

O FUTURO DA COMUNICAÇÃO <na era digital>

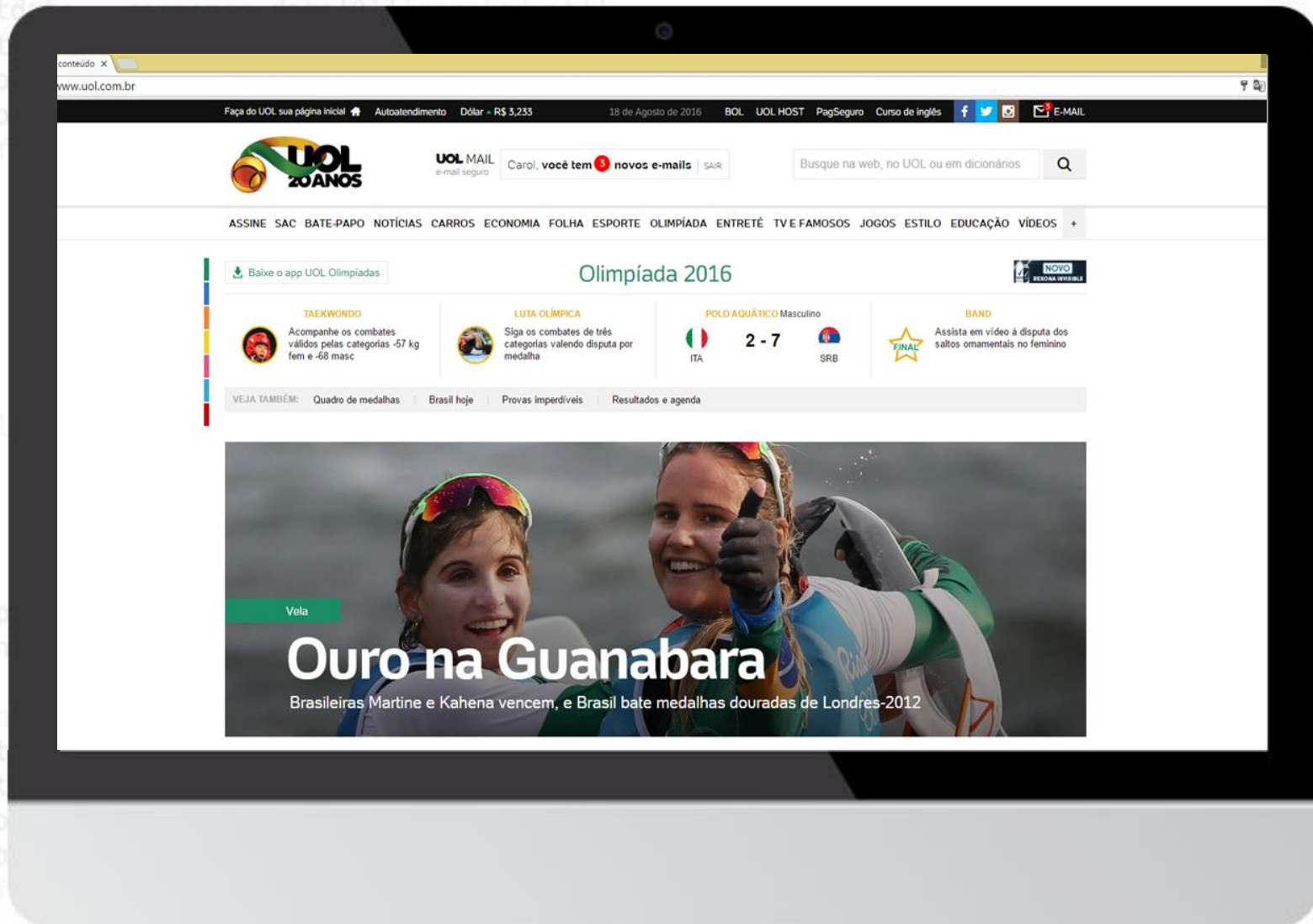


```
for i in people.data.users:
    response = client.api.statuses.user_timeline.get
    print 'Got', len(response.data), 'tweets from', i.screen_name
    if len(response.data) != 0:
        ldate = response.data[0]['created_at']
        ldate2 = datetime.strptime(ldate, '%a %b %d %H:%M:%S')
        today = datetime.now()
        howlong = (today-ldate2).days
        if howlong < daywindow:
            print i.screen_name, 'has tv
            totaltweets += len(response
            for j in response.data:
                if j.entities.urls:
                    for k in j.entities
                        newurl = k['exp
                        urlset.add((new
            print i.screen_name, 'has not tweeted in
            (i.screen_name))
            'the past',
for i in people.data.users:
    response = client.api.statuses.user_timeline.get(screen_name=
    print 'Got', len(response.data), 'tweets from', i.screen_name
    if len(response.data) != 0:
        ldate = response.data[0]['created_at']
        ldate2 = datetime.strptime(ldate, '%a %b %d %H:%M:%S')
        today = datetime.now()
        howlong = (today-ldate2).days
        if howlong < daywindow:
            print i.screen_name, 'has tweeted
            totaltweets += len(response.data)
            for j in response.data:
                if j.entities.urls:
```

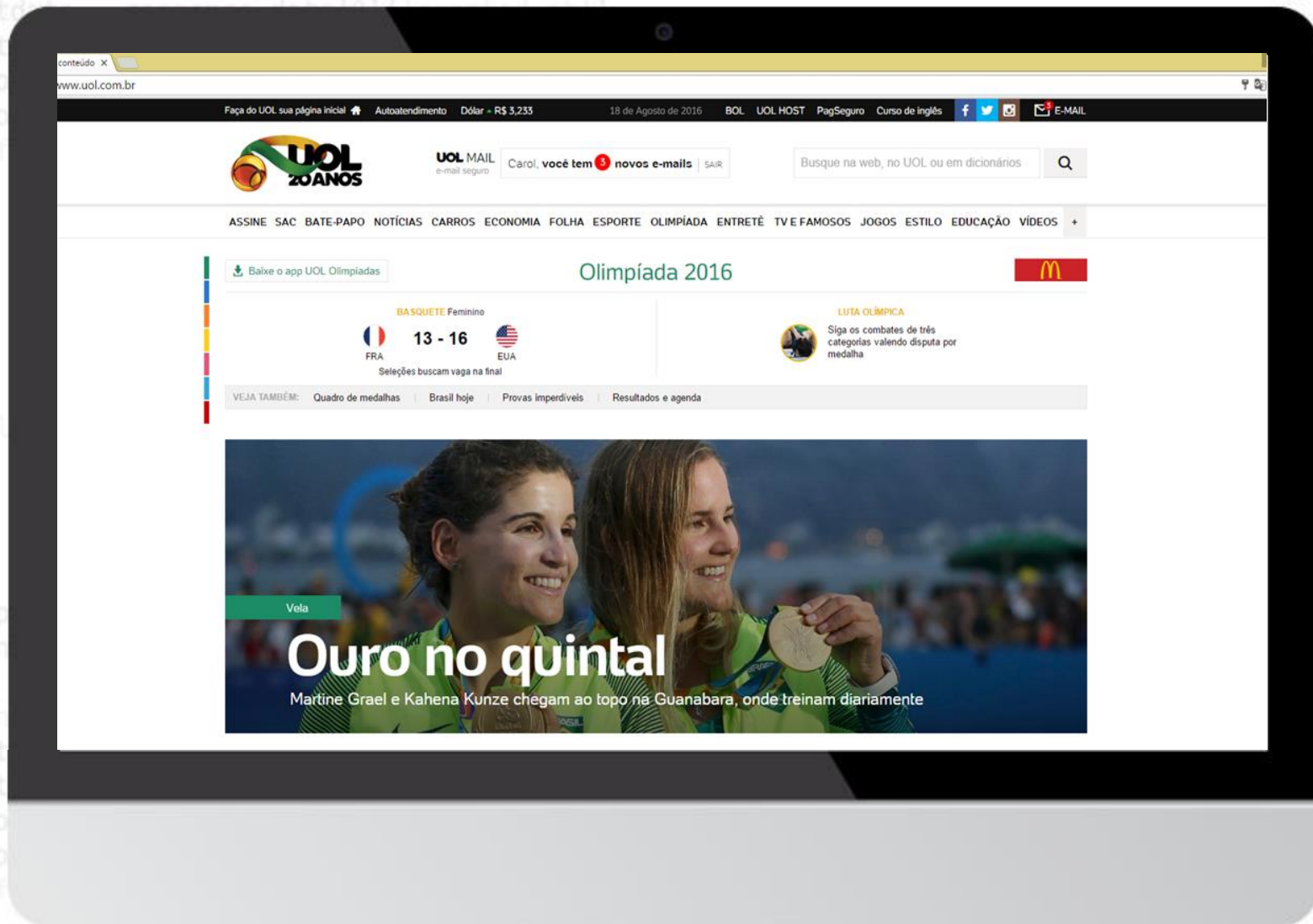
< 08 / 04 / 96 >

Primeira homepage do UOL





< Jornalismo
interage com o
usuário >



< 20 minutos
depois. >



o veículo
vai até
< o leitor >

```
1 in people.data.users:  
response = client.api.statuses.user_timeline.  
print 'Got', len(response.data), 'tweets from'  
if len(response.data) != 0:  
    ltdate = response.data[0]['created_at']  
    ltdate2 = datetime.strptime(ltdate, '%a %b'  
today = datetime.now()  
howlong = (today-ltdate2).days  
if howlong < daywindow:  
    print i.screen_name, 'has tweete  
totaltweets += len(response.data)  
for j in response.data:  
    if j.entities.urls:  
        for k in j.entities  
            newurl = k['exp  
            urlset.add((new  
            'has now tweete  
            (screen_name)')  
            the past',  
1 in people.data.users:  
response = client.api.statuses.user_timeline.  
print 'Got', len(response.data), 'tweets from'  
if len(response.data) != 0:  
    ltdate = response.data[0]['created_at']  
    ltdate2 = datetime.strptime(ltdate, '%a %b'  
today = datetime.now()  
howlong = (today-ltdate2).days  
if howlong < daywindow:  
    print i.screen_name, 'has tweete  
totaltweets += len(response.data)  
for j in response.data:  
    if j.entities.urls:  
        for k in j.entities.urls:
```

< leitor divulga > as matérias que mais gosta



```
for i in people.data.users:  
    response = client.api.statuses.user_timeline.get  
    print('Got', len(response.data), 'tweets from', i.screen_name  
    ldate2 = datetime.strptime(ltdate, '%a %b  
    today = datetime.now()  
    howlong = (today-ltdate2).days  
    if howlong < daywindow:  
        print i.screen_name, 'has  
        totaltweets += len(respo  
        for j in response.data:  
            if j.entities.urls  
                for k in j  
                    newurl  
    else:  
        print i.screen_name, 'tweeted in  
  
for i in peop  
response =  
print 'Got  
if len(res  
    ldate  
    ldate  
    today  
    howlo  
    if how  
        print  
        taltw  
        (data)  
    for j i  
        if
```



o usuário cria sua mídia

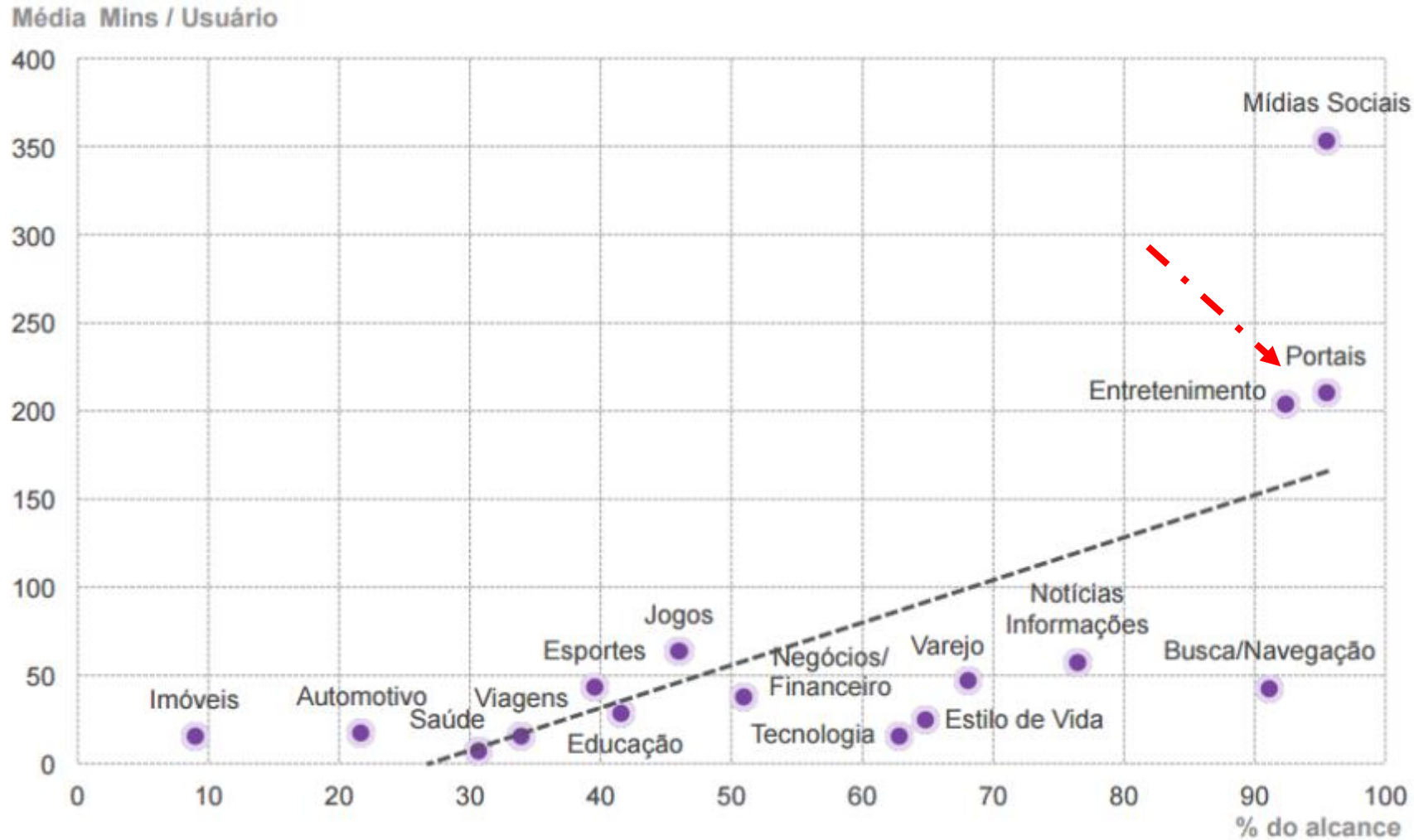




...e o ciberjornalismo observa
< os temas que circulam
nas redes sociais >

- Canal das fontes
- Canal da mídia
- Canal da audiência

< credibilidade importa >



O ciberjornalismo
fura

o filtro
bolha



< profissionalismo é cada vez mais necessário >



Ferramentas
adequadas



Relevância
jornalística
como baliza
para divulgação
de dados



Sigilo
da fonte



Vedação ao
anonimato

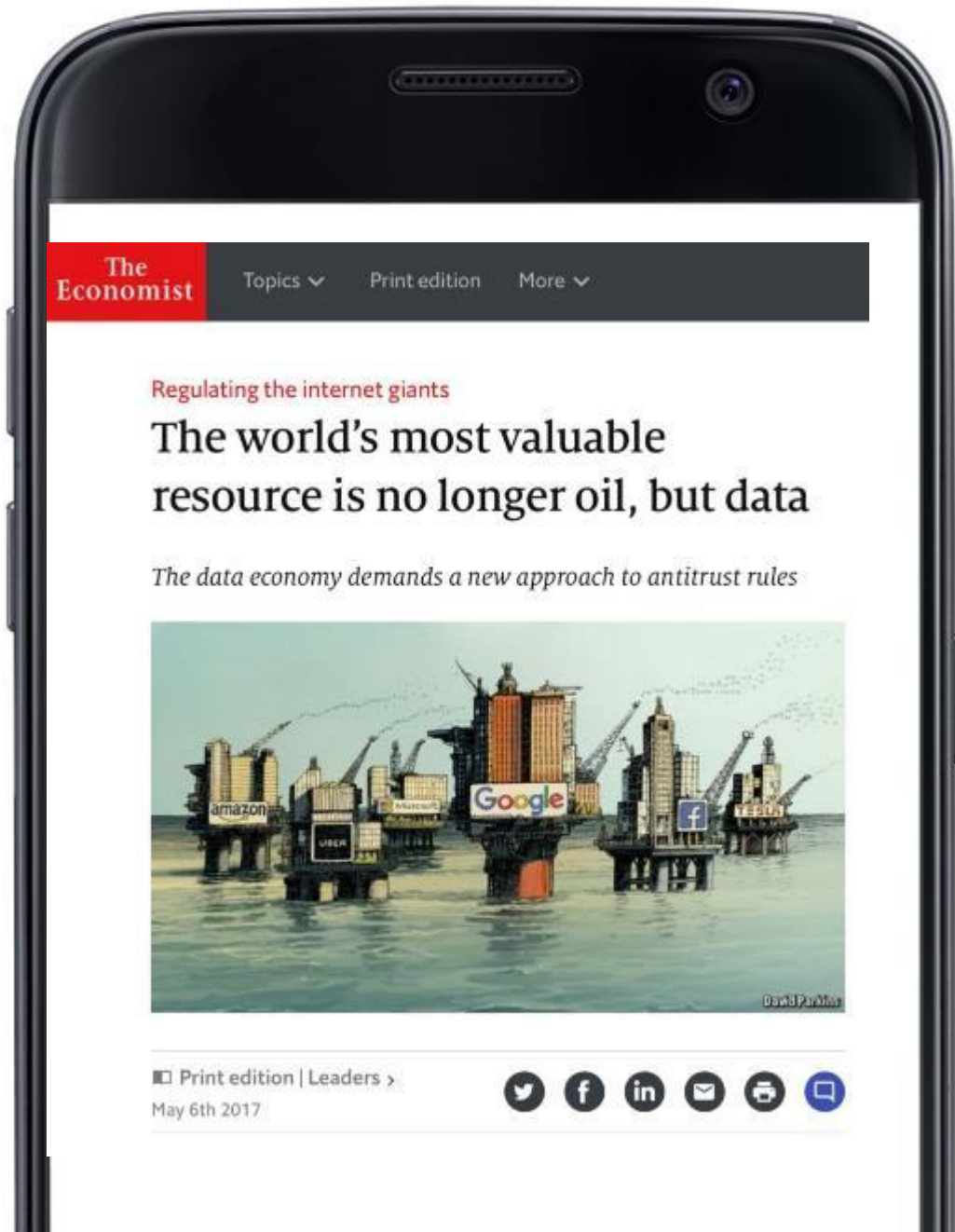
Era da tecnologia na comunicação

Jornalismo se reinventa
para acompanhar as
mudanças (data journalism)
e renova sua relevância

```
for i in people.data.us:
    response = client.api.statuses.user_timeline.get
    print 'Got', i
    if len(respons
        ltdate = r
        ltdate2 =
        today = da
        howlong =
    if howlong
        print
        totalt
    for j
        if

else:
    print

for i in people.da
response = cli
print 'Got', i
if len(respons
    ltdate = r
    ltdate2 =
    today = da
    howlong =
    if howlong
        print
        totalt
    for j
        if
```



Estamos diante de um **novo ciclo social/ comunicação:**

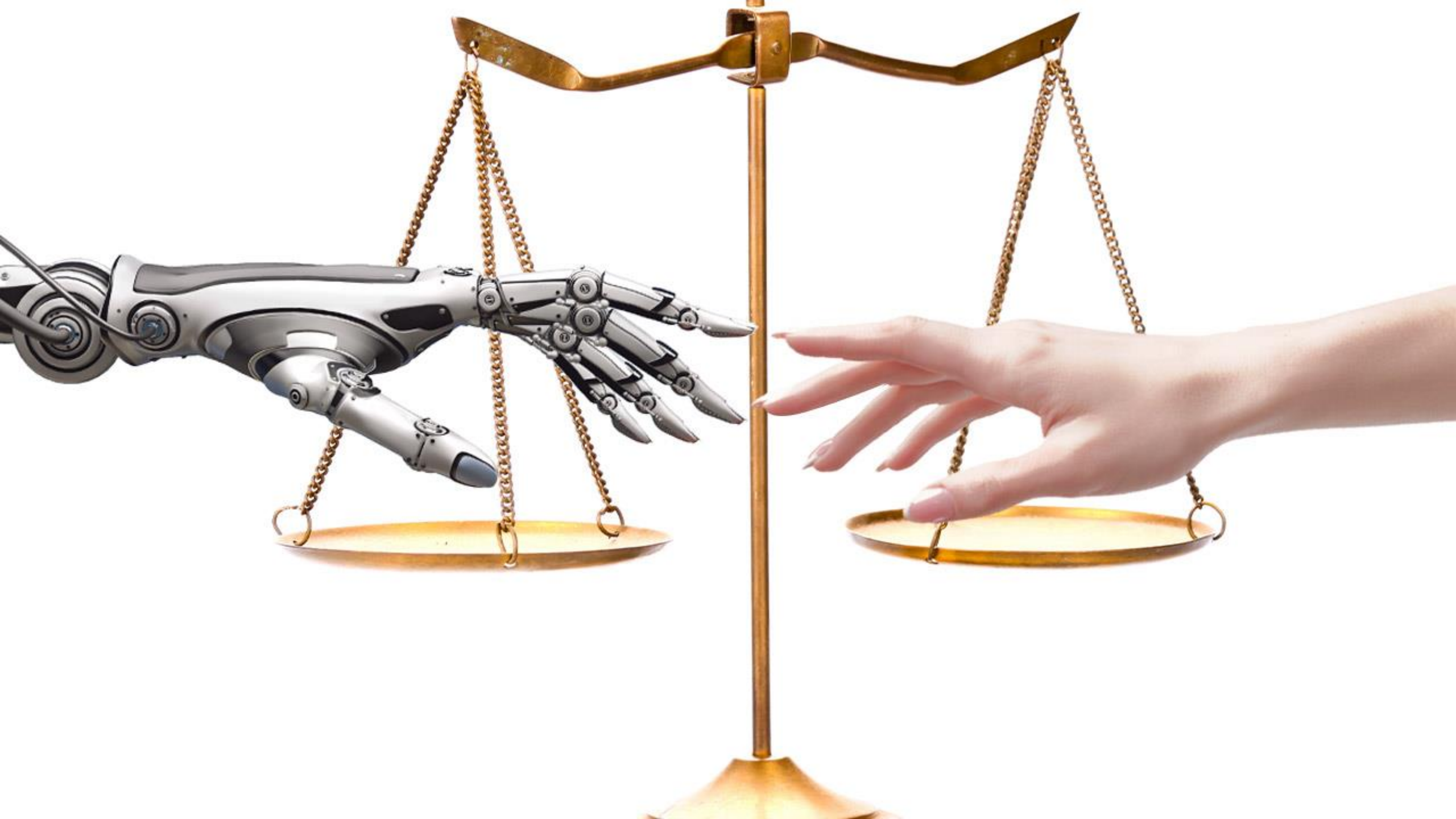


- Era da **expressão da liberdade**
Todos emitem e recebem
- Era do **coletivo**
“eu quero pertencer” é mais forte que
“eu quero ficar só”. “Uberização”
- Era da **abundância de informação**
Vasta biblioteca chamada Internet
(jornalismo de dados).
- Era da **dinamismo na comunicação**
jornalismo se reinventa para acompanhar
as mudanças tecnológicas

Os novos desafios da comunicação **e o direito**

- **Neutralidade** : base da discussão (Marco Civil);
- **Competição e não discriminação**: poder de escolha e inovação (CADE)
- **Datacracia**: ética e transparência no uso de dados (Manuais/ APL proteção de dados)
- **Sigilo da fonte** e limites do direito ao esquecimento (Constituição)





Obrigada!
cconway@uolinc.com



```
for i in people.data.users:
    response = client.api.statuses.user_timeline.get(screen_name=i.screen_name)
    print 'Got', len(response.data), 'tweets from', i.screen_name
    if len(response.data) != 0:
        ldate = response.data[0]['created_at']
        ldate2 = datetime.strptime(ldate, '%a %b %d %H:%M:%S')
        today = datetime.now()
        howlong = (today-ldate2).days
        if howlong < daywindow:
            print i.screen_name, 'has tweeted', len(response.data), 'times in the past', howlong, 'days'
            totaltweets += len(response.data)
            for j in response.data:
                if j.entities.urls:
                    for k in j.entities.urls:
                        newurl = k['expanded_url']
                        urlset.add((newurl, len(response.data)))
        else:
            print i.screen_name, 'has not tweeted in the past', howlong, 'days'

for i in people.data.users:
    response = client.api.statuses.user_timeline.get(screen_name=i.screen_name)
    print 'Got', len(response.data), 'tweets from', i.screen_name
    if len(response.data) != 0:
        ldate = response.data[0]['created_at']
        ldate2 = datetime.strptime(ldate, '%a %b %d %H:%M:%S')
        today = datetime.now()
        howlong = (today-ldate2).days
        if howlong < daywindow:
            print i.screen_name, 'has tweeted', len(response.data), 'times in the past', howlong, 'days'
            totaltweets += len(response.data)
            for j in response.data:
                if j.entities.urls:
```