TEDx-Perience

Pyspark & AWS Glue

Cremonesi Andrea (1074260) - Nigro Marco (1046992)

Get_Watch_Next_by_Idx

La funzione **Get_Watch_Next_by_Idx** restituisce la lista dei dati (*"_id"*, *"title"*, *"url"*) riguardanti i talk correlati ordinati per *num_views* associati al video avente *"_id"* passato come parametro

```
connect_to_db().then(async () => {
    let next_talks_data_list = [];

    try{
        let talk = await talksModel.findOne({_id: body.id});
        let watch_next_ids = talk.watch_next_list;

        for(let i = 0; i < watch_next_ids.length; i++){
            let next_talk = await talksModel.findOne({_id: watch_next_ids[i]}, {title:1, url:1});

            next_talks_data_list.push(next_talk);
        }

        callback(null, {
            statusCode: 200,
            body: JSON.stringify(next_talks_data_list)
        })
    }
    catch(err){
        callback(null, {
            statusCode: err.statusCode || 500,
            headers: { 'Content-Type': 'text/plain' },
            body: 'Could not fetch the next talk data.'
        });
    }
});
```

Input:

- **id:** *_id* del talk di cui desideriamo visualizzare i correlati

```
{
    "id": "7e52bbc6379463aa0a6776d117b5fffd"
}
```

Output

```
[
    {
        "_id": "a2a717eaff1c4f504dc36dc908285207",
        "title": "How does the immune system work?",
        "url": "https://www.ted.com/talks/emma_bryce_how_does_the_immune_system_work"
    },
    {
        "_id": "c4b0ade4a4862ecfc44b19adb7db339d",
        "title": "Why can't we talk about periods?",
        "url": "https://www.ted.com/talks/jen_gunter_why_can_t_we_talk_about_periods"
    },
    {
        "_id": "3d09fa5a4a64a82554252244c9420355",
        "title": "6 tips for better sleep",
        "url": "https://www.ted.com/talks/matt_walker_6_tips_for_better_sleep"
    }
]
```

Get_Top_Tags

La funzione **Get_Top_Tags** restituisce una lista dei tag più popolari.

Funzionamento

La funzione crea una lista dei tag estratti dai primi *n* video (*n* = ***talk_depth***) ordinati per *"**avg_points**"* (Media ponderata tra numero di visualizzazioni e like del video).

La lista viene analizzata e i tag vengono ordinati in base al numero di ripetizioni all'interno della lista stessa.

Alla fine si estraggono i primi ***n_tags***

```
connect_to_db().then(async () => {
    try{
        let tags_list = [];
        let talks_list = await talksDB.find({},{tags:1,_id:0}).sort({avg_points:-1}).limit(body.talks_depth);

        talks_list.forEach((talk) =>{
           talk.tags.forEach((tag) =>{
                tags_list.push(tag);
            });
        })

        // creazione di un oggetto contenente i tag (chiave) e il numero di ripetizioni (valore)
        const map = {};
        for (const num of tags_list){
            map[num] = map[num] ? map[num] + 1 : 1;
        }

        // ordinamento per numero di ripetizioni
        const mapSort = new Map([...Object.entries(map)].sort((a, b) => b[1] - a[1]));

        // creo un array composto dai tag ordinati
        let top_tags = Array.from(mapSort.keys());
        // estraggo i primi n tag
        top_tags = top_tags.slice(0, body.n_tags)

        callback(null, {
            statusCode: 200,
            body: JSON.stringify(top_tags)
        })
    }
    catch(err){
        callback(null, {
            statusCode: err.statusCode || 500,
            headers: { 'Content-Type': 'text/plain' },
            body: 'Could not fetch the next talk data.'
        });
    }
});
```

Get_top_tags - API Request

Input:

- talks_depth: numero di talk da analizzare
- n_tags: numero di tag che si desidera osservare

```
{
    "talks_depth" : 5000,
    "n_tags": 10
}
```

Output

- n_tags di tendenza più popolari

```
[
    "TED",
    "talks",
    "technology",
    "science",
    "culture",
    "TED-Ed",
    "society",
    "animation",
    "TEDx",
    "global issues"
]
```

Get_Top_Trending_Videos

La funzione **Get_Top_Trending_Videos** restituisce la lista dei dati (*"_id"*, *"title"*, *"url"*, *"avg_points"*) dei primi *num_video* talk ordinati per "*avg_points*", quindi quelli più popolari.

```
connect_to_db().then(async () => {
    try{
        let top_trending_list = await talksModel
            .find({},{_id:1, title: 1, url:1, avg_points:1})
            .sort({avg_points:-1})
            .limit(body.num_video);

        callback(null, {
            statusCode: 200,
            body: JSON.stringify(top_trending_list)
        })
    }
    catch(err){
        callback(null, {
            statusCode: err.statusCode || 500,
            headers: { 'Content-Type': 'text/plain' },
            body: 'Could not fetch the next talk data.'
        });
    }
});
```

Input:

- **num_video:** numero di talk da mostrare

```
{
    "num_video": 3
}
```

Output


```
[
    {
        "_id": "b89b3443794b2f415caa0acee1a12976",
        "title": "Do schools kill creativity?",
        "url": "https://www.ted.com/talks/sir_ken_robinson_do_schools_kill_creativity",
        "avg_points": 100
    },
    {
        "_id": "9b2621fa773abe4dd1786e65dec367bf",
        "title": "Your body language may shape who you are",
        "url": "https://www.ted.com/talks/amy_cuddy_your_body_language_may_shape_who_you_are",
        "avg_points": 89.80528982497745
    },
    {
        "_id": "dc83e0c23998fc4ecb077fcbb3e5bf6d",
        "title": "Inside the mind of a master procrastinator",
        "url": "https://www.ted.com/talks/tim_urban_inside_the_mind_of_a_master_procrastinator",
        "avg_points": 83.1614266253372
    }
]
```

Funzionalità utente

Get_Watch_Next_by_Idx

Quando l'utente termina la visione di un video, gli verrà mostrata una lista di video correlati.

Get_Top_Tags

L'utente potrà visualizzare la lista dei tag più popolari tra i video di tendenza.

Interagendo con essi gli verrà mostrata una classifica dei video più popolari associati a tale tag.

Get_Top_Trending_Videos

L'utente potrà visualizzare una classifica dei video più visti e apprezzati dagli utenti.

Criticità

- La presenza dei tag "*TED*" e "*talks*" in ogni video, alterano le classifiche dei tag di tendenza.
- Uno o più tag potrebbero non essere del tutto correlati al video associato
- La presenza di valori nulli relativi al numero di visualizzazioni di alcuni talk porta a un ordinamento errato dei talk correlati
- Non siamo riusciti ad implementare la funzione "**Get_Top_Trending_Words"** a causa delle restrizioni che impedivano l'uso di Amazon Comprehend.

Possibili Evoluzioni

- Rimuovere la presenza dei tag "*TED*" e "*talks*" durante la classificazione dei tag di tendenza.
- Ottenere un'autorizzazione che ci permetta di implementare la funzione "**Get_Top_Trending_Words"** garantendo l'uso di Amazon Comprehend.
- Implementare nuove strategie per la ricerca dei video correlati, per esempio basate su tag e/o titoli pertinenti.

Link utili