

Exportations de Données: Twitter

Page

Les métriques au niveau de la page vous donneront des données agrégées à un niveau quotidien, afin que vous puissiez regarder en arrière et voir les tendances qui se produisent sur votre page au fil du temps.

Tweet

Les métriques au niveau du tweet montrent l'étendue et la force de l'influence d'un tweet, ce qui vous permet d'analyser le contenu que vous diffusez et de déterminer dans quelle mesure il trouve un écho non seulement auprès de votre communauté, mais aussi auprès du public potentiel accessible par l'intermédiaire de votre communauté.

Hashtag

Les métriques au niveau du hashtag indiquent le degré d'attention et d'engagement qu'un certain sujet/événement est capable d'évoquer.

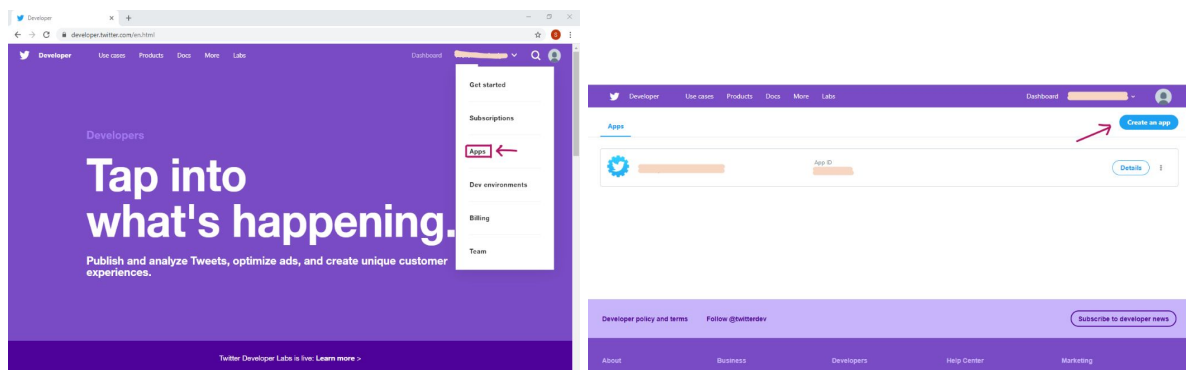
Médias

Les métriques au niveau de la vidéo vous aident à comprendre comment le contenu des médias contribue à votre portée et à votre engagement global.

Comment extraire les tweets et les commentaires ?

Les commentaires de Twitter, ainsi que d'autres données de Twitter, peuvent être extraits en utilisant le compte d'un développeur de Twitter et R.

Tout d'abord, demandez un compte de développeur Twitter à l'adresse <https://developer.twitter.com> et suivez les étapes requises dans la demande. Une fois la demande approuvée, visitez votre page de développeur et créez une application.



Developer

Use cases

Products

Docs

More

Learn

Dashboard

+

callback, leave these blank.

X

+ Add another

Terms of Service URL

Privacy policy URL

Organization name

Organization website URL

Tell us how this app will be used (required)

This field is only visible to Twitter employees. Help us understand how your app will be used. What will it enable you and your customers to do?

Please be detailed.

Cancel

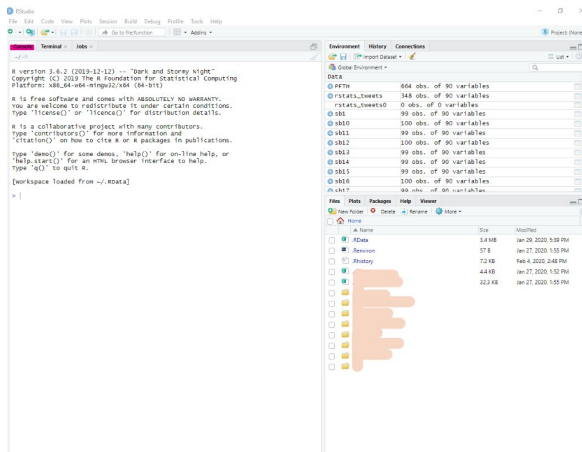
Create

Minimum characters 100

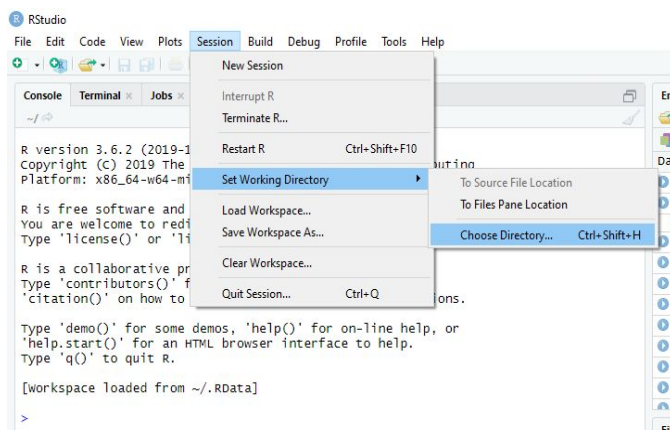
Enfin, rassemblez les clés de l'interface de programmation d'applications (API: Application Programming interface) sous la rubrique "Keys and tokens". Générez votre jeton d'accès et votre secret de jeton d'accès. Les quatre clés sur cette page seront utilisées, donc gardez-les quelque part.

Créer un lien entre R et Twitter

Ouvrez studio R. Maintenant, vous voulez utiliser votre statut de développeur Twitter par le biais des commandes R, donc vous donnez à R l'accès à votre compte de développeur.



*Ceci est l'interface du studio R. L'onglet "Console" est l'endroit où vous tapez vos commandes et où vous obtenez les commentaires de R. Mais d'abord, pour faciliter votre futur travail, créez un répertoire de travail où vous avez l'intention de stocker tous les documents liés.



Maintenant que les réglages de base sont faits, vous pouvez commencer à vous occuper de Twitter. Cliquez dans la console et commencez à taper vos commandes. Pour terminer et exécuter une commande, appuyez sur la touche Enter Key.

Installez le paquet "rtweet" et téléchargez-le

```
install.packages("rtweet")
```

```
library(rtweet)
```

Stocker toutes les clés API

```
api_key <- "..."
```

```
api_secret_key <- "..."
```

```
access_token <- "..."
```

```
access_token_secret <- "..."
```

#remplacez les "..." avec les clés et jetons correspondants.

Authentifiez la connexion

```
token <- create_token (
```

```
app = "NAMEOFYOURAPP", #utiliser le nom que vous avez saisi pour  
votre application
```

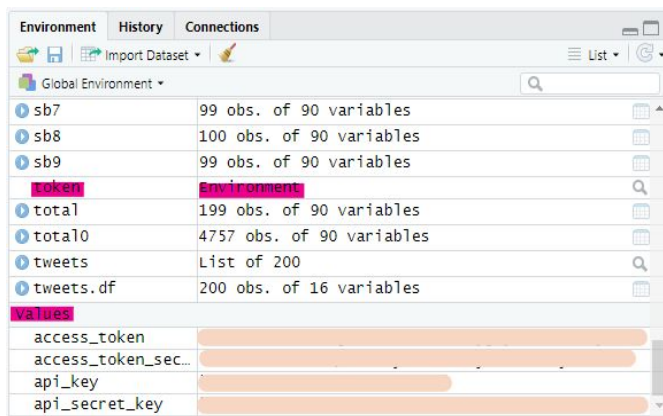
```
consumer_key = api_key,
```

```
consumer_secret = api_secret_key,
```

```
access_token = access_token,
```

```
access_secret = access_token_secret)
```

Cela créera un environnement dans l'espace de travail appelé "token", et vous devriez voir les valeurs de vos clés API dans l'onglet "Environment" dans la partie supérieure droite du studio R. Cela signifie qu'une connexion entre R et Twitter a été établie avec succès !



Extraire les tweets pertinents (approches régulières)

Pour rechercher des tweets en rapport avec un sujet (par exemple selon les hashtags)

```
DF <- search_tweets (q="VOTREMOTCLÉ", n=NOMBRE DE TWEETS)
```

Vous aurez une trame de données qui contient les données demandées. Mais seuls les tweets récents (6-9 jours) peuvent être extraits avec le produit standard API! Pour tout extraire, il est NÉCESSAIRE de passer à un paquet API plus avancé. Voir les approches avancées. Vous pouvez cliquer sur la trame de données dans l'onglet "Environment" pour visualiser l'ensemble des données.

Alternativement, pour extraire les tweets et les réponses d'un certain compte

```
DF<-get_timeline(q="@NOMD'ÉCRAN", n=NOMBRE DE TWEETS)
```

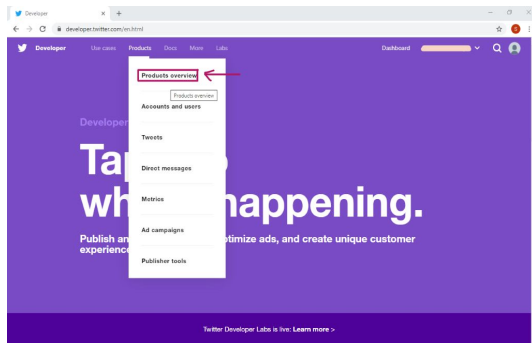
Notez que vous devez faire une recherche par le nom d'écran (celui qui commence par @), et NON par le nom de la page.

Vous pouvez régler le n sur un nombre plus grand pour faciliter la recherche. Cependant, il y a une limite supérieure de 3 200 tweets.

Si vous souhaitez extraire des commentaires historiques plutôt que des tweets, il n'y a pas de moyen direct de le faire avec des approches régulières. Mais vous pouvez vous tourner vers des approches avancées dans la section suivante.

Extraire des commentaires (approches avancées)

Les approches régulières sont pratiques, mais il y a des limites évidentes lorsque vous voulez extraire des commentaires. La première approche vous donne une fiche contenant des commentaires, mais elle est strictement limitée dans le temps ; la seconde approche ne vous permet pas de voir les commentaires des autres utilisateurs en dehors du compte qui vous intéresse. Par conséquent, vous voudrez peut-être passer à une boîte à outils de développeur avancée. La API Sandbox peut être un bon choix car elle est GRATUITE (la mauvaise nouvelle est qu'elle nécessite plus d'entrée manuelle si vous prévoyez avoir des données volumineuses). Pour ce faire, rendez-vous sur la page web de développeur et cliquez sur l'onglet Products. La Sandbox existe en deux versions : 30-Days et archive complète. Choisissez l'une (ou les deux) en fonction de vos besoins et donnez un label (nom aléatoire) au(x) nouvel(aux) environnement(s) de développement.



Products / Overview

APIs and tools to tap into what's happening

The Twitter Developer Platform provides the tools you need to build products with Twitter data, create advertising campaigns on Twitter, and build Twitter into your products. Our data products, Ads API, and publisher tools make it easier to get the most out of Twitter.

Data products

Scale your Twitter data access

API Type	Access Level
Standard APIs	Free
Premium APIs	Pro
Enterprise APIs	Enterprise

Standard APIs

Our free, standard APIs are great for getting started, testing an integration, validating a concept, or creating solutions that complement what you can create with premium and enterprise products. Examples include posting content to Twitter and getting data not available in high volumes.

[Learn more](#)

[Documentation](#) >

[Forums](#) >

Premium APIs

Our premium APIs offer scalable access to Twitter data for those looking to grow, experiment, and innovate. When the standard API doesn't offer the amount of data necessary, upgrading to premium allows you to continue building and growing. Test in the free sandbox and then upgrade to month-to-month access.

[Learn more](#)

[Pricing](#) >

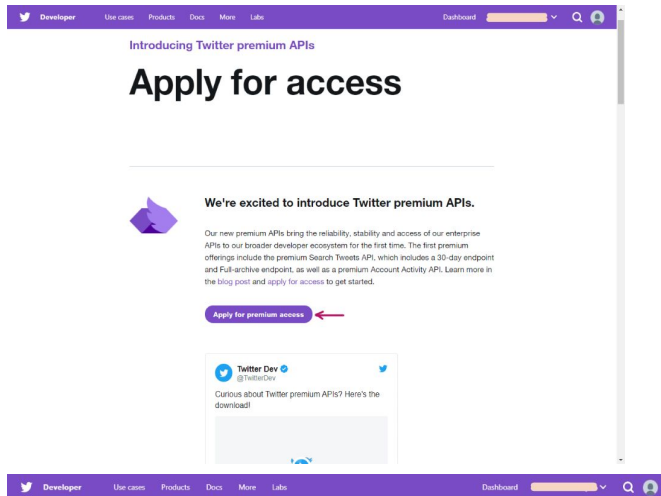
[Forums](#) >

Enterprise APIs

Our enterprise APIs offer the highest level of access and reliability to those who depend on Twitter data. Perfect as you scale beyond premium and need more reliable access, custom tailored packages, or annual contracts. Enterprise API access comes with dedicated account managers and technical support.

[Learn more](#)

[Contact enterprise sales](#) >



Get started with Twitter APIs and tools

Apply for access

All new developers must apply for a developer account to access Twitter APIs.

[Apply for a developer account](#) [Restricted use cases >](#)

Standard APIs

Our free, standard APIs are great for getting started, testing an integration, or validating a concept.

[Apply for a developer account >](#)

Premium APIs

Our premium APIs offer scalable access to Twitter data for those looking to grow, experiment, and innovate.

[Apply for premium access >](#)

Enterprise APIs

Our enterprise APIs offer the highest level of access and reliability to those who depend on Twitter data.

[Apply for enterprise access >](#)

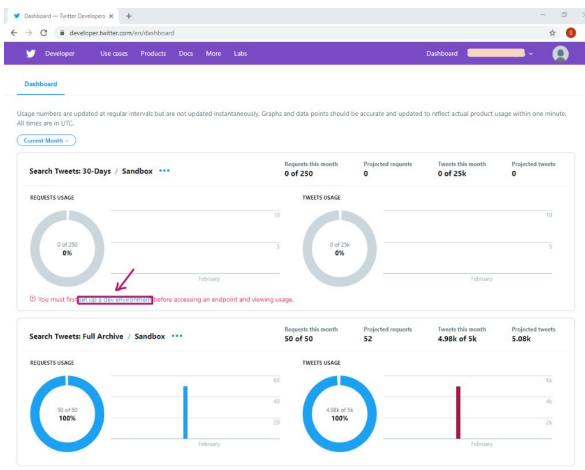
Ads APIs

The Ads API gives partners a programmatic way to integrate with the Twitter Ads platform.

[Apply for Ads APIs access >](#)

Maintenant que vous avez choisi votre ou vos produit(s), vous pouvez voir ce qu'ils vous permettront de faire sur le tableau de bord de développeur (<https://developer.twitter.com/en/dashboard>). C'est également là que vous verrez l'utilisation de vos produits.

Pour activer un produit, créez un environnement de développement et étiquetez-le.



This screenshot shows the 'Dev environments' section. It lists three environments: 'Search Tweets: 30-Days / Sandbox' (not set up), 'Search Tweets: Full Archive / Sandbox' (with fields for dev environment label, app name, and app owner), and 'Account Activity API / Sandbox' (not set up). A red arrow points to the 'Set up dev environment' button for the 30-Days Sandbox environment.

The dialog box 'Set up Search Tweets: 30-Days dev environment' contains two input fields: 'Dev environment label' with the value 'BeCreativeHere' (marked with a red '1') and 'App' with a dropdown menu showing 'Select an app...' (marked with a red '2'). At the bottom right, there are 'Cancel' and 'Complete setup' buttons, with a red arrow (marked with a red '3') pointing to the 'Complete setup' button.

L'exemple ci-dessus est basé sur Sandbox 30-Days, mais vous pouvez choisir en fonction de vos besoins et de votre budget. Sandbox est une offre premium, bien que gratuite.

Une fois la configuration terminée, vous serez redirigé vers le tableau de bord et vous remarquerez que la ligne en rouge "vous devez d'abord configurer un environnement de développement..." sous le produit que vous avez sélectionné a disparu. Cela signifie que vous avez réussi à activer le produit et que vous êtes prêt à procéder à l'extraction avancée !

Rechercher dans l'historique complet (archive complète)

```
DF<- search_fullarchive (q="@NOMD'ÉCRAN", n=100, fromDate =  
YYYYMMDDHHMM, toDate=YYYYMMDDHHMM, env_name = "VOTRELABEL")
```

vous pouvez demander des données 50 fois par mois, 100 tweets par demande, ce qui signifie que n ne doit pas être supérieur à 100.

vous n'avez pas toujours besoin d'avoir le terme fromDate dans votre commande, surtout lorsque vous remontez dans l'histoire à partir d'une date ultérieure pour 100 tweets.

Faites une recherche sur le mois écoulé (30-Days)

```
DF<- search_30day (q="@NOMD'ÉCRAN", n=100, fromDate =  
YYYYMMDDHHMM, toDate=YYYYMMDDHHMM, env_name = "VOTRELABEL")
```

Vous pouvez demander des données 250 fois par mois, 100 tweets par demande.

A nouveau, n<=100 et fromDate n'est pas toujours nécessaire.

Cela vous donne que des données du mois écoulé, et l'option d'archive complète peut faire de même pour vous. Mais c'est utile lorsque vous avez besoin de données du mois dernier de manière urgente et que vous savez qu'il y a eu plus de 5000 tweets où la page a été marquée ou à laquelle on a répondu.

Notez que les données que vous obtiendrez contiendront 1) des commentaires sous les tweets de la page, 2) des retweets des tweets de la page et 3) des tweets créés par d'autres utilisateurs où la page est taguée. Cependant, il n'est pas difficile de savoir lesquels sont des commentaires et de filtrer le reste (voir la section Filtrage pour plus de détails).

Avec les limites supérieures pour le nombre de tweets que vous pouvez obtenir par requête, vous devrez peut-être extraire plusieurs fois lorsqu'il y a trop de tweets. Dans ce cas, vous pouvez fusionner tous les ensembles de données en un grand ensemble bien ordonné. Voici l'un des moyens les plus simples de faire ceci.

```
FINALDATA<-rbind(DF1, DF2,..., DFn)
```

Transférer des données dans un fichier csv

installez le paquet "data.table" et le télécharger

```
install.packages("data.table")
```

```
library(data.table)
```

utilisez la fonction "fwrite".

```
fwrite(FINALDATA, file = "NOMCSV.csv")
```

C'est une façon de faire les choses. La méthode habituelle de `write.csv` ne fonctionne pas parce que la trame de données contient des variables de liste.

Méthode alternative (moins efficace) :

```
sapply(DF,class) #voir quelles sont les variables de type liste
```

```
unlist(DF$NOMDELACOLONNE) #délistier (ne pas les énumérer) ces variables OU
```

```
toString(DF$NOMDELACOLONNE) #transférer les variables sur le string-type (type de chaîne)
```

```
write.csv(DF,"CSVNAME.csv")
```

Le fichier csv sera automatiquement enregistré dans votre répertoire de travail. Vous avez maintenant vos données !