

Package: namegender (via r-universe)

October 18, 2026

Type Package

Title Client for the 'NameGender' Name-to-Gender API

Version 0.7.1

Description Infer the likely gender of first names, email addresses and usernames, and the country distribution of a name, through the 'NameGender' web API <<https://namegender.com>>. Each answer carries the share and number of records behind it and the data version that produced it, so results can be reported and reproduced in research. Also returns letter salutations, checks whether a name looks like a real person's name, runs 'CSV' and 'XLSX' files as file jobs, and verifies the signature of 'NameGender' webhooks. An API key is needed.

License MIT + file LICENSE

Depends R (>= 4.1.0)

Encoding UTF-8

Imports jsonlite, httr2, openssl, utils

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

RoxygenNote 7.3.2

URL <https://namegender.com/docs>,
<https://github.com/anpekesen/namegender-r>

BugReports <https://github.com/anpekesen/namegender-r/issues>

NeedsCompilation no

Author Anil Pekesen [aut, cre], NameGender [cph]

Maintainer Anil Pekesen <info@namegender.com>

Repository <https://cran.r-universe.dev>

Date/Publication 2026-10-18 14:30:02 UTC

RemoteUrl <https://github.com/cran/namegender>

RemoteRef HEAD

RemoteSha 243845aeeb372bb801ebe7e3f1ec1b9e0b0b2de2

Contents

namegender	2
ng_batch	3
ng_name_check	6
ng_salutation	7
ng_webhook_verify	8
Index	10

namegender	<i>NameGender API Client</i>
------------	------------------------------

Description

Create a NameGender client and resolve names, email addresses, usernames, or batches. For files, see [ng_batch](#); for salutations, [ng_salutation](#); for name checks, [ng_name_check](#); for webhooks, [ng_webhook_verify](#).

`ng_countries()` returns the countries a name is recorded in. It is not a country-of-origin or ethnicity inference: registrations is counted volume, comparable only among the countries that publish counted birth statistics, and `attested_in` is presence with no weight attached. Show `basis$note` next to any percentage.

Usage

```
namegender(api_key, base_url = "https://namegender.com/api/v1")

ng_name(client, name, country = NULL, locale = NULL, ip = NULL, ...)

ng_email(client, email, country = NULL, locale = NULL, ip = NULL, ...)

ng_username(client, username, country = NULL, locale = NULL, ip = NULL, ...)

ng_bulk(client, names, country = NULL, type = "name",
        locale = NULL, ip = NULL, ...)

ng_countries(client, name, limit = NULL)
```

Arguments

<code>api_key</code>	A NameGender API key.
<code>base_url</code>	API base URL.
<code>client</code>	A client returned by <code>namegender()</code> .
<code>name</code>	A first or full name. For <code>ng_countries()</code> , the first name to look up.
<code>email</code>	An email address.
<code>username</code>	A username or handle.

names	A character vector of inputs.
country	Optional ISO 3166-1 alpha-2 country code.
type	Bulk input type: name, email, or username.
locale	Optional language tag such as "it-IT" or "pt-BR" (up to 35 characters). Its region is used as the country when country is not given; a tag without a region, such as "en", sets no country.
ip	Optional IP address of your end user (up to 45 characters). Its country is used when neither country nor a regional locale is given. The API does not store it.
limit	Optional number of counted countries to return in registrations, 1 to 100. The server default is 25.
...	Additional API options: ai_fallback = TRUE falls back to a language model for names not in the database (needs AI consent on the account); best_guess = TRUE returns the most likely gender even below the probability threshold.

Value

namegender() returns a client object. Lookup functions return the parsed API response. A lookup result has query, name, first_name, middle_name, last_name, name_type, gender, country, probability, sample_size, took_ms, source, confidence and matched_as, alongside credits_charged, credits_remaining, data_version, request_id and country_source. country_source says which input set the country: "country", "locale" or "ip", or NULL when none did (the priority is country > locale > ip). For ng_bulk() it is on the response envelope, next to took_ms, not on each row of results. A non-2xx response is raised as an error carrying the API's message; its class is namegender_error, with status and the API's body on the condition.

Examples

```
## Not run:
client <- namegender(Sys.getenv("NAMEGENDER_API_KEY"))
ng_name(client, "Ayse", country = "TR")
ng_name(client, "Andrea", country = "IT", best_guess = TRUE)
ng_name(client, "Andrea", locale = "it-IT", ip = "203.0.113.7")
ng_countries(client, "Mehmet", limit = 10)

## End(Not run)
```

ng_batch

File Jobs

Description

Upload a CSV or XLSX file (up to 100 MB and 1,000,000 rows) and get it back with gender columns added. One credit per row, charged only if the job completes.

name_column is required to start: a guessed column that turns out to be wrong would spend credits on the wrong data. To see the columns and the cost first, upload with start = FALSE, read job\$inspection, then call ng_batch_start().

ng_batch_create() sends an Idempotency-Key and retries connection errors and 502/503/504 with the same key (after 1 and then 2 seconds), so a retry never opens a second job. Other errors, 4xx included, are not retried. Pass your own idempotency_key to keep that guarantee across your own retries.

Usage

```
ng_batch_create(client, file, name_column = NULL, country_column = NULL,
  country = NULL, ai_fallback = NULL, best_guess = NULL,
  delete_after_download = NULL, start = TRUE, idempotency_key = NULL,
  retries = 2, filename = NULL, ...)
```

```
ng_batch_start(client, id, name_column, ...)
```

```
ng_batch_get(client, id)
```

```
ng_batch_list(client, limit = NULL, page = NULL)
```

```
ng_batch_cancel(client, id)
```

```
ng_batch_wait(client, id, timeout = 3600, on_progress = NULL)
```

```
ng_batch_download(client, id, path = NULL)
```

Arguments

client	A client returned by namegender().
file	Path to a .csv, .tsv or .xlsx file with a header row, or a raw vector (then filename is required). The file name's extension tells the API the format.
name_column	Header of the column holding the names, exactly as written in the file. Required to start.
country_column	Optional header of a column holding a country code per row.
country	Optional default ISO 3166-1 alpha-2 country code for rows without one.
ai_fallback	TRUE asks a language model for names not in the database (needs AI consent on the account).
best_guess	TRUE returns the more likely gender even when the evidence is weak.
delete_after_download	TRUE lets the result be downloaded once, then deletes it.
start	FALSE uploads and inspects only; the job then carries inspection (columns, preview, cost).
idempotency_key	Optional key, up to 80 printable ASCII characters. One is generated per call when omitted.

retries	How many times to retry a connection error or a 502/503/504.
filename	File name to send, needed when file is a raw vector.
id	A job id, such as "B-7K2M9QX4TA".
limit, page	Optional paging for ng_batch_list().
timeout	Seconds to wait before giving up.
on_progress	Optional function called with the job after every poll.
path	Optional path to write the result file to.
...	Further settings sent as they are. Logical values are sent as "true" or "false"; NULL values are left out.

Value

ng_batch_create(), ng_batch_start(), ng_batch_get() and ng_batch_wait() return the job as a list: id, status (uploaded, queued, processing, completed, failed or cancelled), progress, rows, credits, summary, error and poll_after_seconds, among others. ng_batch_list() returns a page of jobs, newest first, including jobs started from the dashboard.

ng_batch_wait() polls every poll_after_seconds (5 when absent) until the job is completed, failed or cancelled (or still uploaded), and returns it. A failed job is returned, not raised: check status and error\$code. Running past timeout is an error.

ng_batch_cancel() returns the credit of a job that has not started, and deletes a finished one; it returns NULL invisibly.

ng_batch_download() returns the result file as a raw vector, or writes it to path and returns path invisibly. The result appends gender, probability, sample_size, country, source, matched_as, first_name, middle_name, last_name and name_type to every row. A CSV result starts with a UTF-8 byte order mark; read it with fileEncoding = "UTF-8-BOM".

A non-2xx response is raised as a condition of class namegender_error carrying status and the API's body. Up to three jobs can be queued or running at once; a fourth is refused with 429 too_many_batches.

Examples

```
## Not run:
client <- namegender(Sys.getenv("NAMEGENDER_API_KEY"))
job <- ng_batch_create(client, "customers.csv",
  name_column = "first_name", country_column = "country")
done <- ng_batch_wait(client, job$id, on_progress = function(j) message(j$progress, "%"))
if (done$status == "failed") stop(done$error$code)
ng_batch_download(client, done$id, "customers-gender.csv")

## End(Not run)
```

ng_name_check	<i>Name checks</i>
---------------	--------------------

Description

Whether a name typed into a form looks like a real person's name, with the reasons. One credit per name.

It never calls a name fake: assessment is "plausible", "suspicious" or "implausible". Use it to flag records for a closer look, not to reject people automatically. Surnames are judged by their shape only. best_guess, ai_fallback and language do not apply and are not accepted.

Usage

```
ng_name_check(client, name = NULL, country = NULL, locale = NULL, ip = NULL,
  first_name = NULL, last_name = NULL)
```

```
ng_name_check_bulk(client, names, country = NULL, locale = NULL, ip = NULL)
```

Arguments

client	A client returned by namegender() .
name	The full name as typed ("asdf qwerty"), up to 200 characters.
names	A character vector of 1 to 100 full names. The other arguments apply to every name.
country	Optional ISO 3166-1 alpha-2 country code.
locale	Optional language tag such as "en-US", a country hint.
ip	Optional IP address of your end user, a country hint. Not stored by the API.
first_name, last_name	Use instead of name when they are stored separately, up to 100 characters each. They are taken as they are, not parsed.

Value

The parsed API response. `ng_name_check()` returns query, assessment, score (an integer from 0 to 100), signals (a data frame with code, severity, part and value; part and value can be NA), first_name, last_name, name_type ("personal", "organization" or "role") and evidence (first_name_status, which is "counted", "attested", "not_found" or NULL, and first_name_counted_records), alongside credits_charged, credits_remaining, data_version, request_id and country_source.

Signal codes include keyboard_pattern, repeated_characters, placeholder, fictional_character, contains_digits, organization_name, single_name, first_name_not_found and first_name_attested; severity is high, medium, low, info or positive.

`ng_name_check_bulk()` returns summary (total, plausible, suspicious, implausible) and results, a data frame with one row per name in input order and the same fields from query to evidence; the credit fields and country_source are on the envelope.

A non-2xx response is raised as a `namegender_error` condition.

Examples

```
## Not run:
client <- namegender(Sys.getenv("NAMEGENDER_API_KEY"))
ng_name_check(client, "asdf qwerty")$assessment # "implausible"
ng_name_check(client, "Jennifer Null")$assessment # "plausible"
ng_name_check(client, first_name = "Jennifer", last_name = "Null", country = "US")
bulk <- ng_name_check_bulk(client, c("Jennifer Null", "asdf qwerty"))
bulk$results[, c("query", "assessment", "score")]

## End(Not run)
```

ng_salutation

Salutations

Description

A ready-to-use salutation for a letter or an email, such as "Sehr geehrte Frau Dr. Müller," or "Sayın Ahmet Bey, ". One credit per name.

When the gender is not certain enough (`min_probability`), the neutral form is returned rather than a guess: form is "gendered", "neutral" or "organization", and reason says why a form is not gendered. `best_guess` and `ai_fallback` do not apply and are not accepted.

Usage

```
ng_salutation(client, name = NULL, language = NULL, country = NULL,
  locale = NULL, ip = NULL, gender = NULL, min_probability = NULL,
  title = NULL, first_name = NULL, last_name = NULL)
```

```
ng_salutation_bulk(client, names, language = NULL, country = NULL,
  locale = NULL, ip = NULL, gender = NULL, min_probability = NULL,
  title = NULL)
```

Arguments

<code>client</code>	A client returned by <code>namegender()</code> .
<code>name</code>	The full name, titles included ("Dr. Anna Müller"), up to 200 characters.
<code>names</code>	A character vector of 1 to 100 full names. The other arguments apply to every name.
<code>language</code>	Language of the salutation: en, en-US, en-GB, de, de-AT, de-CH, fr, es, it, pt, pt-PT, pt-BR, nl, tr, pl or ja. Without it, the language of locale, then the main language of the country, then English. Any other value is a 422 error.
<code>country</code>	Optional ISO 3166-1 alpha-2 country code, a hint for the gender lookup.
<code>locale</code>	Optional language tag such as "de-AT": a country hint, and the default language.
<code>ip</code>	Optional IP address of your end user, a country hint. Not stored by the API.

gender	A known gender, "male", "female" or "neutral"; skips the lookup. "neutral" always gives the neutral form.
min_probability	Certainty, 50 to 100, below which the neutral form is used. The server default is 90.
title	An academic title kept in a separate field, such as "Dr. ". Used in German and English.
first_name, last_name	Use instead of name when they are stored separately. They are taken as they are, not parsed.

Value

The parsed API response. `ng_salutation()` returns query, language, form, reason, salutation (formal, informal, neutral), parts (opening, courtesy, academic, name, each possibly NULL), gender, gender_source, probability, confidence, first_name, last_name, name_type and country, alongside credits_charged, credits_remaining, data_version, request_id and country_source.

`ng_salutation_bulk()` returns summary (total, gendered, neutral, organization) and results, a data frame with one row per name in input order and the same fields from query to country; the credit fields, language and country_source are on the envelope.

A non-2xx response is raised as a `namegender_error` condition. For an unsupported language its status is 422 and `body$supported` lists the supported languages.

Examples

```
## Not run:
client <- namegender(Sys.getenv("NAMEGENDER_API_KEY"))
ng_salutation(client, "Dr. Anna Müller", language = "de")$salutation$formal
ng_salutation(client, "Ahmet Yılmaz", language = "tr")$salutation$formal
ng_salutation(client, first_name = "Anna", last_name = "Müller", title = "Dr.", language = "de")
bulk <- ng_salutation_bulk(client, c("Dr. Anna Müller", "Müller GmbH"), language = "de")
bulk$resultss$salutation$formal

## End(Not run)
```

ng_webhook_verify

Verify a NameGender Webhook

Description

Check the NameGender-Signature header of a webhook request and return the parsed event.

Pass the body exactly as received. Parsing the JSON and serialising it again changes the bytes, and the signature no longer matches. During a secret rotation the header carries two v1 values; either one matching is enough.

Usage

```
ng_webhook_verify(payload, signature_header, secret, tolerance = 300, now = NULL)
```

Arguments

payload	The raw request body, as a raw vector or a string (read as UTF-8).
signature_header	The value of the NameGender-Signature header, t=<unix seconds>, v1=<hex>.
secret	The endpoint's signing secret from the dashboard.
tolerance	Seconds the timestamp may differ from the clock.
now	Optional current time in Unix seconds, for tests.

Value

The event as a list: id, type (batch.completed, batch.failed, credits.low, credits.depleted or webhook.test), created_at, api_version and data. For a batch event data\$object is the job, as ng_batch_get() returns it.

A missing or malformed header, a timestamp outside the tolerance window, or a signature that does not match is raised as a condition of class namegender_webhook_error (also a namegender_error). Answer such a request with 400.

Examples

```
event <- ng_webhook_verify(
  '{"id":"evt_1","type":"webhook.test"}',
  "t=1700000000,v1=857fcddfea47617c448b7a8e6537bbd59c9922a37c5273b2709812fbadb29e50",
  "whsec_test_vector",
  now = 1700000000
)
event$type
```

Index

namegender, [2](#), [6](#), [7](#)
ng_batch, [2](#), [3](#)
ng_batch_cancel (ng_batch), [3](#)
ng_batch_create (ng_batch), [3](#)
ng_batch_download (ng_batch), [3](#)
ng_batch_get (ng_batch), [3](#)
ng_batch_list (ng_batch), [3](#)
ng_batch_start (ng_batch), [3](#)
ng_batch_wait (ng_batch), [3](#)
ng_bulk (namegender), [2](#)
ng_countries (namegender), [2](#)
ng_email (namegender), [2](#)
ng_name (namegender), [2](#)
ng_name_check, [2](#), [6](#)
ng_name_check_bulk (ng_name_check), [6](#)
ng_salutation, [2](#), [7](#)
ng_salutation_bulk (ng_salutation), [7](#)
ng_username (namegender), [2](#)
ng_webhook_verify, [2](#), [8](#)