Cognitively Inspired Sampled String Matching

Simone Faro[1], Francesco Pio Marino[1,2], and Arianna Pavone[3]

[1] Department of Mathematics and Computer Science, University of Catania
[2] Univ Rouen Normandie, INSA Rouen Normandie, Université Le Havre Normandie, Normandie Univ, LITIS UR 4108, CNRS NormaSTIC FR 3638, IRIB
[3] Department of Mathematics and Computer Science, University of Palermo

Abstract. Human reading is characterized by selective visual attention and efficient text processing, often involving perceptual skipping and sparse sampling of input. These phenomena suggest that human cognition naturally employs strategies for reducing information load while preserving recognition accuracy. Motivated by this observation, this research paper investigates a class of online string matching algorithms based on text sampling, wherein comparisons are restricted to strategically selected positions or characters chosen by structural properties of the pattern. We examine the extent to which such sampling heuristics align with human reading behaviors as observed in eye-tracking data. By conducting a comparative experimental analysis between algorithmic matching processes and human visual search trajectories, we demonstrate that computationally optimal sampling strategies often reflect regularities found in biological perception. These findings support a broader interdisciplinary framework in which cognitively plausible models can inform algorithm design and, conversely, algorithmic efficiency principles may yield insights into perceptual mechanisms.

Keywords: string matching · human reading · sampling

1 Introduction

String matching is a fundamental problem in computer science [5,25], with applications spanning diverse fields such as natural language processing, information retrieval, and computational biology. The goal is to locate all occurrences of a given pattern x of length m within a text y of length n, where both sequences consist of characters from an alphabet Σ of size σ. Over the years, two primary paradigms have been developed to address this problem: online and offline string matching. Online algorithms, such as the Knuth-Morris-Pratt algorithm [22], process the text in real time, achieving a worst-case time complexity of $O(n)$, while heuristics like the Boyer-Moore algorithm [3] enhance practical performance by efficiently skipping portions of the text. In contrast, offline approaches preprocess the text to construct an index, enabling rapid query resolution. Prominent examples include suffix trees [1], which offer an $O(m + occ)$ worst-case time, suffix arrays [24] with a time complexity of $O(m + \log n + occ)$,

where *occ* it the number of occurrences of the searched pattern, and the FM-index [16], a compressed structure derived from the Burrows-Wheeler transform that combines input compression with efficient substring queries. However, these full-indexes require additional storage space, ranging from four to twenty times the size of the text size.

To mitigate this space complexity, sampled string matching has emerged as an alternative approach, first introduced by Vishkin [38]. Instead of constructing a full index, sampled string matching builds a reduced representation of the text by selectively storing sampled portions, significantly reducing memory usage while preserving search efficiency. It has been demonstrated that once a sampled index is constructed, any online string matching algorithm can be applied directly to it, requiring only a small verification phase [10]. This hybrid approach balances memory efficiency with computational performance, making it an attractive alternative to traditional full-index methods.

Beyond Vishkin's theoretical approach, a particularly successful advancement in this area is the Character Distance Sampling (CDS) technique [12], which records distances between occurrences of selected pivot characters instead of storing absolute positions. This method, initially designed for online string matching, achieves remarkable search efficiency while maintaining a compact partial index, often requiring as little as 5% of the original text size.

CDS has also been successfully extended to offline string matching [9,14], demonstrating a search time improvement of up to 91% compared to standard indexed approaches, while using less than 15% of the space required for a full suffix array. These improvements make CDS a promising technique for large-scale text processing, particularly when computational resources are constrained. The code of the preprocessing phase is detailes in Fig. 1.

```
COMPUTE-DISTANCE-SAMPLING(y, n, C)
1.  ȳ ← ⟨⟩
2.  j ← 0
3.  p ← 0
4.  for i ← 1 to n do
5.      if y[i] ∈ C then
7.          ȳ[j] ← i − p
6.          j ← j + 1
8.          p ← i
9.  return (ȳ, j)
```

```
COMPUTE-POSITION-SAMPLING(y, n, C, k)
1.  ẏ ← ⟨⟩
4.  j ← 0
5.  for i ← 1 to n do
8.      if y[i] ∈ C then
10.         ẏ[j] ← i
9.          j ← j + 1
12. return (ẏ,j)
```

Fig. 1. (On the left) The pseudocode of procedure COMPUTE-DISTANCE-SAMPLING for the construction of the *character distance sampling* version of a text y. (On the right) The pseudocode of procedure COMPUTE-POSITION-SAMPLING for the construction of the *character position sampling* version of a text y.

Despite the advantages of sampling algorithms, both techniques have been shown to be ineffective for small alphabets, such as those found in genomic sequences. To overcome this limitation, Faro *et al.* [13] introduced a novel approach leveraging q-grams to artificially expand the alphabet size, thereby enabling the use of sampling-based indexing methods even for small-alphabet domains. This enhancement allows sampled string matching to be applied in bioinformatics and other fields where the underlying alphabet is inherently small.

Although Character Distance Sampling has demonstrated its efficiency, it relies on a distance representation that requires multiple bytes, making it challenging to store compactly. To address this, a new space-efficient decomposition known as Fake Decomposition was introduced [11]. This technique allows for a significant reduction in the space required to store sampled indexes without compromising search accuracy.

In theory, bounding all distances by 256 would allow storage within a single byte. However, this constraint is often impractical in real-world scenarios. Figure 2 presents the greatest and average distances for each character, sorted by rank, in an English text alphabet.

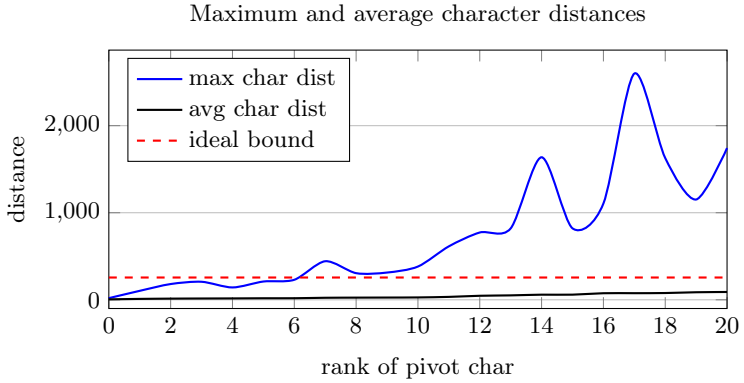

Fig. 2. Maximum and average distances between two consecutive occurrences, computed for the most frequent characters in a natural language text. On the x axis characters are ordered on the base of their rank value, in a non decreasing order. The red line represents the ideal bound 256.

Recently, new sampling representations have been developed to extend sampled string matching beyond traditional exact matching. Monotonic Run-Length Scaling (MRLX) and Monotonic Run-Length Sampling (MRLS) [15] have been proposed to handle non-classical string matching tasks, particularly Order Preserving Pattern Matching (OPPM) [21] and Cartesian Tree Pattern Matching (CTPM) [27]. These methods leverage run-length encoding principles to repre-

sent sampled patterns more effectively, improving performance in structured and numerical data matching applications.

An additional advancement in sampled string matching is Character Context Sampling (CCS), an extension of CDS that refines the sampling process by incorporating the surrounding context of sampled characters [8]. Unlike CDS, which records distances between selected pivot characters, CCS enhances sampling by considering local character distributions and their relationships within a defined window. By capturing richer contextual information, CCS reduces false candidate occurrences while maintaining a compact index, making it particularly effective in cases where traditional sampling techniques struggle due to redundancy or skewed distributions. Experimental results demonstrate that CCS achieves superior search efficiency while preserving a significantly smaller index size, further optimizing sampled string matching techniques.

Beyond pattern searching, sampled string matching techniques, such as Character Distance Sampling, have also proven useful for structural analysis of strings [23]. In particular, CDS has been adapted for computing string periodicity and shortest covers, fundamental tasks in text processing. The sampled representation enables significantly faster computations compared to classical methods, making it an efficient approach for detecting regularities in large-scale datasets.

Recently, a new optimal representation for sampling has been introduced, leveraging a Set Cover [37] approach to refine the selection of pivot characters and enhance the sampled text representation [7]. This advancement addresses key challenges, particularly optimizing the trade-off between index size and search efficiency, while mitigating worst-case scenarios where patterns lack pivot occurrences. This solution explores how this new model improves upon existing sampling strategies, further solidifying the role of sampled string matching as a space-efficient and computationally effective solution for large-scale text searching.

While these algorithmic developments have primarily been driven by computational considerations, there is growing interest in understanding whether such sampling strategies mirror processes in human cognition. In particular, research in cognitive science has shown that human readers do not examine all characters or words sequentially; rather, they employ selective attention and perceptual skipping during reading, focusing only on salient textual regions [26]. This behavior can be understood as a biologically grounded sampling mechanism aimed at optimizing recognition speed and accuracy under resource constraints.

The present work aims to bridge these domains by examining whether algorithmically optimal sampling strategies—such as those used in character-distance-based string matching—reflect or approximate the sampling behavior of human readers. To this end, we conduct a comparative experimental study using controlled visual search tasks with eye-tracking data, contrasting human scanpaths with the sampling paths of string matching algorithms. The results offer new insights into the alignment between artificial and natural strategies of information processing and open the door to cognitively informed models for efficient string analysis.

2 Characters Distance Sampling in Brief

The *Character Distance Sampling* (CDS) technique builds a compact partial index by recording distances between occurrences of selected pivot characters [12]. Formally, given a text y of length n and a pattern x of length m over an alphabet Σ, a sub-alphabet $C \subseteq \Sigma$ is chosen as pivot set. If $\delta(i)$ denotes the position of the i-th occurrence of any $c \in C$, the position-sampled version of y is $\dot{y} = \langle \delta(1), \ldots, \delta(n_c) \rangle$, while the character-distance sampled version is $\bar{y} = \langle \delta(2) - \delta(1), \ldots, \delta(n_c) - \delta(n_c - 1) \rangle$.

Example 1. For $y =$ "agaacgcagtata" and $C = \{\mathrm{a}\}$, one obtains $\dot{y} = \langle 1, 3, 4, 8, 11, 13 \rangle$ and $\bar{y} = \langle 2, 1, 4, 3, 2 \rangle$.

To search a pattern x, its sampled version $\bar{x}$ is computed. Every occurrence of x in y corresponds to an occurrence of $\bar{x}$ in $\bar{y}$, though the reverse requires validation in $O(m)$ time. In practice, $\dot{y}$ is stored and $\bar{y}$ computed on the fly, yielding a partial index of size $32n_c$ bits. This enables up to $40\times$ speed-ups over standard online algorithms at only $\sim 2\%$ of the text size [12].

Beyond exact matching, CDS has been extended to approximate searches (e.g., run-length text sampling for *Order Preserving Pattern Matching* [15,21]), and offers flexibility in handling dynamic texts. Recent work further improves space/time trade-offs via condensed alphabets [13] and fake distance representation [11].

While effective, CDS performance depends on pivot choice (in English, the 8th most frequent character is often optimal). For very short patterns without pivots or small alphabets, classical search or alphabet condensation may be preferable [13].

3 Human Sampling Behavior in Reading

Human reading is a complex cognitive process that relies on the coordinated interplay of selective attention, efficient eye movements, and a predictive understanding of linguistic structure. Contrary to the intuitive idea of serial, exhaustive character-by-character scanning, a substantial body of evidence from cognitive psychology, neuroscience, and psycholinguistics shows that readers sample text sparsely and strategically, employing a process known as *perceptual skipping* [29,33]. This form of selective processing is foundational for understanding how humans achieve rapid and efficient reading despite the brain's limited perceptual and attentional bandwidth.

In practice, reading is less about decoding every visual element and more about selectively targeting high-value segments of text for detailed inspection, while relying on peripheral cues, prior knowledge, and linguistic prediction to fill in the gaps. These selective strategies provide an interesting analogue for algorithm design: just as humans ignore low-informative regions of text, efficient string matching algorithms can strategically skip positions that are unlikely to yield matches.

3.1 Fundamentals of Visual Attention in Reading

Visual perception during reading is governed by the mechanics of the human visual system and the predictive nature of linguistic processing. The eyes execute rapid ballistic movements, called *saccades*, typically lasting 20–50 ms, interspersed with *fixations* lasting approximately 200–250 ms, during which most information is extracted [29]. The *perceptual span*—the region from which useful visual information can be acquired in a single fixation—extends asymmetrically in left-to-right reading systems: about 3–4 characters to the left and 14–15 to the right of fixation [30,31].

Only the foveal region (about 2 degrees of visual angle) supports high-resolution processing, while the parafoveal and peripheral regions provide lower-resolution, pre-attentive cues. These spatial constraints encourage readers to make decisions about where to fixate next based on a trade-off between expected informational value and the cost of moving the eyes. As a result, low-information or predictable words are often skipped, leading to a naturally sparse and efficient sampling trajectory.

3.2 Perceptual Skipping and Linguistic Predictability

Perceptual skipping is strongly modulated by linguistic predictability. High-frequency words and syntactically predictable items are more likely to be skipped, whereas rare or ambiguous words attract longer fixations and regressions [35,20]. The E-Z Reader model [32] proposes that lexical processing occurs in parallel with saccadic programming: as soon as a word's familiarity reaches a threshold, the system can begin programming the next eye movement without completing full lexical access, enabling the reader to leapfrog over predictable content.

From an information-theoretic perspective, this can be framed as an entropy-reduction strategy: regions of low entropy (low informational gain) are bypassed, while those of high entropy attract attention. This mirrors sampling in algorithms, where the goal is to select positions with high discriminatory power to minimize the number of comparisons needed.

3.3 Top-Down and Bottom-Up Integration

Eye guidance in reading results from the integration of top-down and bottom-up processes [19]. Top-down mechanisms include syntactic expectations, semantic context, and discourse-level goals, which allow readers to anticipate upcoming material. Bottom-up mechanisms respond rapidly to local stimulus properties such as word length, letter frequency, and visual contrast.

In algorithmic terms, top-down processes resemble *pattern-based heuristics*, while bottom-up processes correspond to *character-level frequency analysis*. For example, just as a sampling algorithm might prioritize low-frequency pivot characters to maximize filtering efficiency, a reader may fixate on low-predictability words to resolve uncertainty.

3.4 Scanpath Regularities and Eye-Movement Corpus Studies

Large-scale corpora such as GECO and MECO provide detailed records of reading behavior across languages and populations [36,4]. Statistical analyses reveal several robust scanpath regularities:

- Fixations cluster near syntactic boundaries (e.g., noun phrases, clause breaks).
- Skipped words are typically short (< 5 characters), frequent, and syntactically predictable.
- Regression saccades (backward eye movements) are more common in regions of syntactic or semantic difficulty.

These findings suggest an implicit strategy for prioritizing information, closely paralleling index construction in sampled string matching, where structurally salient characters are selected to optimize match verification.

3.5 Theoretical Models of Cognitive Economy

The concept of *cognitive economy*—maximizing information gained while minimizing processing cost—is a core principle in models of visual attention. According to the principle of bounded rationality [34], readers use heuristics that yield sufficiently accurate comprehension without incurring the cost of exhaustive processing. Entropy-based models formalize this by linking fixation placement to regions of maximum expected information gain, a principle directly analogous to sampling algorithms that select comparison points of highest utility.

3.6 Neurocognitive Evidence and Brain-Inspired Models

Neuroimaging studies identify distributed cortical networks, including the superior parietal lobule, frontal eye fields, and occipito-temporal regions, as critical to the control of visual attention during reading [28]. These areas interact with the visual cortex to implement selective attention and gaze control, and their activity supports predictive models of incoming input.

Computational neuroscience perspectives such as predictive coding [18] frame the brain as a Bayesian inference machine, continually updating predictions about the sensory environment and using discrepancies to guide further sampling. This closely parallels adaptive string matching, where expectations about the pattern structure guide pivot selection and skipping heuristics.

3.7 Cognitive Inspiration for Algorithmic Sampling

The parallels between human reading and algorithmic string matching suggest that incorporating cognitive principles could yield algorithms that are both efficient and robust. Core analogies include:

- *Fixation as Pivot Selection*: Just as readers choose fixation points that are most informative for comprehension, algorithms can select pivot characters that best discriminate between matches and mismatches.

- *Skipping as Window Shifting*: Perceptual skipping is analogous to heuristic jumps in algorithms like Boyer–Moore or CDS, where non-promising regions are bypassed.
- *Contextual Expectation as Preprocessing*: Similar to how readers use parafoveal and contextual cues to anticipate upcoming words, CCS algorithms incorporate surrounding character windows to refine candidate selection.

By embedding these principles explicitly, string matching algorithms may achieve better trade-offs between accuracy, speed, and memory, especially in noisy, redundant, or semi-structured data environments.

4 Experimental Results

We conducted a comprehensive evaluation of the cognitively inspired string matching techniques proposed in this work. The primary focus is on *search time efficiency* during candidate verification, while maintaining equivalent index sizes across all methods to ensure fair comparisons.

4.1 Algorithms and Implementation

We implemented and benchmarked the state-of-the-art *Character Distance Sampling* (CDS) algorithm, incorporating two different pivot selection strategies:

- *Algorithmic baseline:* the best-known pivot selection strategy described in [7].
- *Cognitively inspired pivoting:* pivots derived from human fixation data, computed over large-scale eye-tracking corpora.

To integrate the cognitive dimension, we considered two complementary eye-tracking resources:

- GECO [4], an English/Dutch bilingual reading corpus with over 500,000 recorded fixations, providing the core distribution of fixations across characters in words.
- CELER (Augmented GECO) [2], a large-scale dataset of L1/L2 English reading including 365 participants, which we leverage to generalize and augment GECO fixation patterns with broader demographic coverage.

4.2 Pivot Generation from Fixations

Let a word w consist of L characters, indexed from left to right as

$$w = (c_1, c_2, \ldots, c_L).$$

Eye-tracking corpora such as GECO and CELER provide fixation data, i.e., counts of how often participants' gaze landed on or near each character position. From these raw counts, we compute the empirical fixation probability distribution for word length L as follows.

Let N_i denote the number of fixations observed on character c_i, and $N = \sum_{j=1}^{L} N_j$ the total number of fixations for the word across participants. The normalized fixation probability is then

$$f_i = \frac{N_i}{N}, \qquad i = 1, \ldots, L,$$

so that $\sum_{i=1}^{L} f_i = 1$.

Pivot Expectation. The cognitively inspired pivot is defined as the expectation of this discrete probability distribution:

$$\mu_w = \mathbb{E}[i] = \sum_{i=1}^{L} i \cdot f_i.$$

Since the pivot must correspond to a discrete character position, we map the expectation to the nearest integer index:

$$\mathrm{PVL}(w) = \mathrm{round}(\mu_w).$$

Thus, $\mathrm{PVL}(w)$ identifies the character in w most representative of human visual attention according to fixation statistics.

Corpus Augmentation. In our study, GECO serves as the base corpus providing fine-grained fixation probabilities. CELER is used as an augmentation source, allowing us to smooth and generalize the empirical distribution by pooling across a much larger participant base. Concretely, given two distributions $f_i^{(\mathrm{GECO})}$ and $f_i^{(\mathrm{CELER})}$ for a given word length L, we define the augmented fixation distribution as a convex combination:

$$f_i^{(\mathrm{Aug})} = \lambda\, f_i^{(\mathrm{GECO})} + (1 - \lambda)\, f_i^{(\mathrm{CELER})},$$

with $\lambda \in [0, 1]$ controlling the relative weight. In our experiments we considered $\lambda = 0.5$ unless otherwise noted, yielding a balanced augmentation.

Example 2 (Pivot generation for "intense"). To illustrate the procedure, consider the word $w =$ *intense* of length $L = 7$, i.e. $w = (c_1, \ldots, c_7)$. From the GECO corpus we observe fixation counts $N = \{4, 18, 46, 64, 46, 18, 4\}$ with $\sum_{i=1}^{7} N_i = 200$, yielding normalized probabilities

$$f_i^{\mathrm{GECO}} = \frac{N_i}{200}, \qquad i = 1, \ldots, 7.$$

The expectation of the distribution is

$$\mu_w^{\mathrm{GECO}} = \sum_{i=1}^{7} i \cdot f_i^{\mathrm{GECO}} = 4.00,$$

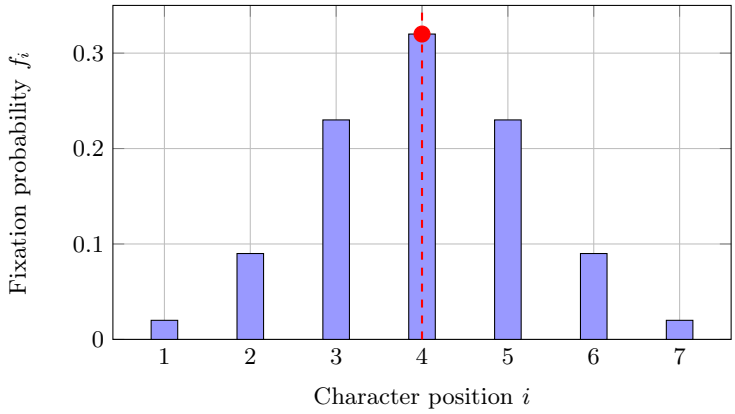

Fig. 3. Fixation probabilities f_i for the word *intense* ($L = 7$). Bars show GECO-derived probabilities; the dashed line marks the expectation $\mu_w = 4.00$ and the red dot highlights the discrete pivot $\mathrm{PVL}(w) = 4$.

and the pivot position is obtained as $\mathrm{PVL}_{\mathrm{GECO}}(w) = \mathrm{round}(\mu_w^{\mathrm{GECO}}) = 4$, corresponding to the central character "e".

When augmenting GECO with CELER, we form the convex combination

$$f_i^{(\mathrm{Aug})} = \lambda f_i^{\mathrm{GECO}} + (1-\lambda) f_i^{\mathrm{CELER}}, \quad \lambda = 0.5,$$

which smooths the distribution across a larger participant base. The resulting expectation is $\mu_w^{(\mathrm{Aug})} = 3.98$, leading to the same discrete pivot $\mathrm{PVL}_{\mathrm{Aug}}(w) = 4$.

Figure 3 illustrates this example: bars represent fixation probabilities f_i, the dashed line indicates the expectation μ_w, and the red point highlights the selected pivot. Both GECO and the augmented distribution thus converge on the central vowel as the cognitively inspired pivot, showing consistency across corpora.

4.3 Datasets and Experimental Setup

All implementations were developed in `C` and evaluated using the `Smart` benchmarking framework [6], compiled with `gcc` under the `-O3` optimization flag to enable aggressive optimizations such as loop unrolling and inlining. To ensure that results reflect algorithmic efficiency rather than system-level artifacts, all experiments were executed in a single-threaded environment with fixed random seeds for reproducibility.

As input text we used a 100 MB English segment from the *Pizza & Chili* corpus [17], which has become a standard benchmark in the field due to its large size and linguistic representativeness. The text was preprocessed by removing control symbols and lowercasing all characters, producing a consistent stream of tokens while maintaining natural word frequency distributions. Patterns were

randomly extracted from this text with lengths $m = 2^p$ for $p \in \{4, 5, 6, 7\}$, corresponding to $m \in \{16, 32, 64, 128\}$. For each value of m, we selected 1000 distinct patterns via uniform random sampling over all substrings of length m, a procedure designed to balance frequent and rare contexts while ensuring fair comparisons across algorithms. All algorithms were then run on the same pattern set, so that performance differences can be attributed solely to the pivot selection strategy.

Each configuration was executed in 100 independent runs, and results were averaged to obtain stable measurements. The evaluation criterion was strictly *search time efficiency*, providing a direct comparison of candidate verification costs across pivot selection strategies.

The experiments were carried out on a MacBook Pro equipped with a 2.7 GHz Intel Core i7 processor, four cores, 16 GB of 2133 MHz LPDDR3 RAM, a 256 KB L2 cache, and an 8 MB L3 cache. No background processes were allowed during benchmarking, ensuring stable performance across all runs.

4.4 Performance Results

Figure 4 reports the runtime comparisons across different pattern lengths and numbers of pivots. The CDS variant using the best-known algorithmic pivot selection consistently achieved the fastest search times, demonstrating the efficiency of mathematically optimized strategies in a computational setting. In contrast, cognitively inspired pivot choices—while grounded in human fixation behavior—produced slower searches, suggesting that human reading patterns do not directly translate into optimal algorithmic performance.

As visible across all configurations, the *Human* alignment entails a consistent runtime penalty compared to the *Best* strategy. On average, the slowdown ranges between 15% and 30%, with the largest deviations observed for intermediate pattern lengths ($m = 32$ and $m = 64$), while the gap narrows to around 10–15% for very short or very long patterns. This suggests that fixation-driven pivots approximate but do not fully capture the optimal distribution required for efficient sampling, thereby highlighting a trade-off between cognitive plausibility and computational efficiency.

However, while the previous analysis focused on pivot generation at the single-word level, practical deployment requires extending the procedure to entire texts. To this end, we extract the pivot $\mathrm{PVL}(w)$ for each word w in the corpus, thereby producing a sequence of pivot positions across the vocabulary. We then compute the empirical frequency distribution of these pivots, identifying which character indices occur most often as fixation-driven anchors. Finally, we select the top-K most frequent pivots as the representative set for downstream sampling. This procedure ensures that cognitively plausible pivot information is condensed into a compact and reusable form, bridging the gap between word-level fixation statistics and efficient large-scale text processing.

Although we also varied the regularization weight λ, the fixation maps produced by GECO and CELER remained largely similar under our protocol. A

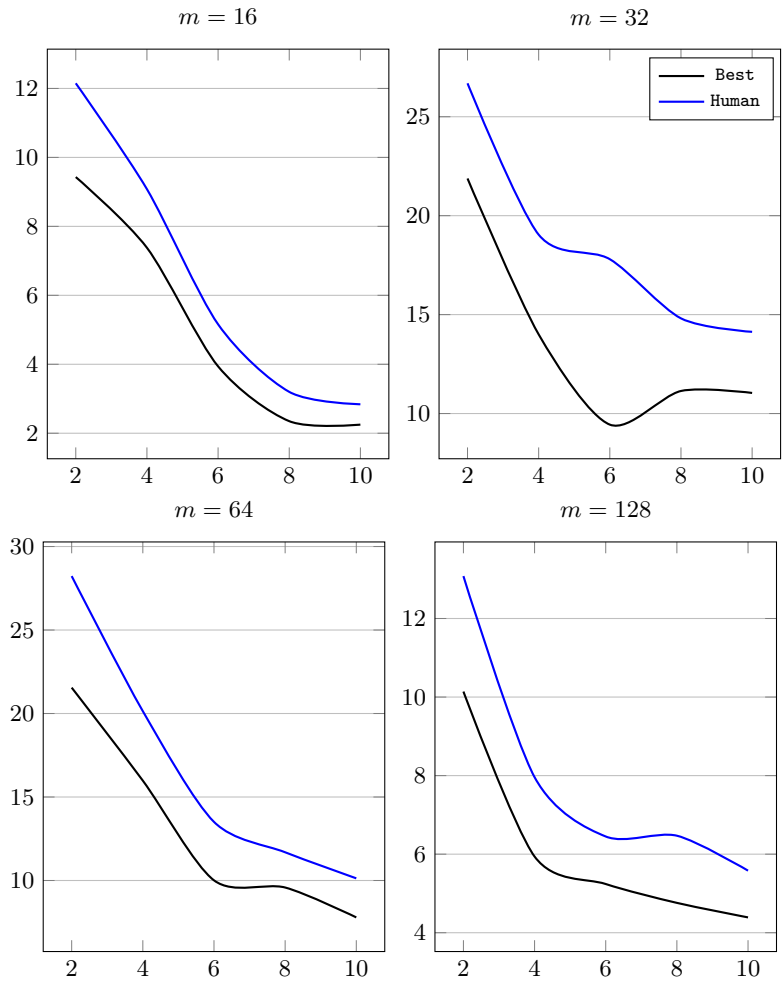

Fig. 4. Runtime comparison of the CDS algorithm under two pivot selection strategies: the *Best* configuration (mathematically optimal pivot, as defined in [7]) and the *Human* configuration (pivots derived from eye-tracking fixation data). The x-axis represents the number of pivots used in the sampling process, while the y-axis reports the average execution time in milliseconds.

more informative sensitivity analysis is likely to emerge on datasets with more pronounced distributional differences (e.g., domain shift or altered noise profiles), where any λ-dependent effects could be amplified.

5 Conclusion

This work introduces a novel perspective on string matching by aligning algorithmic sampling techniques with principles observed in human reading behavior. Inspired by cognitive models of visual attention and perceptual skipping, we explored the Character Distance Sampling via entropy-driven pivot selection, mirroring key facets of human information processing. Experimental results demonstrated that CDS with the best-known pivot selection consistently outperformed cognitively-selected strategies, underscoring the potential of mathematically optimized approaches over human-inspired heuristics in certain computational contexts. Quantitatively, the *Human* configuration exhibited an average slowdown of about 15–30% compared to the *Best* baseline, with the largest deviations observed for intermediate pattern lengths, while the gap narrowed to 10–15% for very short or very long patterns. This reflects how fixation-derived pivots, concentrated around central characters, approximate but do not fully match the distribution required for optimal sampling. These findings suggest that while cognitive models can guide innovative algorithmic designs, rigorous optimization remains critical for achieving peak performance. Future work will investigate hybrid methods that blend cognitive plausibility with mathematical efficiency, and extend the CDS framework to broader classes of pattern matching problems.

Before closing, we highlight several limitations and scope restrictions. First, the cognitive models of reading that inspired this work are typically formulated at the *word level*, whereas our algorithms operate at the *character level*; this mismatch may reduce the fidelity of the analogy. Second, typographic factors such as font, spacing, and layout, known to affect human fixation patterns, were not modeled here, though they could meaningfully influence the distribution of pivots in natural reading. Third, the behavioral data underpinning our discussion stem from corpora and experimental settings that differ from the text domains used in our computational evaluation, which may limit direct comparability. Finally, one of the key related works we reference (i.e. [7]) is a preprint, and thus its findings should be interpreted with caution until they undergo peer review.

References

1. Alberto Apostolico. The myriad virtues of subword trees. In Alberto Apostolico and Zvi Galil, editors, *Combinatorial Algorithms on Words*, pages 85–96, Berlin, Heidelberg, 1985. Springer Berlin Heidelberg.
2. Yevgeni Berzak, Chie Nakamura, Amelia Smith, Emily Weng, Boris Katz, Suzanne Flynn, and Roger Levy. CELER: A 365-Participant Corpus of Eye Movements in L1 and L2 English Reading. *Open Mind*, pages 1–10, 04 2022.
3. R.S. Boyer and J.S. Moore. A fast string searching algorithm. *Communications of the ACM*, 20(10):762–772, 1977.
4. Uschi Cop, Nicolas Dirix, Denis Drieghe, and Wouter Duyck. Presenting geco: An eyetracking corpus of monolingual and bilingual sentence reading. *Behavior Research Methods*, 49(2):602–615, 2017.

5. Simone Faro and Thierry Lecroq. The exact online string matching problem: A review of the most recent results. *ACM Comput. Surv.*, 45(2):13:1–13:42, 2013.
6. Simone Faro, Thierry Lecroq, Stefano Borzi, Simone Di Mauro, and Alessandro Maggio. The string matching algorithms research tool. In Jan Holub and Jan Zdárek, editors, *Proceedings of the Prague Stringology Conference 2016, Prague, Czech Republic, August 29-31, 2016*, pages 99–111. Department of Theoretical Computer Science, Faculty of Information Technology, Czech Technical University in Prague, 2016.
7. Simone Faro, Thierry Lecroq, and Francesco Pio Marino. Optimal text sampling through set cover. `https://ssrn.com/abstract=5337025`, July 2025. SSRN preprint, not peer reviewed.
8. Simone Faro, Thierry Lecroq, Francesco Pio Marino, Arianna Pavone, and Stefano Scafiti. Improving sampled matching through character context sampling. In Ugo de'Liguoro, Matteo Palazzo, and Luca Roversi, editors, *Proceedings of the 25th Italian Conference on Theoretical Computer Science, Torino, Italy, September 11-13, 2024*, volume 3811 of *CEUR Workshop Proceedings*, pages 300–312. CEUR-WS.org, 2024.
9. Simone Faro and Francesco Pio Marino. Reducing time and space in indexed string matching by characters distance text sampling. In Jan Holub and Jan Zdárek, editors, *Prague Stringology Conference 2020, Prague, Czech Republic, August 31 - September 2, 2020*, pages 148–159. Czech Technical University in Prague, Faculty of Information Technology, Department of Theoretical Computer Science, 2020.
10. Simone Faro, Francesco Pio Marino, and Andrea Moschetto. Beyond horspool: A comparative analysis in sampled matching. In Jan Holub and Jan Zdárek, editors, *Prague Stringology Conference 2024, Prague, Czech Republic, August 26-27, 2024*, pages 16–26. Czech Technical University in Prague, Faculty of Information Technology, Department of Theoretical Computer Science, 2024.
11. Simone Faro, Francesco Pio Marino, Antonino Andrea Moschetto, Arianna Pavone, and Antonio Scardace. The great textual hoax: Boosting sampled string matching with fake samples. In Andrei Z. Broder and Tami Tamir, editors, *12th International Conference on Fun with Algorithms, FUN 2024, June 4-8, 2024, Island of La Maddalena, Sardinia, Italy*, volume 291 of *LIPIcs*, pages 13:1–13:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.
12. Simone Faro, Francesco Pio Marino, and Arianna Pavone. Efficient online string matching based on characters distance text sampling. *Algorithmica*, 82(11):3390–3412, 2020.
13. Simone Faro, Francesco Pio Marino, and Arianna Pavone. Enhancing characters distance text sampling by condensed alphabets. In Claudio Sacerdoti Coen and Ivano Salvo, editors, *Proceedings of the 22nd Italian Conference on Theoretical Computer Science, Bologna, Italy, September 13-15, 2021*, volume 3072 of *CEUR Workshop Proceedings*, pages 1–15. CEUR-WS.org, 2021.
14. Simone Faro, Francesco Pio Marino, and Arianna Pavone. Improved characters distance sampling for online and offline text searching. *Theor. Comput. Sci.*, 946:113684, 2023.
15. Simone Faro, Francesco Pio Marino, Arianna Pavone, and Antonio Scardace. Towards an efficient text sampling approach for exact and approximate matching. In Jan Holub and Jan Zdárek, editors, *Prague Stringology Conference 2021, Prague, Czech Republic, August 30-31, 2021*, pages 75–89. Czech Technical University in Prague, Faculty of Information Technology, Department of Theoretical Computer Science, 2021.

16. Paolo Ferragina and Giovanni Manzini. Indexing compressed text. *J. ACM*, 52(4):552–581, jul 2005.
17. Paolo Ferragina and Gonzalo Navarro. *Pizza&Chili*. Available online: pizzachili.dcc.uchile.cl/, 2005.
18. Karl Friston. A theory of cortical responses. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 360(1456):815–836, 2005.
19. John M. Henderson. *Eye movements and scene perception*, pages 593–606. Oxford University Press, 2011.
20. Alan Kennedy and Joel Pynte. Parafoveal-on-foveal effects in normal reading: Evidence from corpus analysis. *Quarterly Journal of Experimental Psychology Section A*, 56(5):869–890, 2003.
21. Jinil Kim, Peter Eades, Rudolf Fleischer, Seok-Hee Hong, Costas S. Iliopoulos, Kunsoo Park, Simon J. Puglisi, and Takeshi Tokuyama. Order-preserving matching. *Theoretical Computer Science*, 525:68–79, 2014. Advances in Stringology.
22. D.E. Knuth, J.H. Morris, and V.R. Pratt. Fast pattern matching in strings. *SIAM Journal on Computing*, 6(2):323–350, 1977.
23. Thierry Lecroq and Francesco Pio Marino. Fast computation of the period and of the shortest cover of a string using its character-distance-sampling representation. *CoRR*, abs/2407.18216, 2024.
24. U. Manber and G. Myers. Suffix arrays: A new method for on-line string searches. *SIAM Journal on Computing*, 22(5):935–948, 1993.
25. Francesco Pio Marino, Simone Faro, and Antonio Scardace. Practical implementation of a quantum string matching algorithm. In Arianna Pavone and Caterina Viola, editors, *Proceedings of the 2024 Workshop on Quantum Search and Information Retrieval, QUASAR 2024, Pisa, Italy, 3 June 2024*, pages 17–24. ACM, 2024.
26. Uroš Nedeljković, Kata Jovančić, and Nace Pušnik. You read best what you read most: An eye tracking study. *J Eye Mov Res*, 13(2), Nov 2020.
27. Sung Gwan Park, Amihood Amir, Gad M. Landau, and Kunsoo Park. Cartesian tree matching and indexing. In Nadia Pisanti and Solon P. Pissis, editors, *30th Annual Symposium on Combinatorial Pattern Matching, CPM 2019, June 18-20, 2019, Pisa, Italy*, volume 128 of *LIPIcs*, pages 16:1–16:14, 2019.
28. Cathy J Price. A review and synthesis of the first 20 years of pet and fmri studies of heard speech, spoken language and reading. *NeuroImage*, 62(2):816–847, 2012.
29. Keith Rayner. Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, 124(3):372–422, 1998.
30. Keith Rayner. Psycholinguistically driven eye movements in reading: A review of models and empirical results. *Visual Cognition*, 20(3):145–168, 2012.
31. Keith Rayner, Timothy J Slattery, and Nathalie N Bélanger. Eye movements, the perceptual span, and reading speed. *Psychon Bull Rev*, 17(6):834–839, Dec 2010.
32. Erik D Reichle, Alexander Pollatsek, David L Fisher, and Keith Rayner. The e-z reader model of eye-movement control in reading: Comparisons to other models. *Behavioral and Brain Sciences*, 26(4):445–476, 2003.
33. Elizabeth R Schotter, Bernhard Angele, and Keith Rayner. Parafoveal processing in reading. *Attention, Perception, & Psychophysics*, 74(1):5–35, 2014.
34. Herbert A. Simon. *Models of Man: Social and Rational*. Wiley, New York, 1957.
35. Adrian Staub. The effect of lexical predictability on distributions of eye fixation durations. *Psychon Bull Rev*, 18(2):371–376, Apr 2011.
36. Roger P.G. Van Gompel, Martin H. Fischer, Wayne S. Murray, and Robin L. Hill. Chapter 1 - eye-movement research: An overview of current and past developments.

In Roger P.G. Van Gompel, Martin H. Fischer, Wayne S. Murray, and Robin L. Hill, editors, *Eye Movements*, pages 1–28. Elsevier, Oxford, 2007.
37. Vijay V. Vazirani. *Approximation Algorithms*. Springer, 2001.
38. U. Vishkin. Deterministic sampling—a new technique for fast pattern matching. *SIAM Journal on Computing*, 20(1):22–33, 1991.