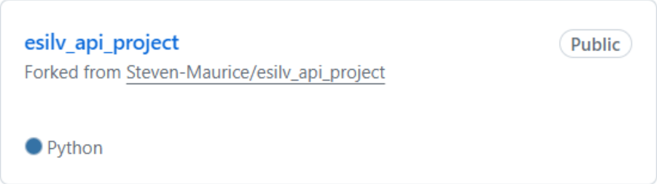
Advanced Python Git Linux Project

The first part of this project was under Linux. We created a branch named after the names of the group members. This branch comes directly from the Github repository Esilv_Api_Project created on purpose.

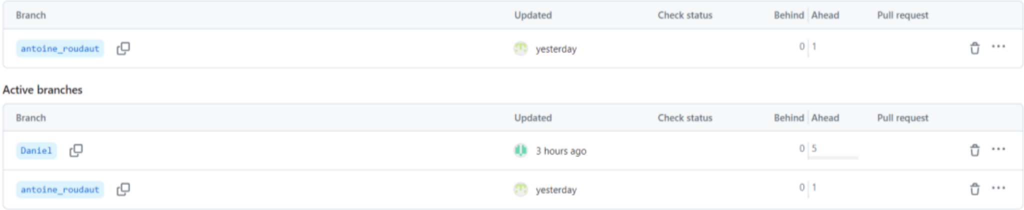


esilv_api_project

Public

Forked from Steven-Maurice/esilv_api_project

Python



Branch	Updated	Check status	Behind	Ahead	Pull request
antoine_roudaut	yesterday		0	1	

Active branches

Branch	Updated	Check status	Behind	Ahead	Pull request
Daniel	3 hours ago		0	5	
antoine_roudaut	yesterday		0	1	

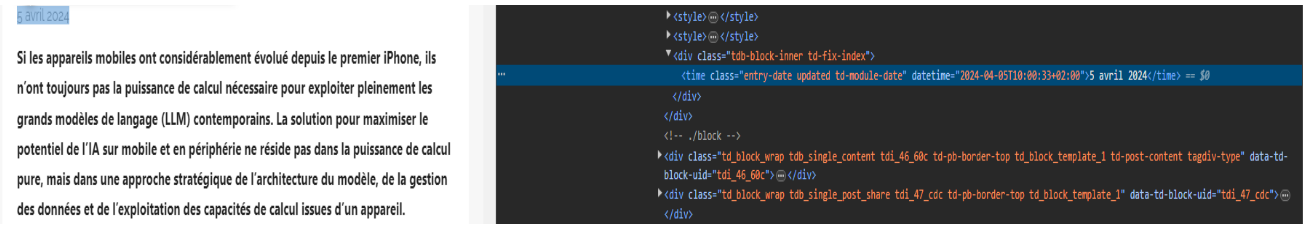
Inside this branch we created a working directory named after the site we choose : ActuIA.

The goal of our API AlmightIA : For the user, it could analyze a website that contains articles; these articles are to be treated by our API. This API could be some kind of a notebook that will allow us to sort articles that we find interesting or not for any purpose of ours.

I] Fetching data

The second part of the project was mainly in Python. As a reminder, the aim of this project was to retrieve information from a chosen site and create endpoints for different purposes. Our first task was to retrieve the data from the site we had chosen. This was quite complicated because it was something we had never done before and was completely new to us; hence, we worked on a different .py file to test some data fetching methods. We chose to obtain our data by scrapping and using the BeautifulSoup library. Thanks to this library, we were able to obtain our data on our site.

In order to get the corresponding tags, we used “CTRL+Shift+C”.



5 avril 2024

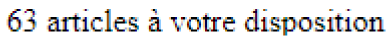
Si les appareils mobiles ont considérablement évolué depuis le premier iPhone, ils n'ont toujours pas la puissance de calcul nécessaire pour exploiter pleinement les grands modèles de langage (LLM) contemporains. La solution pour maximiser le potentiel de l'IA sur mobile et en périphérie ne réside pas dans la puissance de calcul pure, mais dans une approche stratégique de l'architecture du modèle, de la gestion des données et de l'exploitation des capacités de calcul issues d'un appareil.

```
▸<style>⋯</style>
▸<style>⋯</style>
▾<div class="tdb-block-inner td-fix-index">
   <time class="entry-date updated td-module-date" datetime="2024-04-05T10:00:33+02:00">5 avril 2024</time> == $0
  </div>
</div>
<!-- ./block -->
▸<div class="td_block_wrap tdb_single_content tdi_46_60c td-pb-border-top td_block_template_1 td-post-content tagdiv-type" data-td-block-uid="tdi_46_60c">⋯</div>
▸<div class="td_block_wrap tdb_single_post_share tdi_47_cdc td-pb-border-top td_block_template_1" data-td-block-uid="tdi_47_cdc">⋯
</div>
```

This approach may not be optimal because, for some articles, the tags are not the same (we adapted our code to the 2 cases that came up the most in this website…).

II] Focus on the API

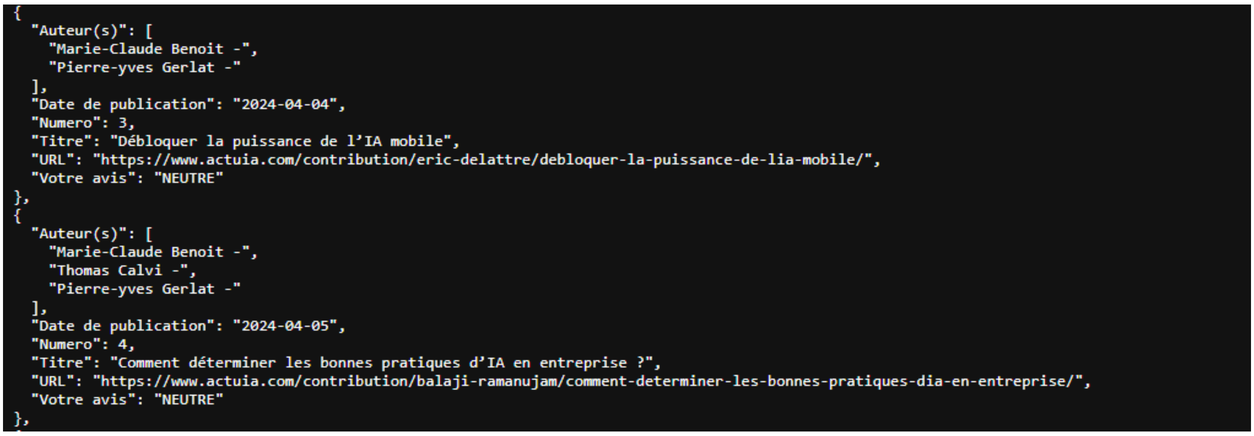
First of all, we created a function “articles_dispo” that retrieves the number of articles offered on the site. This is important because the user is asked to enter the number of articles they wish to view, and this number cannot exceed the number of articles available.



63 articles à votre disposition

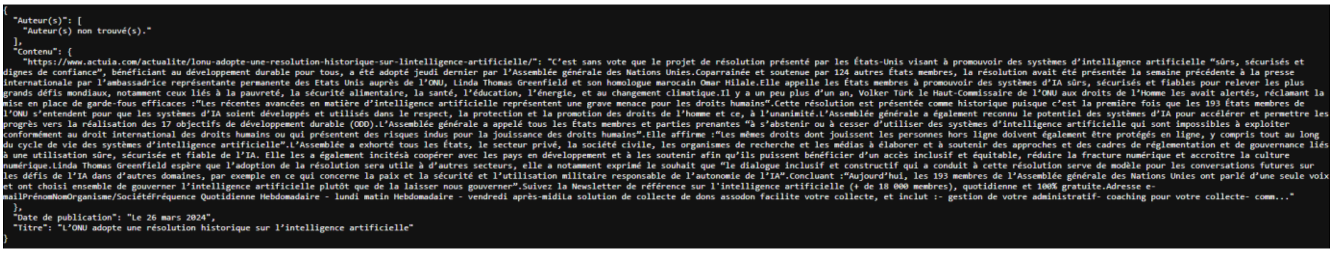
/articles : After this, we created one of the most important function; “scrap_me” which takes one argument: the number of articles we want. Inside the function we put the URL of our site and we take all the available URLs. We now have a list of URLs and depending on the number of articles the user wants, we go to these URLs and combine: the author, the date, the title and the URL. We store all these results in a list of dictionaries whose keys are the URL and whose value is the data being searched for.

These informations were displayed in this route. Moreover, we added a key in the dictionary that would refer to the user’s opinion about the article.



```
{
  "Auteur(s)": [
    "Marie-Claude Benoit -",
    "Pierre-yves Gerlat -"
  ],
  "Date de publication": "2024-04-04",
  "Numero": 3,
  "Titre": "Débloquer la puissance de l’IA mobile",
  "URL": "https://www.actuia.com/contribution/eric-delattre/debloquer-la-puissance-de-lia-mobile/",
  "Votre avis": "NEUTRE"
},
{
  "Auteur(s)": [
    "Marie-Claude Benoit -",
    "Thomas Calvi -",
    "Pierre-yves Gerlat -"
  ],
  "Date de publication": "2024-04-05",
  "Numero": 4,
  "Titre": "Comment déterminer les bonnes pratiques d’IA en entreprise ?",
  "URL": "https://www.actuia.com/contribution/balaji-ramanujam/comment-determiner-les-bonnes-pratiques-dia-en-entreprise/",
  "Votre avis": "NEUTRE"
},
```

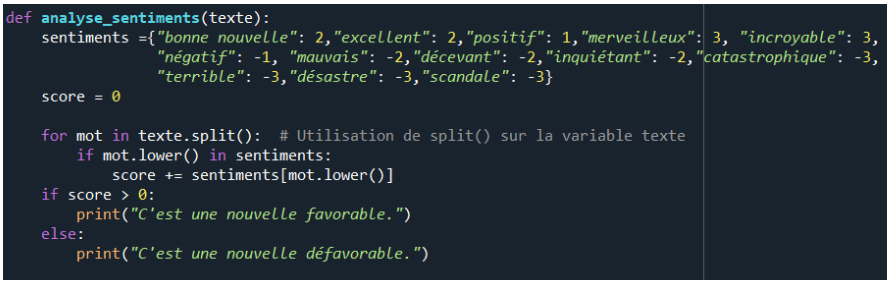
/articles/<number> : We created another function “scrapping_content” which takes in argument an URL and gives in outpout the content of the article.



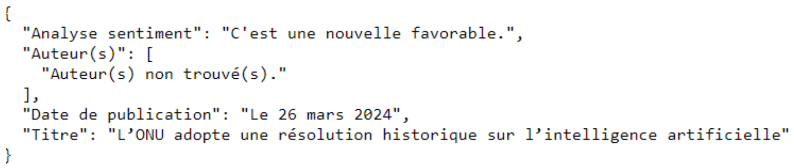
```
{
  "Auteur(s)": [
    "Auteur(s) non trouvé(s)."
  ],
  "Contenu": {
    "https://www.actuia.com/actualite/lonu-adopte-une-resolution-historique-sur-lintelligence-artificielle/": "C’est sans vote que le projet de résolution présenté par les États-Unis visant à promouvoir des systèmes d’intelligence artificielle “sûrs, sécurisés et
dignes de confiance”, bénéficiant au développement durable pour tous, a été adopté jeudi dernier par l’Assemblée générale des Nations Unies.Coparrainée et soutenue par 124 autres États membres, la résolution avait été présentée la semaine précédente à la presse
internationale par l’ambassadrice représentante permanente des Etats Unis auprès de l’ONU, Linda Thomas Greenfield et son homologue marocain Omar Hilale.Elle appelle les États membres à promouvoir des systèmes d’IA sûrs, sécurisés et fiables pour relever les plus
grands défis mondiaux, notamment ceux liés à la pauvreté, la sécurité alimentaire, la santé, l’éducation, l’énergie, et au changement climatique.Il y a un peu plus d’un an, Volker Türk le Haut-Commissaire de l’ONU aux droits de l’Homme les avait alertés, réclamant la
mise en place de garde-fous efficaces :“Les récentes avancées en matière d’intelligence artificielle représentent une grave menace pour les droits humains”.Cette résolution est présentée comme historique puisque c’est la première fois que les 193 États membres de
l’ONU s’entendent pour que les systèmes d’IA soient développés et utilisés dans le respect, la protection et la promotion des droits de l’homme et ce, à l’unanimité.L’Assemblée générale a également reconnu le potentiel des systèmes d’IA pour accélérer et permettre les
progrès vers la réalisation des 17 objectifs de développement durable (ODD).L’Assemblée générale a appelé tous les États membres et parties prenantes “à s’abstenir ou à cesser d’utiliser des systèmes d’intelligence artificielle qui sont impossibles à exploiter
conformément au droit international des droits humains ou qui présentent des risques indus pour la jouissance des droits humains”.Elle affirme :“Les mêmes droits dont jouissent les personnes hors ligne doivent également être protégés en ligne, y compris tout au long
du cycle de vie des systèmes d’intelligence artificielle”.L’Assemblée a exhorté tous les États, le secteur privé, la société civile, les organismes de recherche et les médias à élaborer et à soutenir des approches et des cadres de réglementation et de gouvernance liés
à une utilisation sûre, sécurisée et fiable de l’IA. Elle les a également incitésà coopérer avec les pays en développement et à les soutenir afin qu’ils puissent bénéficier d’un accès inclusif et équitable, réduire la fracture numérique et accroître la culture
numérique.Linda Thomas Greenfield espère que l’adoption de la résolution sera utile à d’autres secteurs, elle a notamment exprimé le souhait que “le dialogue inclusif et constructif qui a conduit à cette résolution serve de modèle pour les conversations futures sur
les défis de l’IA dans d’autres domaines, par exemple en ce qui concerne la paix et la sécurité et l’utilisation militaire responsable de l’autonomie de l’IA”.Concluant :“Aujourd’hui, les 193 membres de l’Assemblée générale des Nations Unies ont parlé d’une seule voix
et ont choisi ensemble de gouverner l’intelligence artificielle plutôt que de la laisser nous gouverner”.Suivez la Newsletter de référence sur l'intelligence artificielle (+ de 18 000 membres), quotidienne et 100% gratuite.Adresse e-
mailPrénomNomOrganisme/SociétéFréquence Quotidienne Hebdomadaire - lundi matin Hebdomadaire - vendredi après-midiLa solution de collecte de dons assodon facilite votre collecte, et inclut :- gestion de votre administratif- coaching pour votre collecte- comm..."
  },
  "Date de publication": "Le 26 mars 2024",
  "Titre": "L’ONU adopte une résolution historique sur l’intelligence artificielle"
}
```

With these 3 functions we were already meeting 3 of the expectations of the subject: Fetches a list of articles from the site, displays information about the articles, including the article number, title, publication date and accesses the content of a specified article.

/articles/<number>/ml : The last part was to apply a machine learning script on the data. We decided to use feelings analysis on the content of the articles. The function takes in argument a string and outputs if the article is favorable or unfavorable news for IA world. To do this we used a dictionary with as key words (as much as we could) and as values weights and depending off the meaning of each words. Here our first version with a small dictionary (it evolved since).



```python
def analyse_sentiments(texte):
    sentiments ={"bonne nouvelle": 2,"excellent": 2,"positif": 1,"merveilleux": 3, "incroyable": 3,
                "négatif": -1, "mauvais": -2,"décevant": -2,"inquiétant": -2,"catastrophique": -3,
                "terrible": -3,"désastre": -3,"scandale": -3}
    score = 0

    for mot in texte.split():  # Utilisation de split() sur la variable texte
        if mot.lower() in sentiments:
            score += sentiments[mot.lower()]
    if score > 0:
        print("C'est une nouvelle favorable.")
    else:
        print("C'est une nouvelle défavorable.")
```

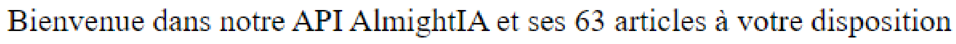


```
{
  "Analyse sentiment": "C'est une nouvelle favorable.",
  "Auteur(s)": [
    "Auteur(s) non trouvé(s)."
  ],
  "Date de publication": "Le 26 mars 2024",
  "Titre": "L’ONU adopte une résolution historique sur l’intelligence artificielle"
}
```

With these functions we met all the expectations of the subject and the last part was to create and defining the routes off our API.

Several routes have been created for different uses,

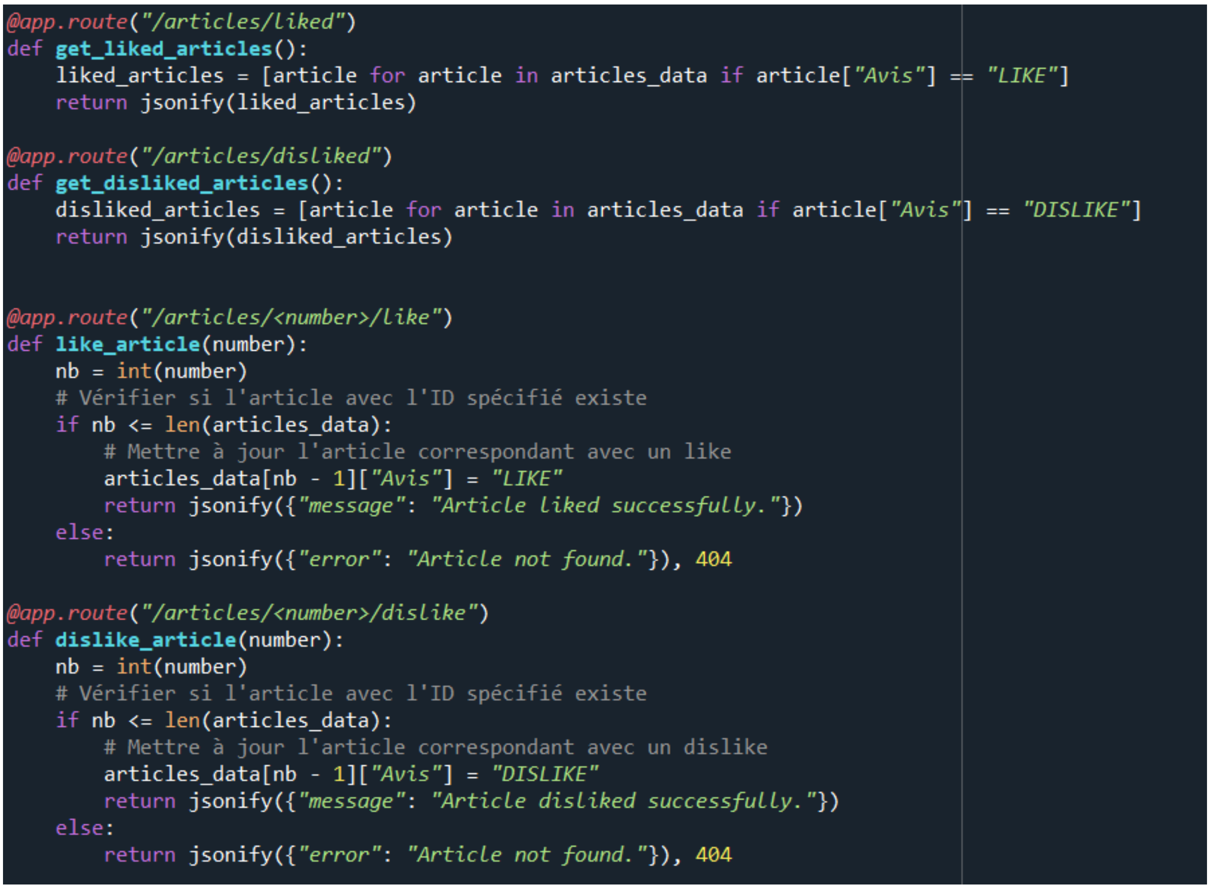
/ : the first one is to access to the API



Bienvenue dans notre API AlmightIA et ses 63 articles à votre disposition

The second route is to get the data from the website. Depending how many articles the user wants our API scrap it from the website site and give him in the asked form. The third one is to get the content of the specific article by typing the number associate to.

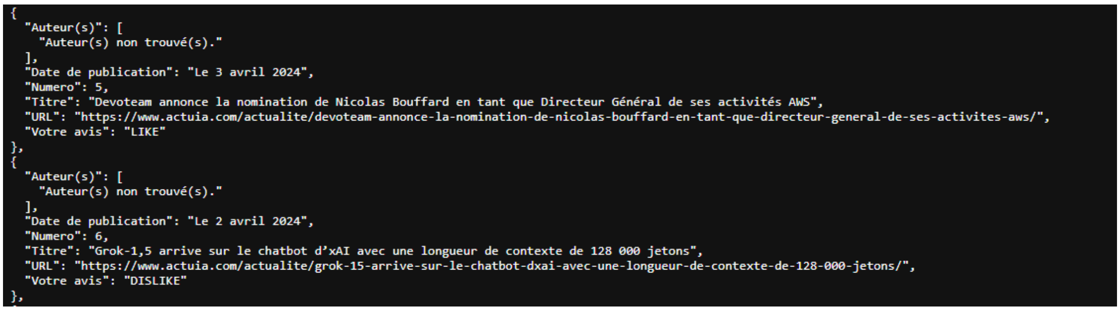
But we decided to go further by creating additional routes. Thus our API is also able to saved and displays the articles liked and disliked by the user.



```python
@app.route("/articles/liked")
def get_liked_articles():
    liked_articles = [article for article in articles_data if article["Avis"] == "LIKE"]
    return jsonify(liked_articles)

@app.route("/articles/disliked")
def get_disliked_articles():
    disliked_articles = [article for article in articles_data if article["Avis"] == "DISLIKE"]
    return jsonify(disliked_articles)


@app.route("/articles/<number>/like")
def like_article(number):
    nb = int(number)
    # Vérifier si l'article avec l'ID spécifié existe
    if nb <= len(articles_data):
        # Mettre à jour l'article correspondant avec un like
        articles_data[nb - 1]["Avis"] = "LIKE"
        return jsonify({"message": "Article liked successfully."})
    else:
        return jsonify({"error": "Article not found."}), 404

@app.route("/articles/<number>/dislike")
def dislike_article(number):
    nb = int(number)
    # Vérifier si l'article avec l'ID spécifié existe
    if nb <= len(articles_data):
        # Mettre à jour l'article correspondant avec un dislike
        articles_data[nb - 1]["Avis"] = "DISLIKE"
        return jsonify({"message": "Article disliked successfully."})
    else:
        return jsonify({"error": "Article not found."}), 404
```

/articles/<number>/like (or dislike) : Allows us to like or dislike the corresponding article

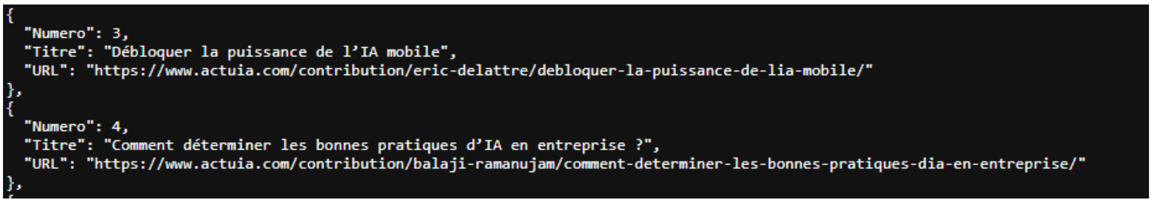
Here’s an example of our output after we liked or disliked an article. (on route **/articles**)



```
{
  "Auteur(s)": [
    "Auteur(s) non trouvé(s)."
  ],
  "Date de publication": "Le 3 avril 2024",
  "Numero": 5,
  "Titre": "Devoteam annonce la nomination de Nicolas Bouffard en tant que Directeur Général de ses activités AWS",
  "URL": "https://www.actuia.com/actualite/devoteam-annonce-la-nomination-de-nicolas-bouffard-en-tant-que-directeur-general-de-ses-activites-aws/",
  "Votre avis": "LIKE"
},
{
  "Auteur(s)": [
    "Auteur(s) non trouvé(s)."
  ],
  "Date de publication": "Le 2 avril 2024",
  "Numero": 6,
  "Titre": "Grok-1,5 arrive sur le chatbot d’xAI avec une longueur de contexte de 128 000 jetons",
  "URL": "https://www.actuia.com/actualite/grok-15-arrive-sur-le-chatbot-dxai-avec-une-longueur-de-contexte-de-128-000-jetons/",
  "Votre avis": "DISLIKE"
},
```

/articles/liked (or disliked) : Displays the liked or disliked articles

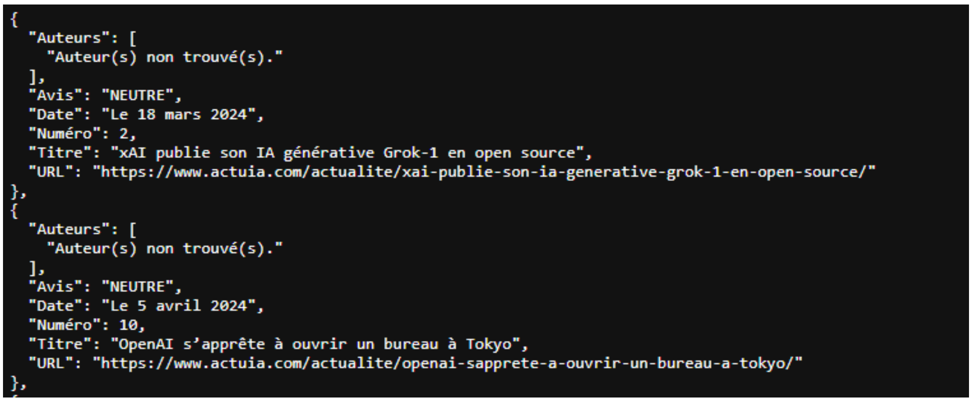
/get_data : displays a list of articles that we have (no details except number and URL)



```
{
  "Numero": 3,
  "Titre": "Débloquer la puissance de l’IA mobile",
  "URL": "https://www.actuia.com/contribution/eric-delattre/debloquer-la-puissance-de-lia-mobile/"
},
{
  "Numero": 4,
  "Titre": "Comment déterminer les bonnes pratiques d’IA en entreprise ?",
  "URL": "https://www.actuia.com/contribution/balaji-ramanujam/comment-determiner-les-bonnes-pratiques-dia-en-entreprise/"
},
```

/get_data/keyword/<word> : allows the user to display a list of articles that matches with a specific keyword

For example, if word == “générative”,



```
{
  "Auteurs": [
    "Auteur(s) non trouvé(s)."
  ],
  "Avis": "NEUTRE",
  "Date": "Le 18 mars 2024",
  "Numéro": 2,
  "Titre": "xAI publie son IA générative Grok-1 en open source",
  "URL": "https://www.actuia.com/actualite/xai-publie-son-ia-generative-grok-1-en-open-source/"
},
{
  "Auteurs": [
    "Auteur(s) non trouvé(s)."
  ],
  "Avis": "NEUTRE",
  "Date": "Le 5 avril 2024",
  "Numéro": 10,
  "Titre": "OpenAI s’apprête à ouvrir un bureau à Tokyo",
  "URL": "https://www.actuia.com/actualite/openai-sapprete-a-ouvrir-un-bureau-a-tokyo/"
},
```

Conclusion To sum up, we created a branch directly from the github repository Esilv_Api_Project. After this we did the part relative to web scrapping and formatting data in the manner requested. After did this we created the route of our API and we were able to push all of our work in the branch under git.