Package: blaster (via r-universe)

November 2, 2026

Type Package

Title Native R Implementation of an Efficient BLAST-Like Algorithm

Version 1.0.9

Date 2026-04-24

Maintainer Manu Tamminen <mavatam@utu.fi>

Description Implementation of an efficient BLAST-like sequence comparison algorithm, written in 'C++11' and using native R datatypes. Blaster is based on 'nsearch' - Schmid et al (2018) <doi:10.1101/399782>.

License BSD_3_clause + file LICENSE

Imports Rcpp (>= 1.0.5)

LinkingTo Rcpp

RoxygenNote 7.1.1

URL https://github.com/tamminenlab/blaster

Repository https://tamminenlab.r-universe.dev

Date/Publication 2026-04-24 12:12:32 UTC

RemoteUrl https://github.com/tamminenlab/blaster

RemoteRef HEAD

RemoteSha b14a7d5b2df38ea27c9bb49744fafaef6dccfd87

Contents

blast 2
blaster 3
read_fasta 3

Index **5**

blast *Runs BLAST sequence comparison algorithm.*

Description

Runs BLAST sequence comparison algorithm.

Usage

```
blast(
  query,
  db,
  maxAccepts = 1,
  maxRejects = 16,
  minIdentity = 0.75,
  alphabet = "nucleotide",
  strand = "both",
  output_to_file = FALSE
)
```

Arguments

query	A dataframe of the query sequences (containing Id and Seq columns) or a string specifying the FASTA file of the query sequences.
db	A dataframe of the database sequences (containing Id and Seq columns) or a string specifying the FASTA file of the database sequences.
maxAccepts	A number specifying the maximum accepted hits.
maxRejects	A number specifying the maximum rejected hits.
minIdentity	A number specifying the minimal accepted sequence similarity between the query and hit sequences.
alphabet	A string specifying the query and database alphabet: 'nucleotide' or 'protein'. Defaults to 'nucleotide'.
strand	A string specifying the strand to search: 'plus', 'minus' or 'both'. Defaults to 'both'. Only affects nucleotide searches.
output_to_file	A boolean specifying the output type. If TRUE, the results are written into a temporary file a string containing the file name and location is returned. Otherwise a dataframe of the results is returned. Defaults to FALSE.

Value

A dataframe or a string. A dataframe is returned by default, containing the BLAST output in columns QueryId, TargetId, QueryMatchStart, QueryMatchEnd, TargetMatchStart, TargetMatchEnd, QueryMatchSeq, TargetMatchSeq, NumColumns, NumMatches, NumMismatches, NumGaps, Identity and Alignment. A string is returned if 'output_to_file' is set to TRUE. This string points to the file containing the output table.

Examples

```
query <- system.file("extdata", "query.fasta", package = "blaster")
db <- system.file("extdata", "db.fasta", package = "blaster")

blast_table <- blast(query = query, db = db)

query <- read_fasta(filename = query)
db <- read_fasta(filename = db)
blast_table <- blast(query = query, db = db)

prot <- system.file("extdata", "prot.fasta", package = "blaster")
prot_blast_table <- blast(query = prot, db = prot, alphabet = "protein")
```

---

blaster *Blaster*

Description

Blaster implements an efficient BLAST-like sequence comparison algorithm.

Author(s)

Manu Tamminen <mavatam.@utu.fi>, Timothy Julian <tim.julian@eawag.ch>, Steven Schmid <stevschmid@gmail.com>

---

read_fasta *Reads the contents of nucleotide or protein FASTA file into a dataframe.*

Description

Reads the contents of nucleotide or protein FASTA file into a dataframe.

Usage

```
read_fasta(
  filename,
  filter = "",
  non_standard_chars = "error",
  alphabet = "nucleotide"
)
```

Arguments

filename	A string specifying the name of the FASTA file to be imported.
filter	An optional string specifying a sequence motif for sequence filtering. Only keeps those sequences containing this motif. Also splits the matched sequences and provides the split parts in two additional columns.
non_standard_chars	A string specifying instructions for handling non-standard nucleotide or amino acid characters. Options include 'remove', 'ignore' or throw an 'error'. Defaults to 'error'.
alphabet	A string specifying the query and database alphabet: 'nucleotide' or 'protein'. Defaults to 'nucleotide'.

Value

A dataframe containing FASTA ids (Id column) and sequences (Seq column). If 'filter' is specified, the split sequences are stored in additional columns Part1 and Part2.

Examples

```
query <- system.file("extdata", "query.fasta", package = "blaster")

query <- read_fasta(filename = query)
```

Index

blast, 2
blaster, 3

read_fasta, 3