarly-sized reads.
    a. The false negative rate of deletion errors is roughly 60% greater than that of substitution errors, 60% greater than gap character errors, and double that of insertion errors.
5. Gap characters and data loss behaves similarly to substitution errors with a higher rate of rejection and lower rate of acceptance.
    a. Both of these exhibit their most major drops in accuracy at edit distance 7 and mechanically behave very similar. As a result, only the slight differences in false accept/reject differentiate the two.
6. The Shouji pre-alignment algorithm is overall very resilient to read-based errors and can be successfully used in such applications when sizes are the same.
    a. Although some error rates remain high, within reasonable error expectations of $0.24 \pm 0.06\%$ from Pfeiffer et al., the error remains relatively contained and accuracy hovers around 99.7% with only slight potential for false negatives above standard Shouji alignment [2]. As a whole, these false negatives likely pose little risk at the abundance they occur at in these experiments.

References

[1] Alser, M., Hassan, H., Kumar, A., Mutlu, O., and Alkan, C. Shouji: a fast and efficient pre-alignment filter for sequence alignment. *Bioinformatics 35*, 21 (03 2019), 4255–4263.

[2] Pfeiffer, F., Gröber, C., Blank, M., Händler, K., Beyer, M., Schultze, J. L., and Mayer, G. Systematic evaluation of error rates and causes in short samples in next-generation sequencing. *Scientific Reports 8*, 1 (Jul 2018), 10950.