

ÖNÁLLÓ LABORATÓRIUM

Konzulens: Dr. Forstner Bertalan

Készítette: Fogti István

Budapesti Műszaki és
Gazdaságtudományi Egyetem | 2023

MI IS A BAJ?

Az oktatói lét egyik nehezen automatizálható feladata azoknak a kisZH vagy laborbeugró feladatoknak a javítása, amelyekre szöveges válasz érkezik.

Egyrészt

Egy-egy zárthelyit vagy beugrót akár több százan is megírhatnak.

Másrészt

Igaz-hamis, illetve a-b-c-d feleletválasztós kérdésekkel nem lehet a hallgató önálló tudását tesztelni.

Explain the difference between malloc() and new in C++.

malloc is a function to allocate memory in C. new is an operator for doing the same in c++. In the background new uses malloc as well. After calling new, delete must be called, whereas after calling malloc free must be called to deallocate the memory.

A quiz készítője a kérdés generálásakor megadhat értékelési útmutatót

2 - malloc() is a C function used for dynamic memory allocation, while new is a C++ operator that also allocates memory dynamically;

2 - Memory allocated with malloc() must be explicitly deallocated with free(), whereas memory allocated with new should be deallocated with delete;

2 - new is type-safe, whereas malloc() returns a void pointer, requiring explicit type casting;

Comment

Rich text editor toolbar with icons for undo, redo, bold, italic, bulleted list, numbered list, decrease indent, increase indent, link, unlink, smiley, image, video, microphone, camera, and source code.

Good job on your answer! You were able to earn 4 out of 6 points for your response. However, there are a few thoughts that were missing from your answer that I wanted to bring to your attention. One important point to note is that "new" in C++ is type-safe. On the other hand, "malloc()" in C returns a void pointer, which means

Mark

4 out of 6.00

"Great job on your answer! You were able to earn 4 out of the 6 points for this question. In order to improve your answer, it would be helpful to mention that the "new" operator in C++ is considered type-safe. This means that it automatically deduces the type of the object being created and allocates memory accordingly. On the other hand, the "malloc()" function in C returns a void pointer, which requires explicit type casting when used. Including this information in your answer would make it more comprehensive and demonstrate a deeper understanding of the topic. Keep up the good work!"

automatikus pontozás

NAGY NYELVI MODELLEK

- 1 Az elmúlt években egyre nagyobb figyelmet nyernek az LLM-ek.
- 2 Ezek a modellek nem csak egy-egy szó jelentését értik egy szövegen belül, hanem a szavak egymáshoz való viszonyulását is, az egész kontextust.
- 3 Az LLM-eket ezen új képességük teszik alkalmassá a szöveges válaszok kiértékelésére, hiszen képesek a megadott válasz tartalmát értelmezni, és nem a konkrét szavak alapján értékelik azt.

CÉLKITŰZÉSEK

● Egy algoritmus

Egy olyan algoritmus megalkotása, amely segítségével automatizálni lehet szöveges válaszok javítását, egy előre megadott értékelési útmutató alapján.

● Moodle integráció

A kész algoritmus integrálása moodle környezetbe.

A MODELL KIVÁLASZTÁSA

● GPT?

- + state-of-the-art teljesítmény
- + API
- + nagy közösség
- költségek

● FLAN-T5?

- + open-source
- + fine-tune
- teljesítmény

● Bert?

- + open-source
- + fine-tune
- teljesítmény

● Sentencetransformers??

- + open-source
- + fine-tune
- teljesítmény
- LLM?

GPT

GPT-4 modell végzi a kritériumok és a válaszok összevetését

```
def get_assessment(prompt):
    response = client.chat.completions.create(
        model="gpt-4",
        messages=[
            {"role": "system", "content": "You are a nice teacher controlling the answers of your students. \
            Output the word 'ok', if the content of the criteria is present in the answer. \
            Otherwise output every thought composed in the criteria that is not present in the answer."},
            {"role": "user", "content": prompt},
        ]
    )
```

```
li-docker > requirements.txt
blinker==1.6.2
click==8.1.6
colorama==0.4.6
Flask==2.3.2
itsdangerous==2.1.2
Jinja2==3.1.2
MarkupSafe==2.1.3
Werkzeug==2.3.6
openai
IPython
python-dotenv
```

GPT-3.5-turbo összefoglalja a válasz hiányosságait

```
def summarize_comments(comment, score, maxscore):
    response = client.chat.completions.create(
        model="gpt-3.5-turbo",
        messages=[
            {"role": "system", "content": "You are helping a teacher summarize comments given to a student's answer in an exam."},
            {"role": "user", "content": "I collected the thoughts that are missing from an answer. Compose this comment in an exact v
            The answer reached " + str(score) + " points out of " + str(maxscore) + ". \
            The comment: " + comment},
        ]
    )
```

SENTENCETRANSFORMERS

- **Embedding vektorok**
- **Egy gondolat szétszórva a szövegben?**
- **Cosine similarity**
 - Tulajdonképpen valami hasonlót csinálhat a háttérben a gpt is.
- **Euclidean similarity**
- **Terület specifikusság?**

SENTENCE TRANSFORMERS

```
def evaluate(criterias, answer):
    global score
    answer_chunks = split_into_clauses(answer)
    answer_chunks.append(answer)
    answer_embeddings = model.encode(answer_chunks, convert_to_tensor=True)
    missing_information = []

    for criteria in criterias:
        contains = False
        criteria_clauses = split_into_clauses(criteria[1])
        cosine_scores = util.cos_sim(criteria_embedding, answer_embeddings)

        numofclauses = len(criteria_clauses)
        NumOfCorrectClauses = 0
        for crit_clause in criteria_clauses:
            criteria_embedding = model.encode(crit_clause, convert_to_tensor=True)
            euclidean_dist_arr = []

            for ans in answer_embeddings:
                crit_arr = criteria_embedding.cpu().numpy()
                answ_arr = ans.cpu().numpy()
                dist = answ_arr - crit_arr
                euclidean_dist_arr.append(np.linalg.norm(dist))

            for cos in cosine_scores:
                for c in cos:
                    if c > 0.79:
                        for dist in euclidean_dist_arr:
                            if dist < 0.51:
                                NumOfCorrectClauses += 1
                                break

        if numofclauses == NumOfCorrectClauses:
            score += int(criteria[0])
        else:
            missing_information.append(str(criteria[1]))

    return [score, missing_information]
```

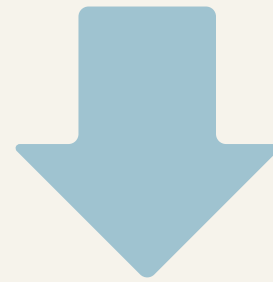
SENTENCETRANSFORMERS

Pontosság

0	Eredeti modell	~40%
1	A válasz feldarabolása	72.33%
2	A kritérium feldarabolása	76%
3	Euklideszi távolság hozzáadása	80%

MOODLE INTEGRÁCIÓ

Grader info megadása a kérdés definiálásakor



A review ablak megnyitásakor kérés a modell konténerhez.



A kérdésbe csomagolt graderinfo és válasz alapján a modell értékeli, és a responseban elküldi a pontszámot, és a szöveges értékelést.

Explain the difference between malloc() and new in C++.

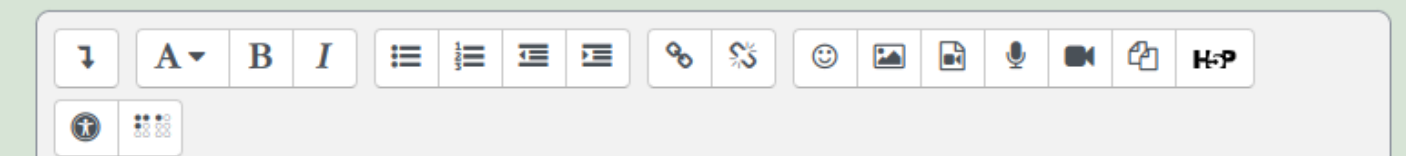
malloc is a function to allocate memory in C. new is an operator for doing the same in c++. In the background new uses malloc as well. After calling new, delete must be called, whereas after calling malloc free must be called to deallocate the memory.

2 - malloc() is a C function used for dynamic memory allocation, while new is a C++ operator that also allocates memory dynamically;

2 - Memory allocated with malloc() must be explicitly deallocated with free(), whereas memory allocated with new should be deallocated with delete;

2 - new is type-safe, whereas malloc() returns a void pointer, requiring explicit type casting;

Comment



Good job on your answer! You were able to earn 4 out of 6 points for your response. However, there are a few thoughts that were missing from your answer that I wanted to bring to your attention. One important point to note is that "new" in C++ is type-safe. On the other hand, "malloc()" in C returns a void pointer, which means

Mark

4 out of 6.00

MOODLE INTEGRÁCIÓ

```
if ($currentmark == "" && $commenttext == "" && $typename == "essay") {  
    $currentmark = "4";  
    $graderinfo = $qa->get_question()->graderinfo;  
    $formattedgraderinfo = strip_tags($graderinfo);  
    $answer = $qa->get_response_summary();  
  
    $data = array('criteria' => (string)$formattedgraderinfo, 'answer' => (string)$answer); // replace with your data  
    $url = 'http://localhost:5000/'; // path to your API  
    $payload = array(  
        'http' => array(  
            'header' => "Content-type: application/json\r\n",  
            'method' => 'POST',  
            'content' => json_encode($data),  
        ),  
    );  
    $context = stream_context_create($payload);  
    $response = file_get_contents($url, false, $context);  
  
    $responseData = json_decode($response, true); // parse the JSON into an associative array  
  
    $currentmark = $responseData[0];  
    $commenttext = $responseData[1];  
}
```

TERVEK A JÖVŐRE

● Automatizálás

- A válaszok beküldése után egyből érkezzen feedback.

● RAG architektúra

● Magyar verzió

● Integráció

- Jelenlegi moodle integráció tulajdonképpen a core függvények módosítása.

● ZH generálás

Budapesti Műszaki és
Gazdaságtudományi Egyetem | 2023

KÖSZÖNÖM A FIGYELMET

Készítette: Fogti István