automatic Essay Grading and Pligarism check Models API

2.0.1 OAS3

/openapi.json

An API that use NLP models to return the plagiarism , grading result

default

GET / Read Root

read index (root) for testing purpose

Args: query (str): query string

Returns: str: "message": hi

Parameters

Try it out

No parameters

Responses

Code	Description	Links

Code	Description	Links
200	Successful Response Media type application/json Controls Accept header. **Example Value** Schema "string"	*No links*

POST /grading/predict Predict Grad

Predict the grading result

Args

answers : dict[str,str] dict of student ids and their answers first element is the model answer second etc... elements are the students answers

Returns

Dict[str,float] dict of studetns ids and their grades

example

```
>>> predict_grad("essays_dict":
>>> {"7821": "ahmed is eating a pizza",
>>> "156": "i go to school by bus",
>>> "3": "i am eating a pizza",
>>> "4": "ahmed is eating a pizza"}
>>>  "grading_model": "automatic_keywords_grading")

"grades": {
    "156": 0.028096482157707214,
    "3": 0.45442324082056684,
    "4": 0.9999998410542806
        }
```

Parameters

Try it out

No parameters

Request body required

application/json

Example Value Schema

```
{
  "essays_dict": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "grading_model": "siamese_ner_grading",
  "cased": false
}
```

Responses

Code	Description	Links
200	Successful Response	*No links*

Media type

application/json

Controls `Accept` header.

Example Value Schema

```
{
  "grades": {
    "additionalProp1": 0,
    "additionalProp2": 0,
    "additionalProp3": 0
  }
}
```

Code	Description	Links
422	Validation Error	*No links*

Media type

application/json

Example Value Schema

```
{
  "detail": [
    {
      "loc": [
        "string"
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

POST `/plagiarism/predict` Predict Plagiarism

Predict the pligarism result

Args

essays_dict : dict[str,str] dict of student ids and their answers

Returns

dict[str,dict[str,float]] dict of cheating studetns ids and their probabilities of plagiarism with other students

example

> predict_plagiarism('{1:"Hello World", 2:"Hello World"}, 3:"hi there"},cased: False') [{1: {2: 0.98}, 2: {1: 0.98}}]

Parameters Try it out

No parameters

Request body required application/json

Example Value Schema

```
{
  "essays_dict": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "cased": false
}
```

Responses

Code	Description	Links
200	Successful Response	*No links*

Media type

application/json

Controls `Accept` header.

Example Value Schema

```
{
  "plagiarism_results": [
    {
      "additionalProp1": {
        "additionalProp1": 0,
        "additionalProp2": 0,
        "additionalProp3": 0
      },
      "additionalProp2": {
        "additionalProp1": 0,
        "additionalProp2": 0,
        "additionalProp3": 0
      },
      "additionalProp3": {
        "additionalProp1": 0,
        "additionalProp2": 0,
        "additionalProp3": 0
      }
    }
  ]
}
```

Code	Description	Links
422	Validation Error	*No links*

Schemas

ESSAYSDICT {

description:

ESSAYSDICT input model validation

Args: BaseModel : inherit from pydantic BaseModel to validate the input

Attributes: essays_dict (dict): essays_dict contains dicts of the student's answers and their ids grading_model (str): grading_model grading model to use choose from ["siamese_ner_grading", "keywords_grading" , "manual_keywords_grading"] cased (bool): cased strings grading or not

example: >>> "essays_dict": { >>> "1235dx": >>> "answer_1" >>> "24463dxcf": >>> "answer_2" >>> } >>> "grading_model": "keywords_grading" >>> "cased": False

essays_dict* **Essays Dict** {...}

grading_model **string**
title: Grading Model
default: siamese_ner_grading

cased **boolean**
title: Cased
default: false

}

GradingResponse {

description:

StudentsDict output model validation it's a json string contains a List of Dictionaries of str [ids] : float [scores]

Args: BaseModel : inherit from pydantic BaseModel to validate the output

Attributes: grades ([Dict[str, float]]): scores contains a list of dicts of the student's grades and their ids

example: >>> "essays_dict": { >>> "1235dx": >>> "answer_1" >>> "24463dxcf": >>> "answer_2" >>> }

```
>>> "grades":
>>>     {
>>>         "7818ert": 1,
>>>         "4581rdc": 0.5,
>>>     }
```

grades* **Grades** {...}

}

HTTPValidationError {

detail

Detail [...]

}

PlagiarismResponse {

description:

PlagiarismResponse output model validation it's a json string contains a List of Dictionaries of str [ids] : float [scores]

Args: BaseModel : inherit from pydantic BaseModel to validate the output

Attributes: List[Dict[str, Dict[str,float]]] List of Dictionaries of str [ids] : float [scores]

scores (List[Dict[str, float]]): scores contains a list of dicts of the student's answers and their ids

example: >>> "plagiarism_results": [>>> {"1235dx": 0.9, "24463dxcf": 0.8}, >>> {"1235dx": 0.8, "24463dxcf": 0.9} >>>]

plagiarism_results*

Plagiarism Results [...]

}

StudentsDict {

description:

StudentsDict input model validation

Args: BaseModel : inherit from pydantic BaseModel to validate the input

Attributes: essays_dict (dict): essays_dict contains dicts of the student's answers and their ids cased (bool): cased strings grading or not

example: >>> "essays_dict": { >>> "1235dx": >>> "answer_1" >>> "24463dxcf": >>> "answer_2" >>> } >>> "cased": False

essays_dict*

Essays Dict {...}

cased

boolean
title: Cased
default: false

}

```
ValidationError   {
   loc*               Location    [...]
   msg*               string
                      title: Message
   type*              string
                      title: Error Type

}
```