

# Package: openscoring (via r-universe)

October 27, 2024

**Title** 'Open Scoring' API Client

**Version** 1.0.5

**Description** Creativity research involves the need to score open-ended problems. Usually done by humans, automatic scoring using AI becomes more and more accurate. This package provides a simple interface to the 'Open Scoring' API  [<https://openscoring.du.edu/docs>](https://openscoring.du.edu/docs), leading creativity scoring technology by Organiscak et al. (2023)  [<doi:10.1016/j.tsc.2023.101356>](https://doi.org/10.1016/j.tsc.2023.101356). With it, you can score your own data directly from an R script.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**URL** <https://github.com/jakub-jedrusiak/openscoring>

**BugReports** <https://github.com/jakub-jedrusiak/openscoring/issues>

**Imports** cli, dplyr, glue, httr, jsonlite, lifecycle, purrr, rlang, stringr

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Repository** <https://jakub-jedrusiak.r-universe.dev>

**RemoteUrl** <https://github.com/jakub-jedrusiak/openscoring>

**RemoteRef** HEAD

**RemoteSha** 9e7fc80da84d396a36aaed71b1de85f2ed5c949

## Contents

ocsai . . . . .	2
<b>Index</b>	<b>4</b>

---

ocsai *Score with an AI A basic function to score the creativity with an AI. See [R href=https://openscoring.du.edu/scoringllm](https://openscoring.du.edu/scoringllm) the OpenScoring site for more information. Requires an internet connection.*

---

## Description

Score with an AI A basic function to score the creativity with an AI. See [the OpenScoring site](#) for more information. Requires an internet connection.

## Usage

```
ocsai(
  df,
  item,
  answer,
  model = c("1.6", "1-4o", "davinci3", "chatgpt2", "1.5", "chatgpt", "babbage2",
            "davinci2"),
  language = "English",
  scores_col = ".originality",
  quiet = FALSE,
  chunk_size = 50
)
```

## Arguments

df	A data frame.
item	The column name of the items or other kind of prompt.
answer	The column name of the responses. Commas will be replaced with spaces for scoring.
model	The model to use. Should be one of "1.6", "1-4o", "davinci3", "chatgpt2". Deprecated models are kept for compatibility.
language	The language of the input. Only works for the 1.5 model upwards. Should be one of "Arabic", "Chinese", "Dutch", "English", "French", "German", "Hebrew", "Italian", "Polish", "Russian", "Spanish".
scores_col	The column name to store the scores in. Defaults to ".originality".
quiet	Whether to print the citation reminder.
chunk_size	The number of rows to send to the API at once. Defaults to 50. If a request is too large, it will be split into 10-row chunks.

## Details

Available models:

- ocsai-1.6: Update to the multi-lingual, multi-task 1.5 model, trained on GPT 4o instead of 3.5.

- ocsai1-4o: GPT-4o-based model, trained with more data and supporting multiple tasks. Last update to the Ocsai 1 models (i.e. the original ones).
- ocsai-chatgpt2: GPT-3.5-size chat-based model, trained with more data and supporting multiple tasks. Scoring is slower, with slightly better performance than ocsai-davinci.
- ocsai-davinci3: GPT-3 Davinci-size model. Trained with the method from Organisciak et al. 2023, but with the additional tasks (uses, consequences, instances, complete the sentence) from Acar et al 2023, and trained with more data.
- ocsai-1.5: Beta version of new multi-lingual, multi-task model, trained on GPT 3.5.
- ocsai-chatgpt: GPT-3.5-size chat-based model, trained with same format and data as original models. Scoring is slower, with slightly better performance than ocsai-davinci2. For more tasks and trained on more data, use davinci-ocsai2
- ocsai-babbage2: GPT-3 Babbage-size model from the paper, retrained with new model API. Deprecated, mainly because other models work better.
- ocsai-davinci2: GPT-3 Davinci-size model from the paper, retrained with a new model API.

## Value

The input data frame with the scores added.

## Examples

```
df <- data.frame(
  stimulus = c("brick", "hammer", "sponge"),
  response = c("butter for trolls", "make Thor jealous", "make it play in a kids show")
)

df <- ocsai(df, stimulus, response, model = "davinci3")

# The 1.5 model and upwards works for multiple languages
df_polish <- data.frame(
  stimulus = c("cegła", "młotek", "gąbka"),
  response = c("masło dla trolli", "wywoływanie zazdrości u Thora", "postać w programie dla dzieci")
)

df_polish <- ocsai(df_polish, stimulus, response, model = "1.5", language = "Polish")
```

# Index

ocsai, [2](#)