

3

As motifs are expanded they are continually re-evaluated. Their scores are determined using a scoring function and ranked from highest to lowest before being subjected to selection by the BEAM algorithm as shown in **Figure 2**. The scoring function measures the degree of over-representation for each motif in the set of genes, but can be changed to measure other properties such as preferred motif position.

Figure 3 shows that motif scores increase until the motif reaches full length, at which point more letters

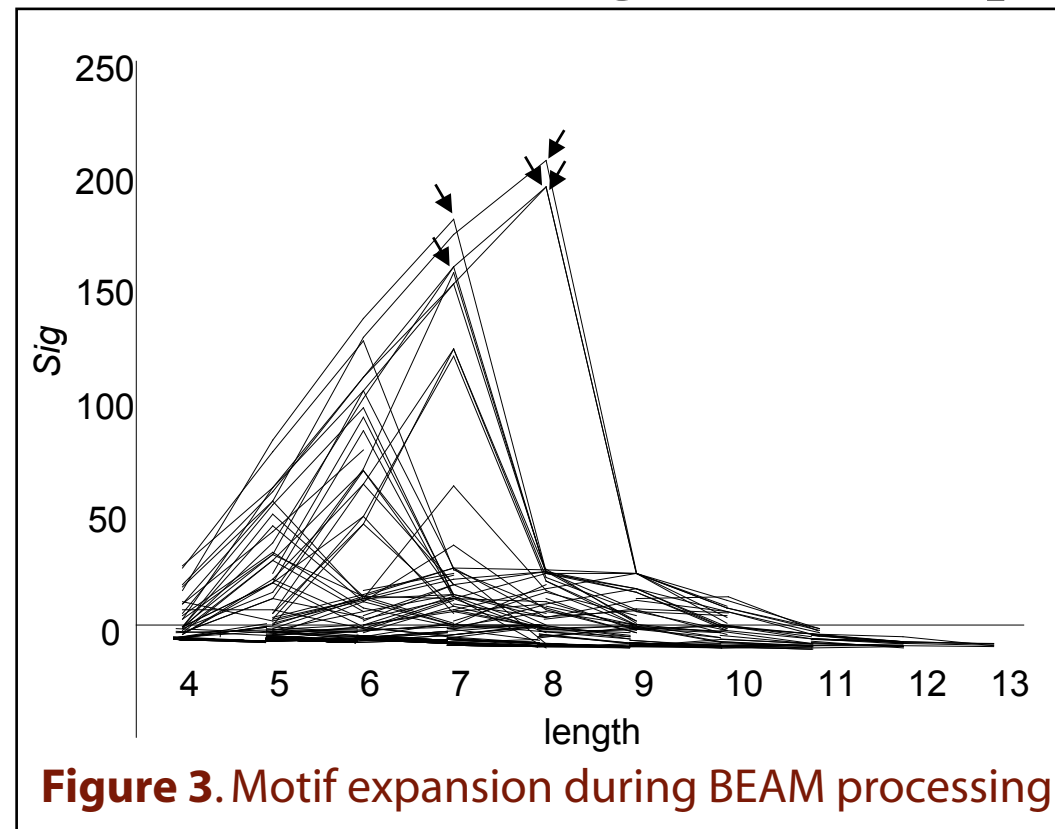


Figure 3. Motif expansion during BEAM processing

cause a drop in score. This allows BEAM to find motifs of any length. **Figure 4** shows the fraction of motifs that were in the top b (beam width) motifs of the prior round. Increasing beam width increases the fraction surviving from round to round, but also introduces noise. More than 90% of the top 100 motifs in any round end up in the top 1000 in the next round. In other words, high scoring motifs are not lost in subsequent rounds.

cause a drop in score. This allows BEAM to find motifs of any length.

Figure 4 shows the fraction of motifs that were in the top b (beam width)

We measured the error rate for BEAM as we varied the beam width as shown in **Figure 5**.

The number of false positives (FP) and false negatives (FN) decreases with increasing beam widths until it plateaus at 1000, the default value used by BEAM. Interestingly,

enumerating all motifs (FULL) leads to a higher false positive rate because unique long motifs occur by chance in front of any given gene set

that do not grow from shorter motifs using the iterative beam process of our algorithm.

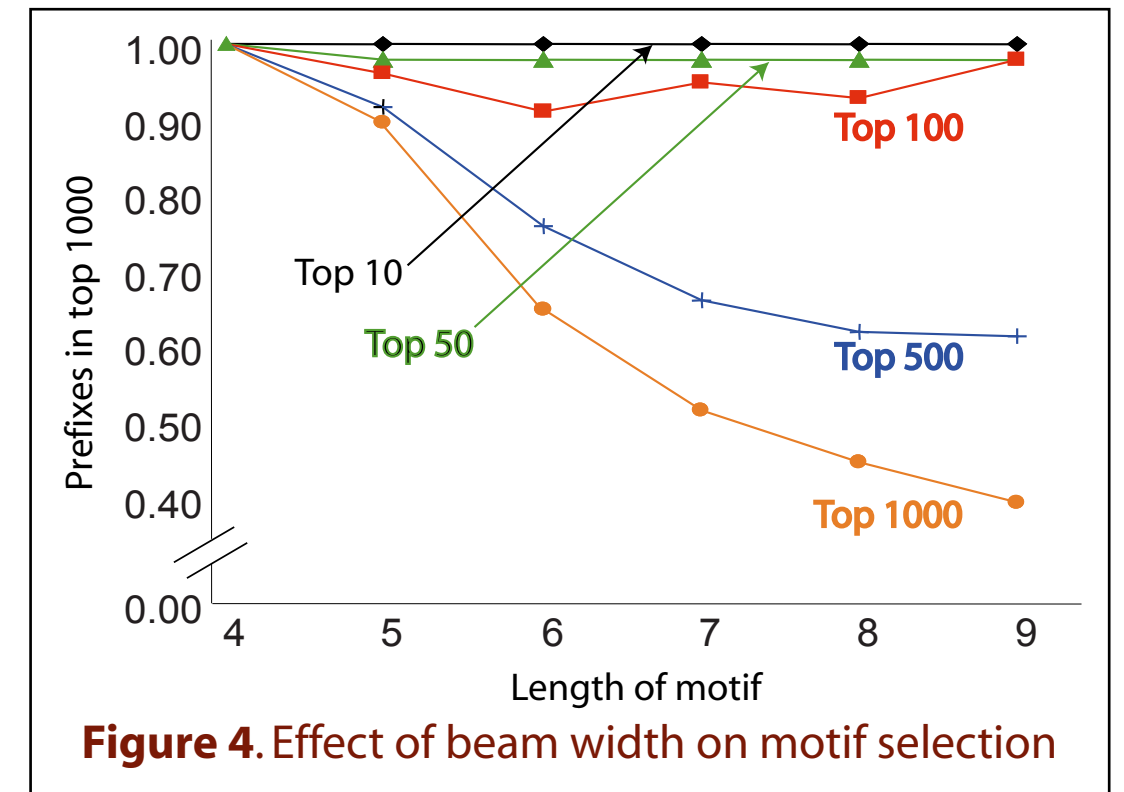


Figure 4. Effect of beam width on motif selection

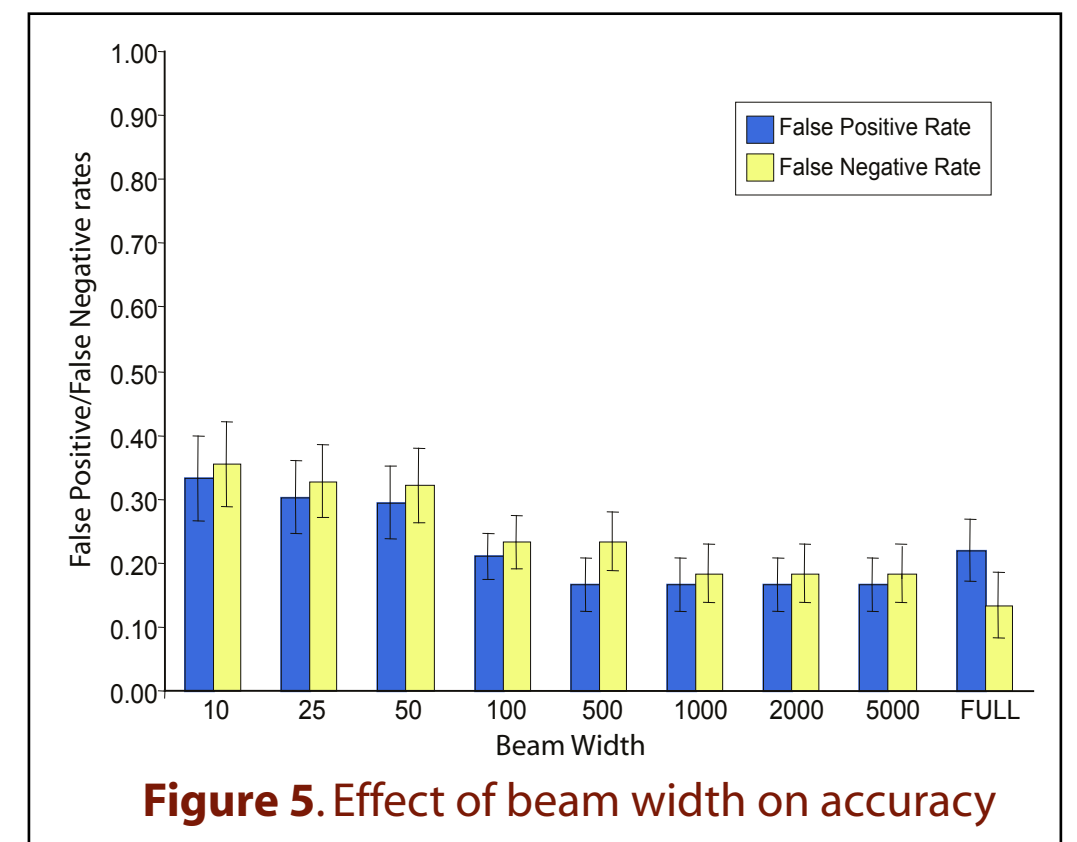


Figure 5. Effect of beam width on accuracy