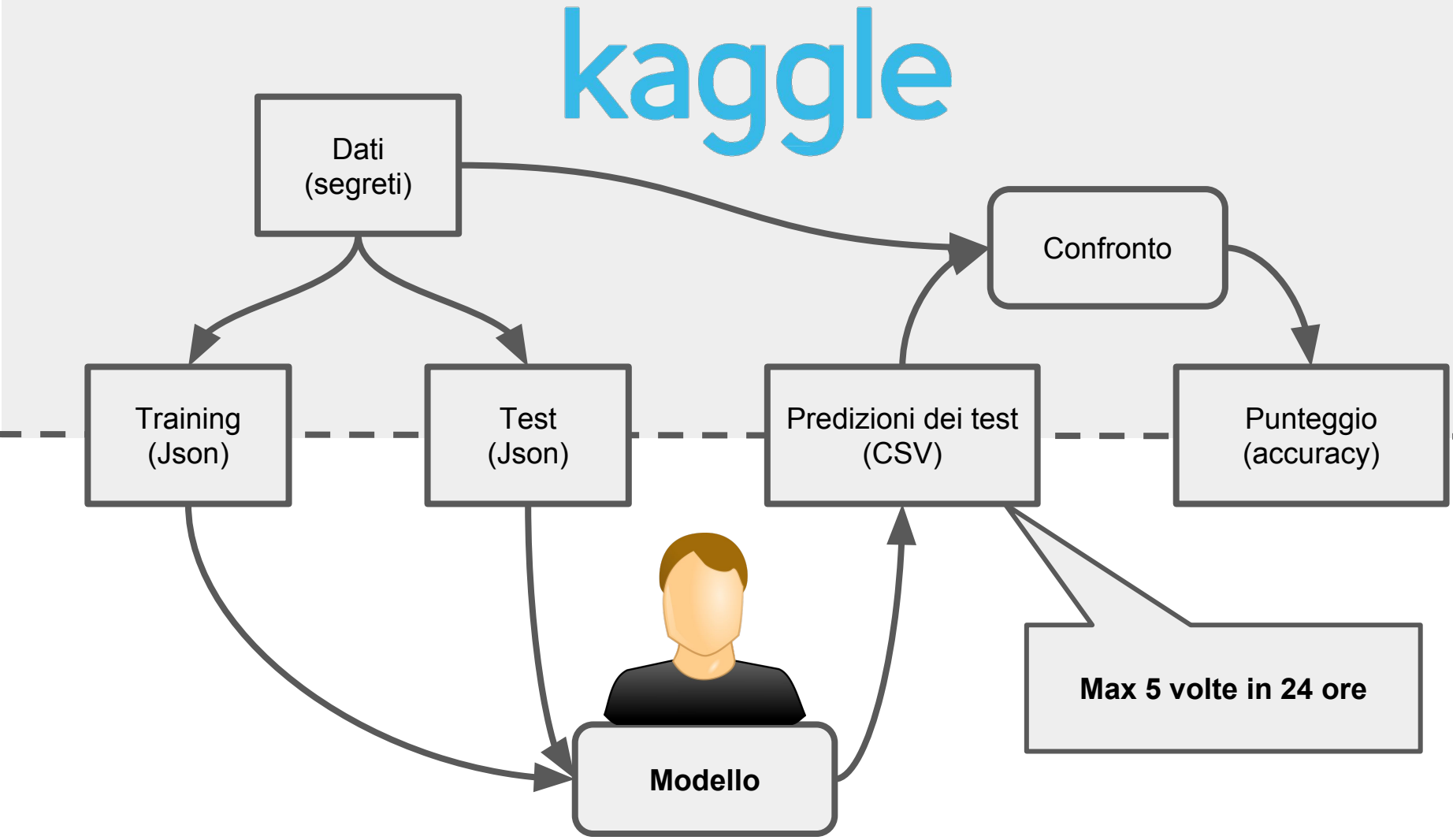
Kaggle
“What’s cooking”

Federico Poli

Contenuti

- Descrizione
  - Una competizione di Kaggle
  - I dati
  - Idee per la classificazione
- Modellazione
  - Flusso di lavoro
  - Estrazione delle features
  - Modelli utilizzati
  - Ensemble
- Classifica finale
- Riferimenti
- Varie slide extra

Una competizione di Kaggle



I Dati

1. Training


```
{
  "id": 24717,
  "cuisine": "indian",
  "ingredients": [
    "tumeric",
    "vegetable stock",
    "tomatoes",
    "garam masala",
    "red lentils",
    "red chili peppers",
    "onions",
    "spinach",
    "sweet potatoes"
  ]
},
… 40.000
```


2. Test


```
{
  "id": 44883,
  "ingredients": [
    "pasta",
    "marinara sauce",
    "dried basil",
    "chicken fingers",
    "mozzarella",
    "salt",
    "parmesan cheese"
  ]
},
… 10.000
```

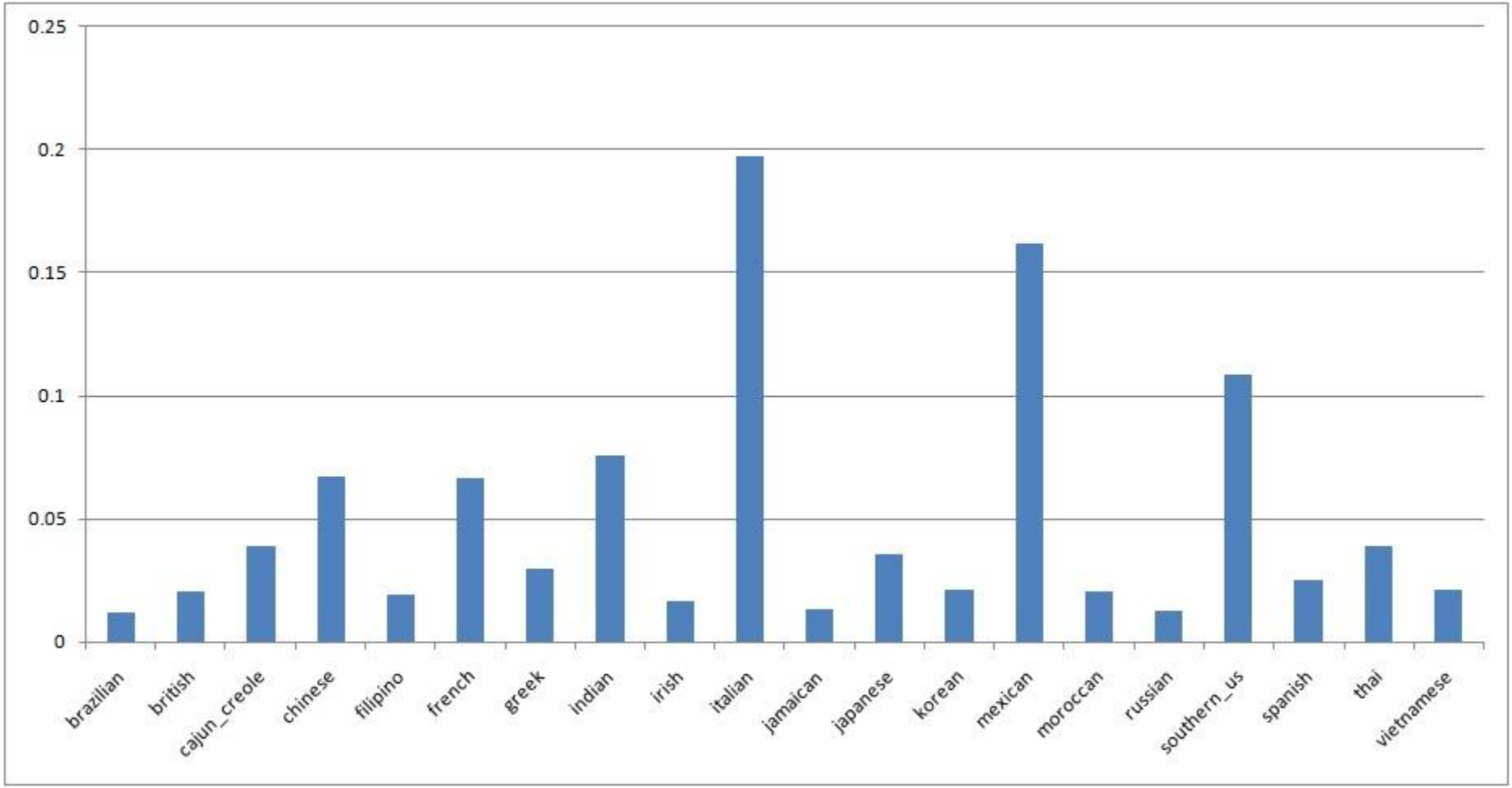

3. Previsione

44883, "italian"
…

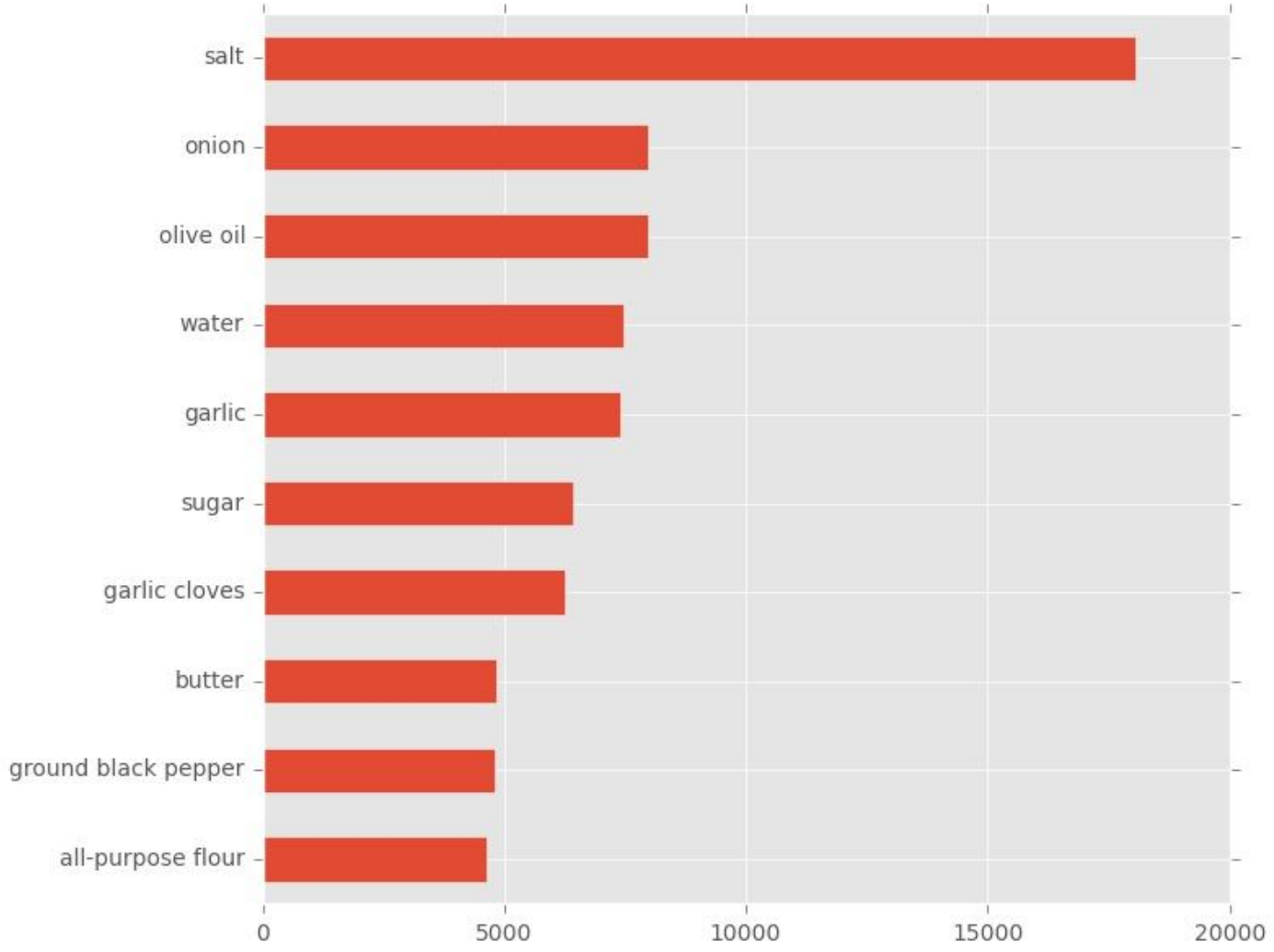
20 classi:

- brazilian
- british
- cajun_creole
- chinese
- filipino
- french
- greek
- indian
- irish
- italian
- ...

I Dati: frequenza degli esempi

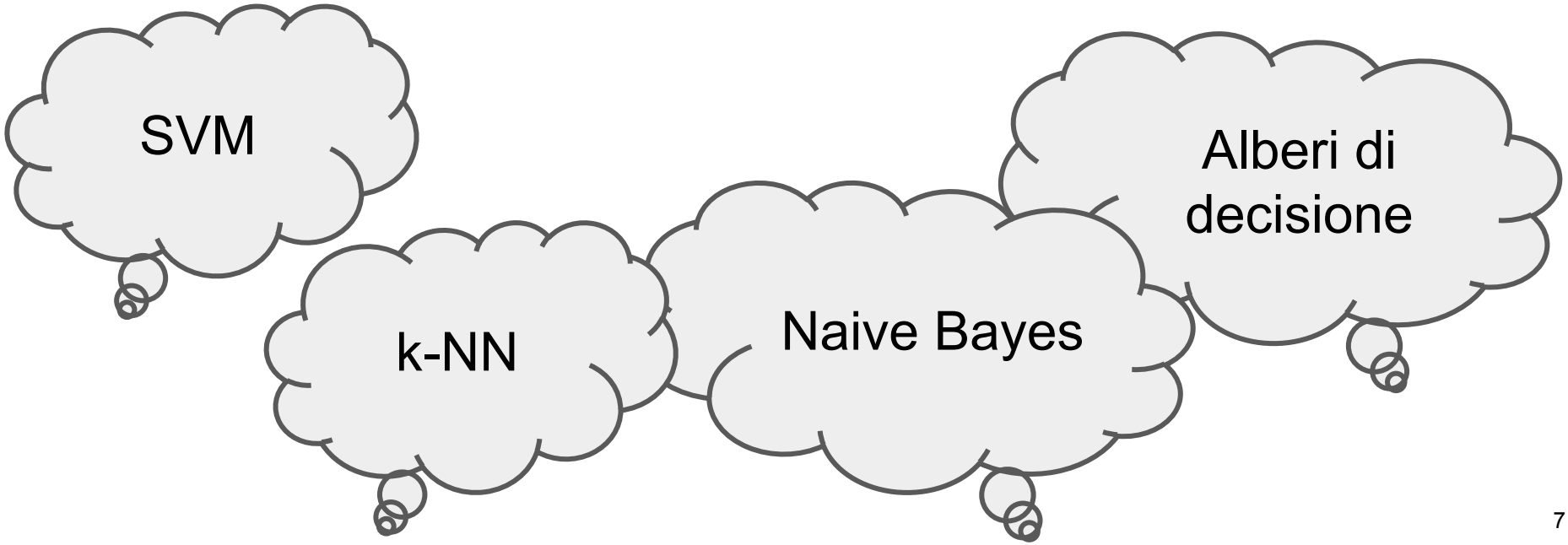


I Dati: ingredienti più usati

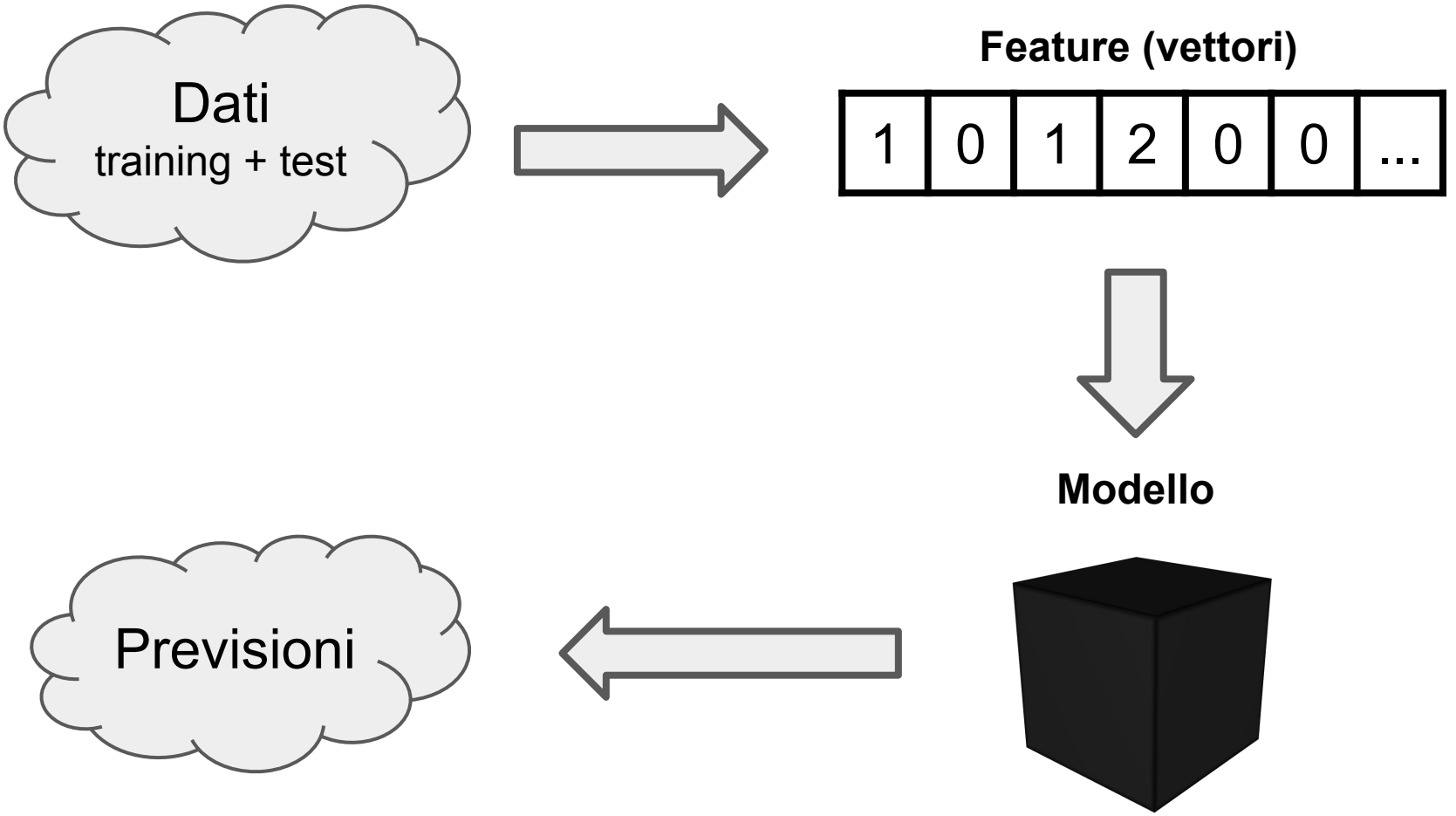


Idee per la classificazione

- Se usano mozzarella e parmigiano → italiana
- Se usano salsa di soia e miso → giapponese o cinese
- Se usano peperoncino serrano → giamaicana o messicana



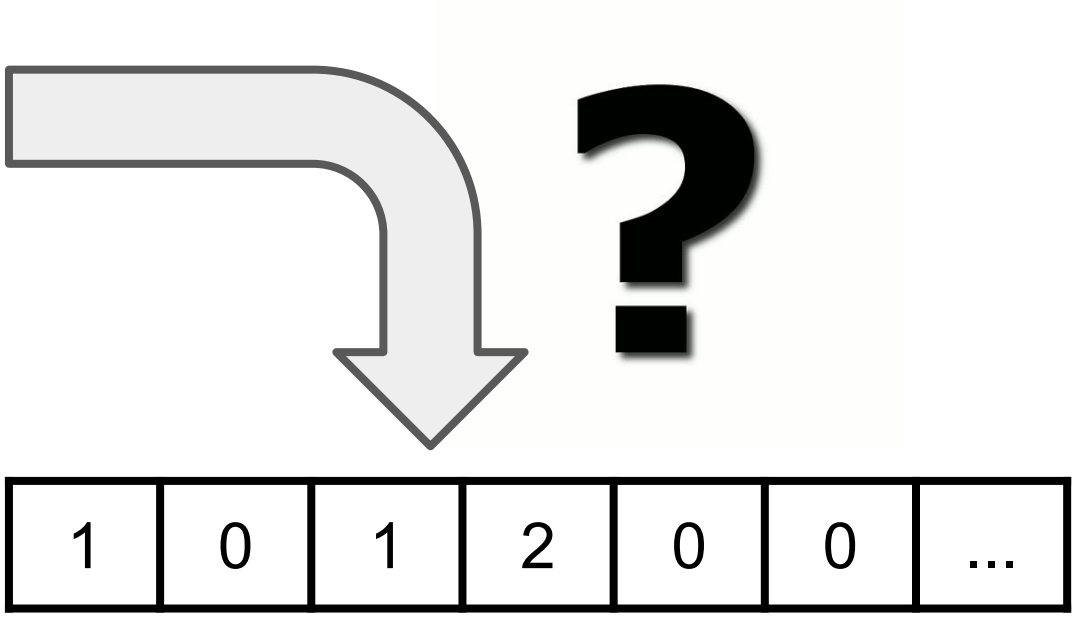
Flusso di lavoro



Estrazione delle features


```
{
 "id": 24717,
 "cuisine": "indian",
 "ingredients": [
   "tumeric",
   "vegetable stock",
   "tomatoes",
   "garam masala",
   "red lentils",
   "red chili peppers",
   "onions",
   "spinach",
   "sweet potatoes"
 ]
},
...
```

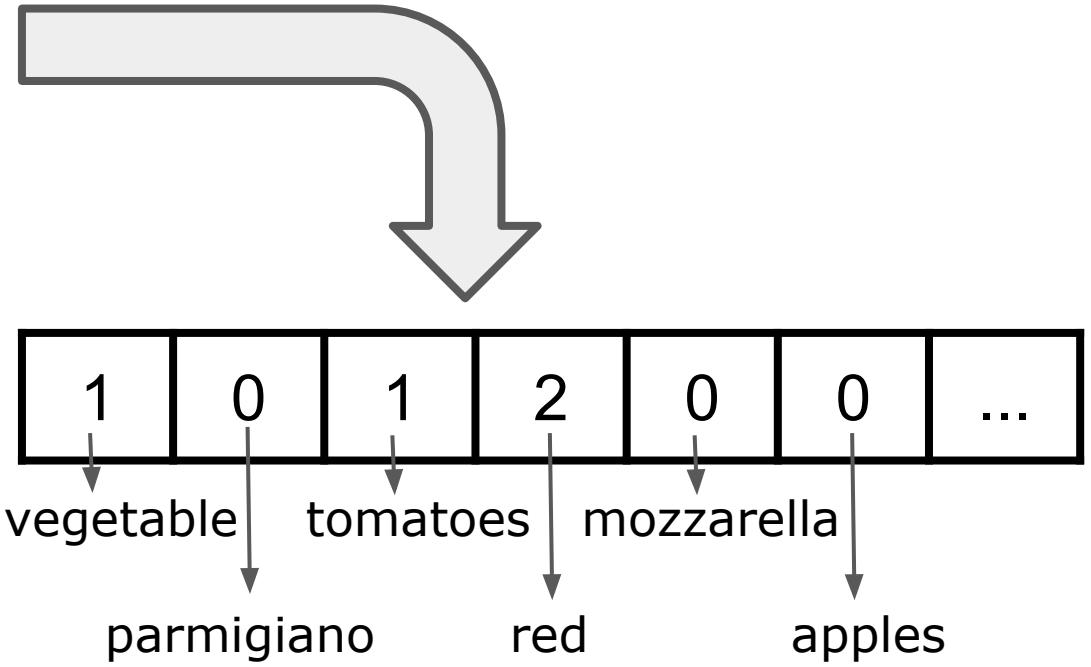

Come trasformare in un vettore?



Feature: conteggio delle parole


```
{
 "id": 24717,
 "cuisine": "indian",
 "ingredients": [
   "tumeric",
   "vegetable stock",
   "tomatoes",
   "garam masala",
   "red lentils",
   "red chili peppers",
   "onions",
   "spinach",
   "sweet potatoes"
 ]
},
...
```


Quali **parole** sono usati nella ricetta e in che numero?



Feature: normalizzazione

es: “McCormick® Pure Vanilla Extract”, "McCormick Ground White Pepper"

Normalizzazione:

- minuscolo
- trasforma i caratteri “strani”
- i caratteri a-z formano “parole”

“mcvormick, r, pure, vanilla, extract, mcvormick, ground, white, pepper”

Conteggio

2	0	1	0	0	1	...

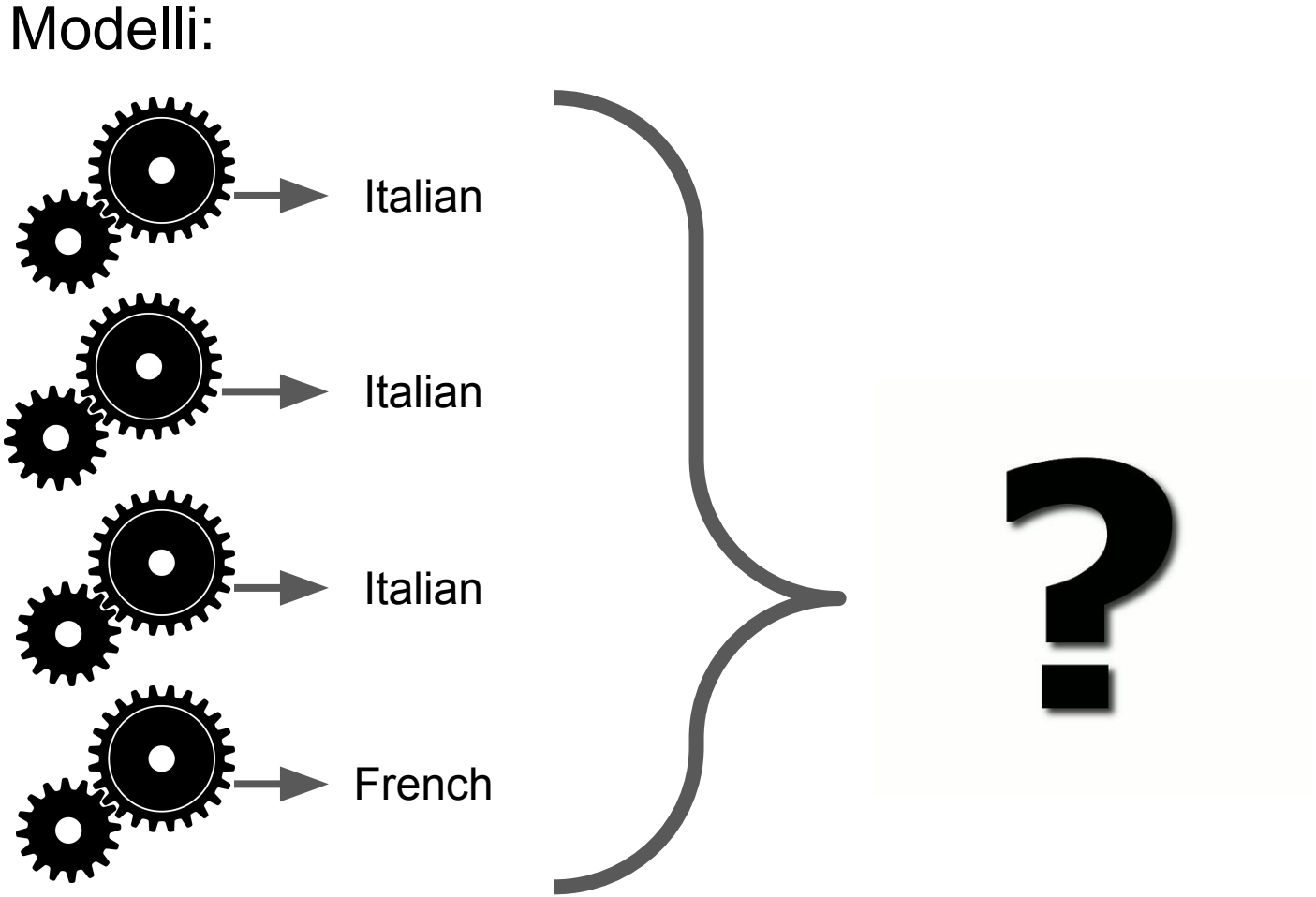
Modelli utilizzati

- SVM, kernel lineare (SVC in sklearn)
- Naive Bayes (MultinomialNB in sklearn)
- Random forest
- Logistic regression

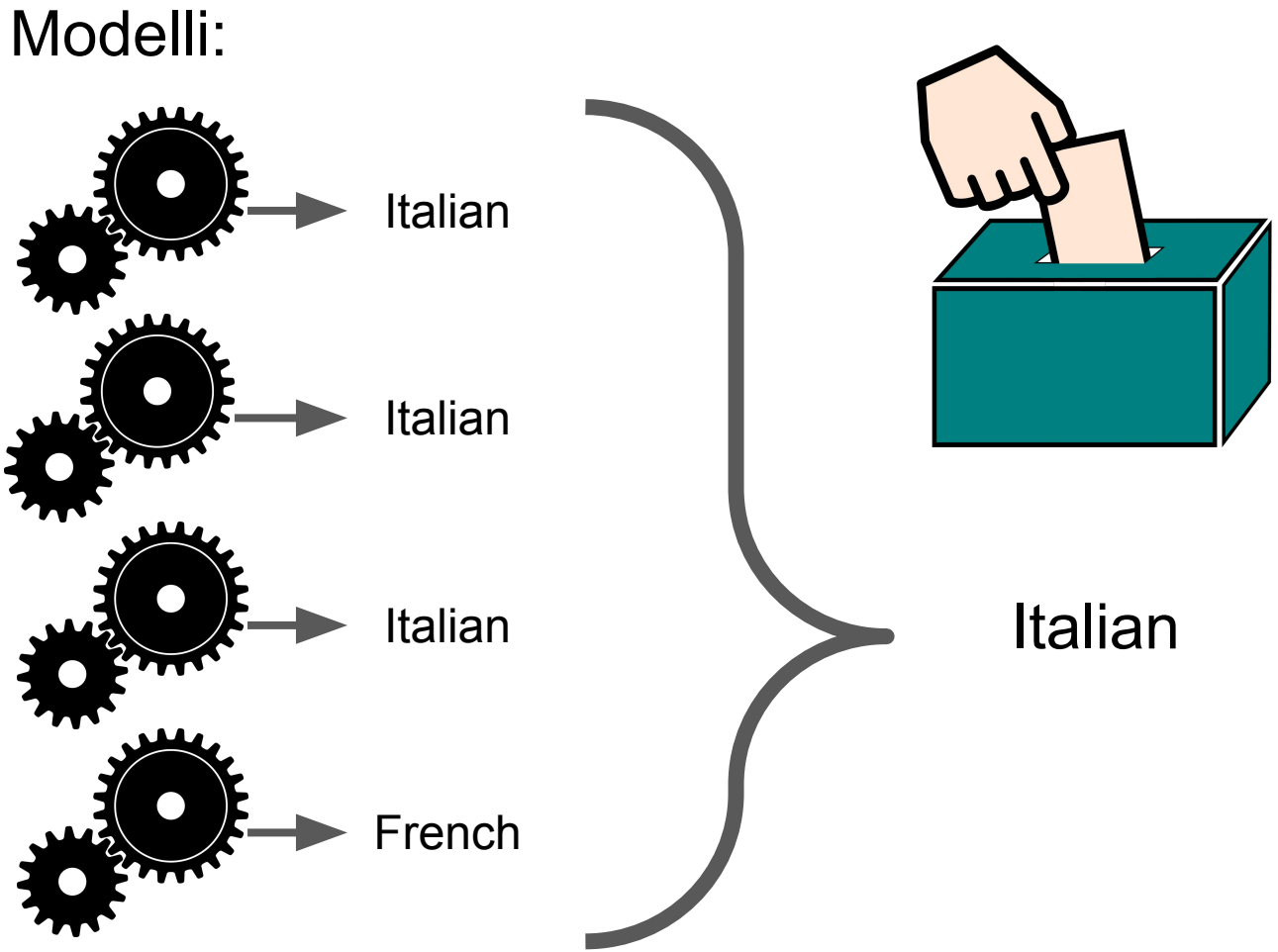
Scelta degli iperparametri con GridSearchCV, per tentativi

Modello	**Accuracy training**	**Accuracy Kaggle**
SVM lineare	0.78895	0.78992
Naive Bayes	0.73515	0.73492
Random Forest	0.76069	0.76116
Logistic regression	0.78800	0.78862

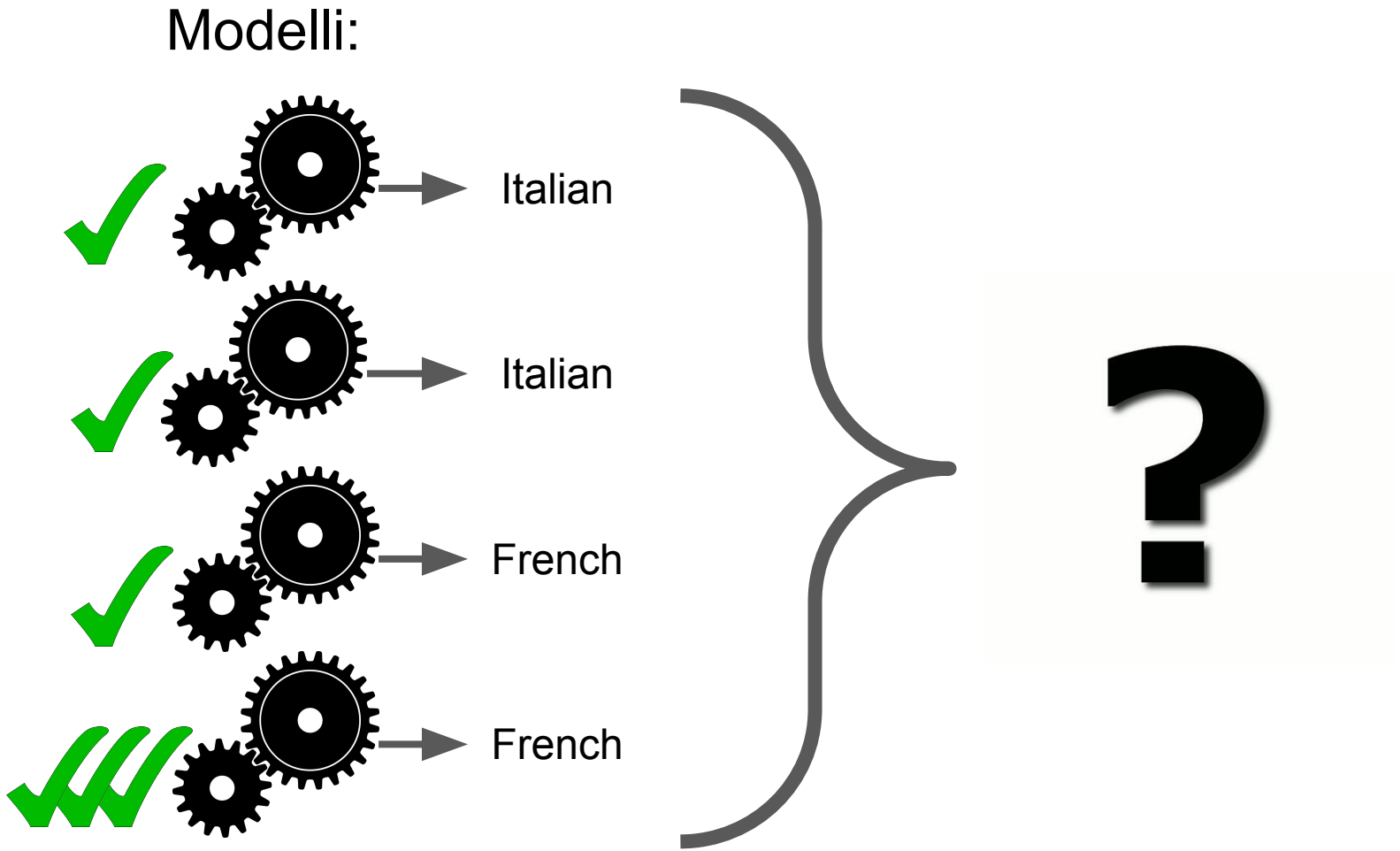
Ensemble con votazione



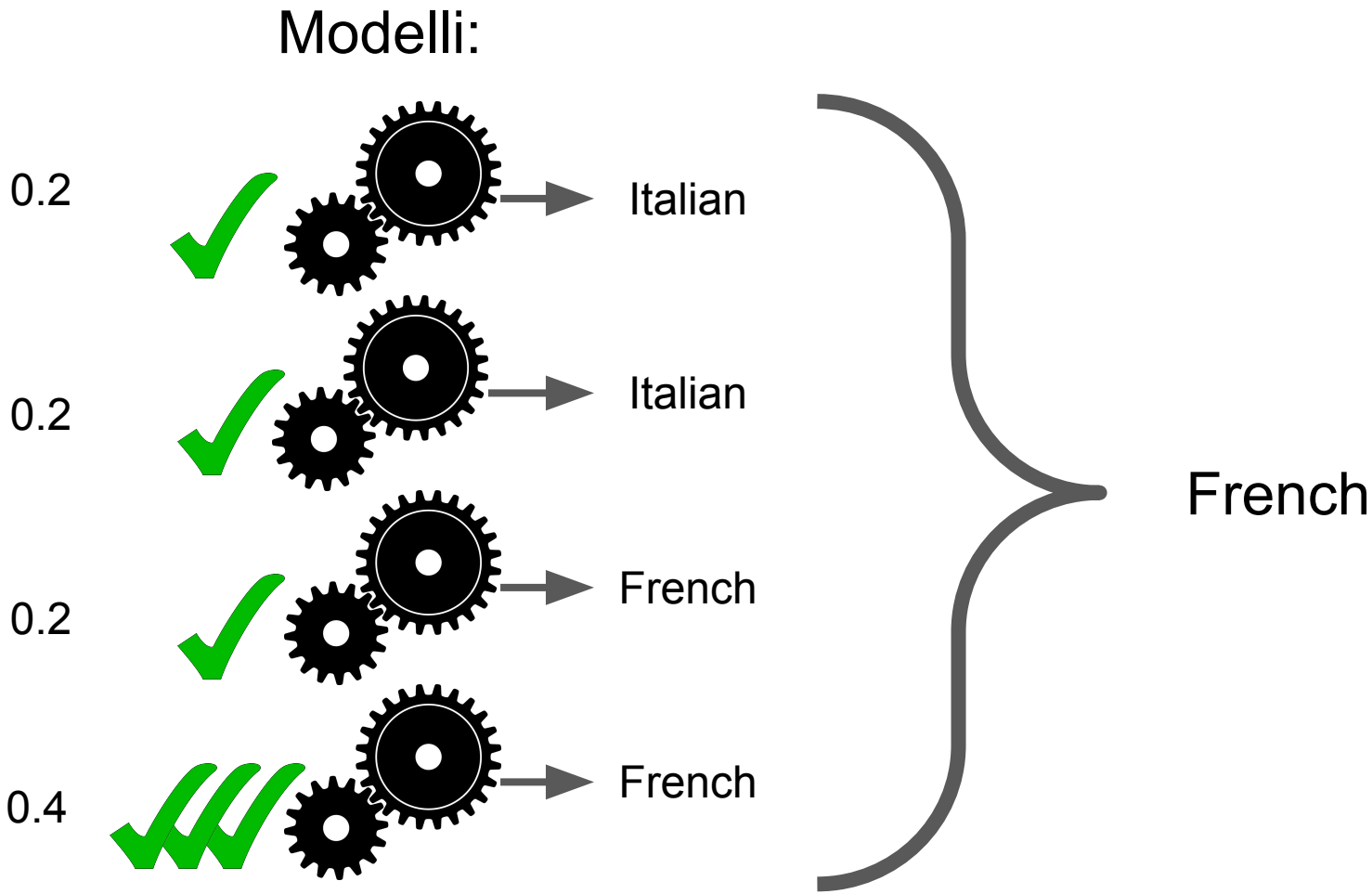
Ensemble con votazione



Ensemble con votazione pesata



Ensemble con votazione pesata



Ensemble con votazione pesata

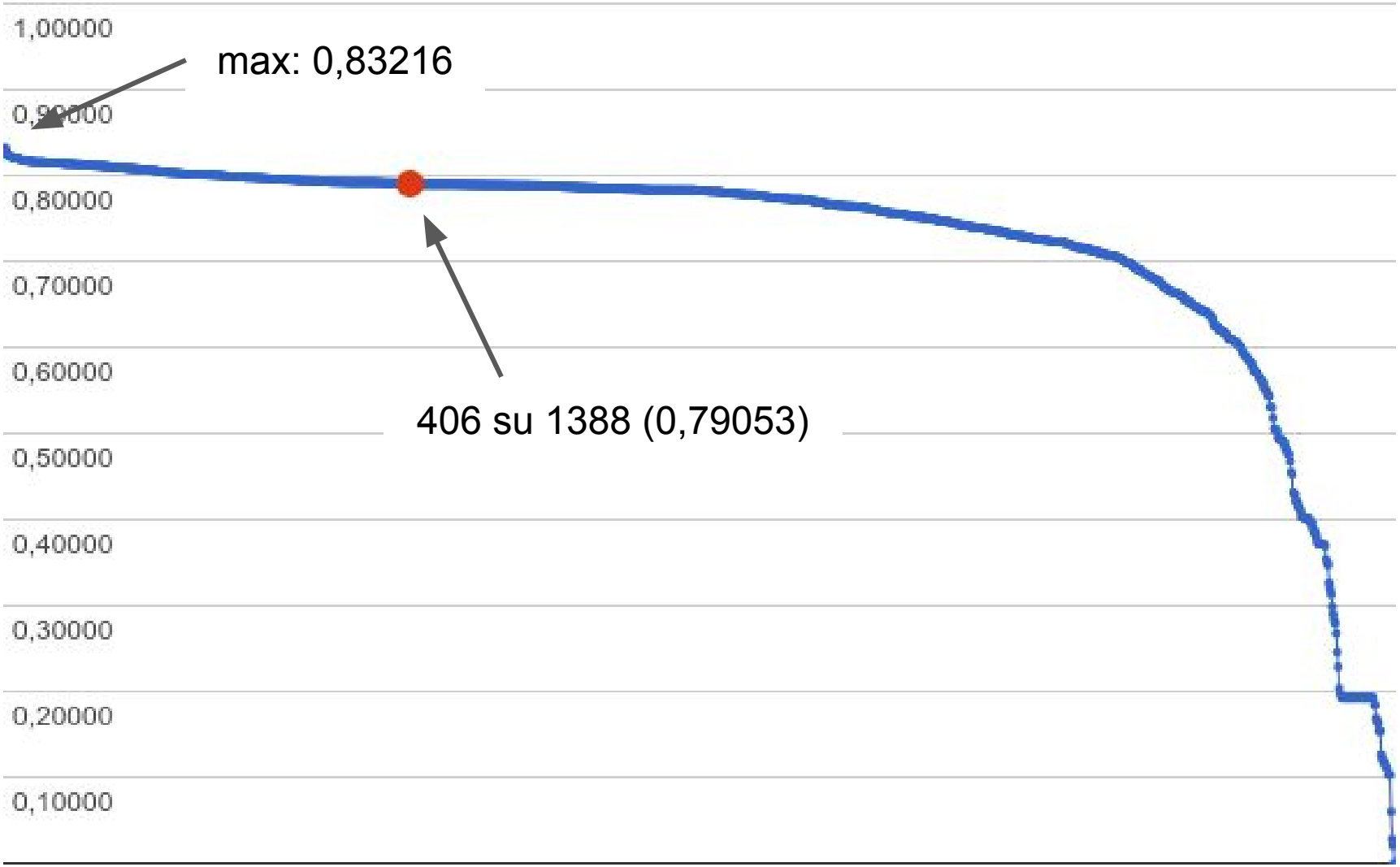
Come scegliere i pesi?

- Training sul 70% dei dati
- Predizione sul restante 30%
- Ricerca dei pesi ottimi con scipy

Modello	**Peso**
SVM lineare*	0.27
Naive Bayes	0.10
Random Forest	0.08
Logistic regression	0.53

Accuracy Kaggle:
0.78982
… vari tentativi ...
0.79053

Classifica finale



Riferimenti

- Tutto il codice è su Github: https://github.com/fpoli/kaggle-cooking

- Kaggle “What’s cooking”: https://www.kaggle.com/c/whats-cooking

kaggle Yummly™

Domande?

Segue la parte extra

Extra: trasformazione tf-idf

Alcuni ingredienti (sale, acqua, …) sono usati in tantissime cucine, bisognerebbe dargli meno importanza.

Trasformazione tf-idf (TfidfTransformer in sklearn):

- meno peso alle parole che compaiono in tante ricette

$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

TF-IDF

Term x within document y

$tf_{x,y}$ = frequency of x in y

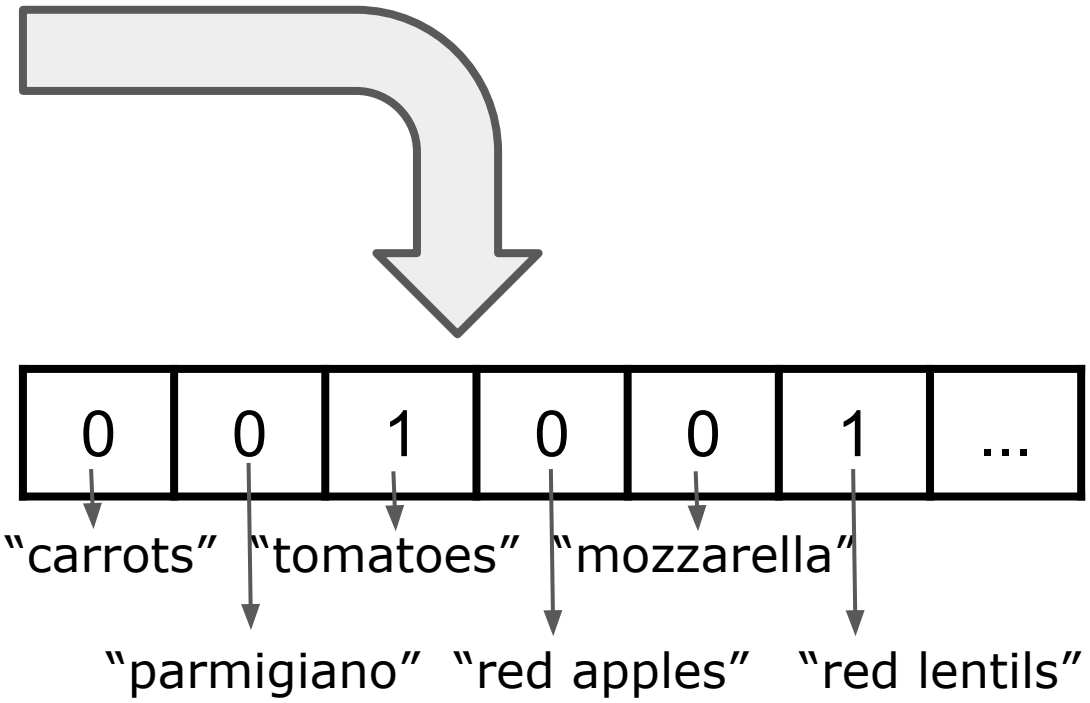
df_x = number of documents containing x

N = total number of documents

Extra: conteggio degli ingredienti


```
{
 "id": 24717,
 "cuisine": "indian",
 "ingredients": [
   "tumeric",
   "vegetable stock",
   "tomatoes",
   "garam masala",
   "red lentils",
   "red chili peppers",
   "onions",
   "spinach",
   "sweet potatoes"
 ]
},
...
```


Si poteva usare:
“quali **ingredienti** sono presenti nella ricetta?”



Extra: ingredienti vs. parole

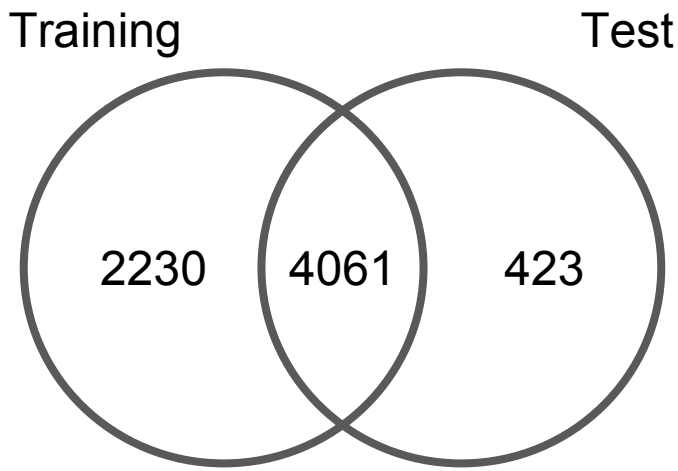
Conteggio degli ingredienti

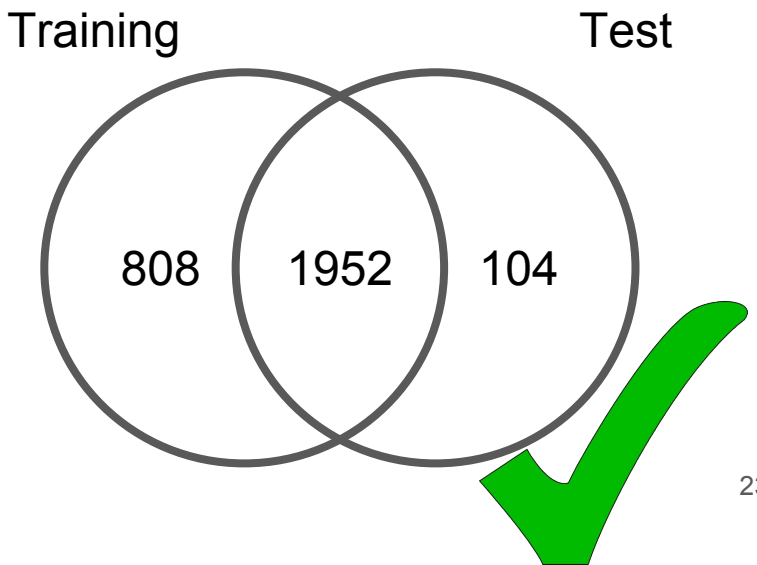
- richiede più memoria

Conteggio delle parole

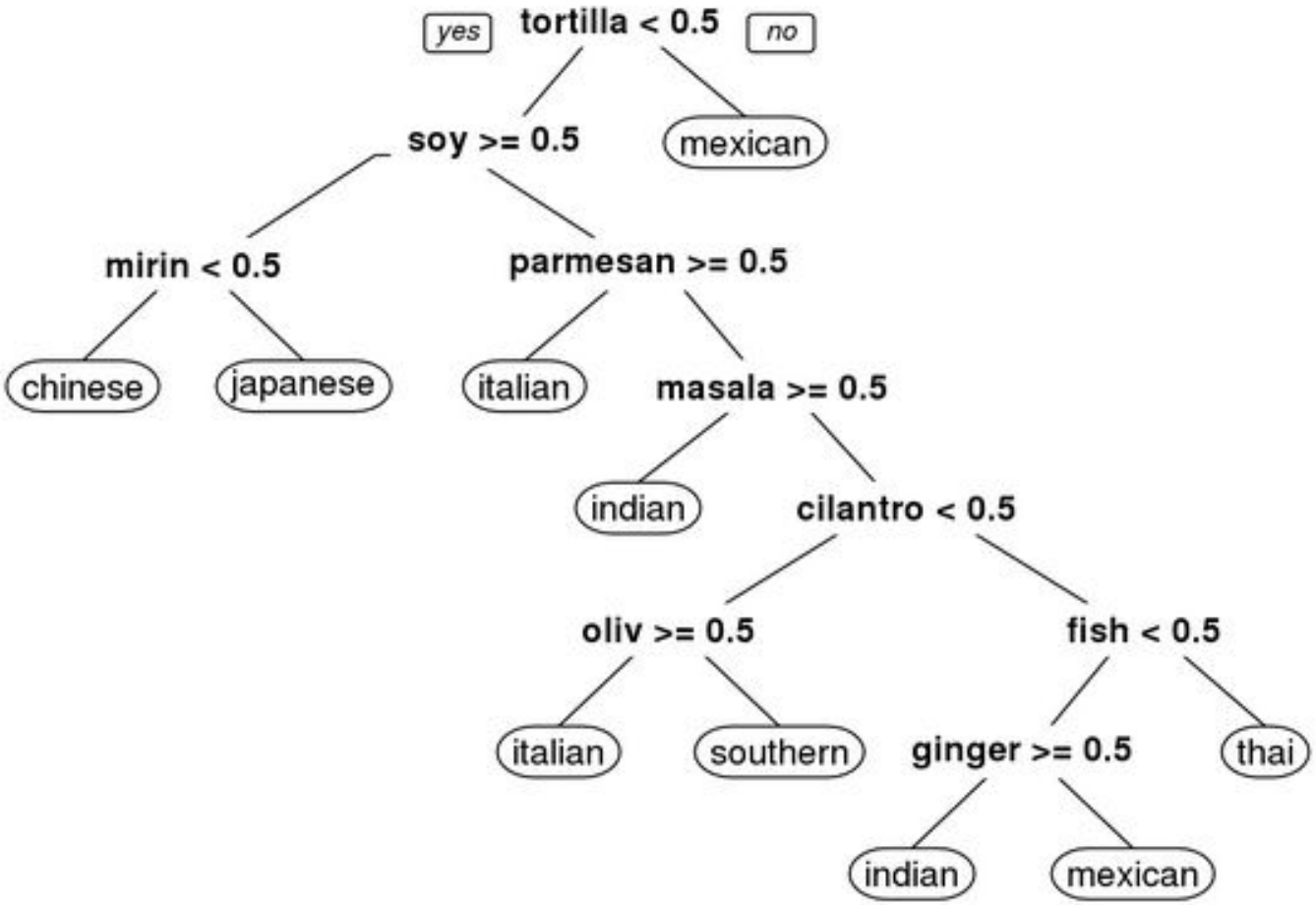
- è un po' più robusto

es: "McCormick® Pure Vanilla Extract",
"McCormick Ground White Pepper"

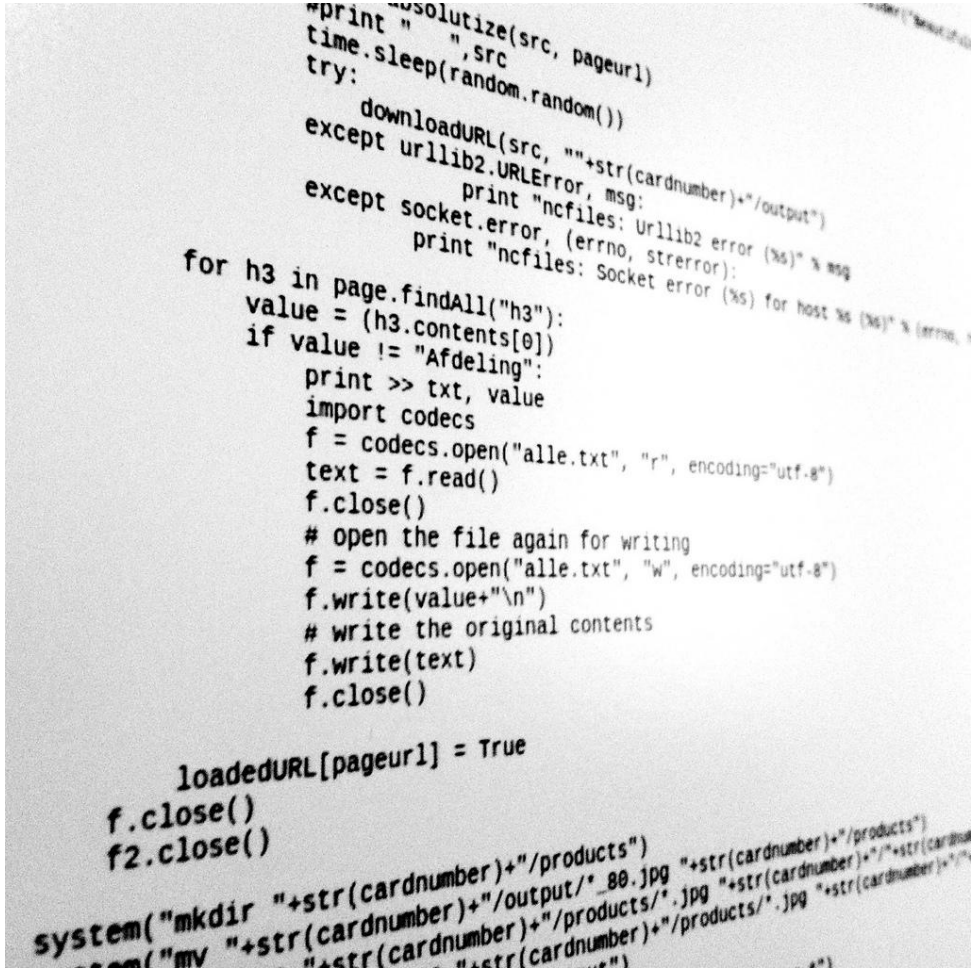




Extra: esempio di albero di decisione



Extra: volete vedere il codice?



```
                "print "    ",src
                time.sleep(random.random())
                try:
                    downloadURL(src, ""+str(cardnumber)+"/output")
                except urllib2.URLError, msg:
                        print "ncfiles: Urllib2 error (%s)" % msg
                except socket.error, (errno, strerror):
                        print "ncfiles: Socket error (%s) for host %s (%s)" % (errno,

    for h3 in page.findAll("h3"):
        value = (h3.contents[0])
        if value != "Afdeling":
            print >> txt, value
            import codecs
            f = codecs.open("alle.txt", "r", encoding="utf-8")
            text = f.read()
            f.close()
            # open the file again for writing
            f = codecs.open("alle.txt", "w", encoding="utf-8")
            f.write(value+"\n")
            # write the original contents
            f.write(text)
            f.close()

        loadedURL[pageurl] = True
    f.close()
    f2.close()

system("mkdir "+str(cardnumber)+"/products")
system("mv "+str(cardnumber)+"/output/*_80.jpg "+str(cardnumber)+"/products")
system("mv "+str(cardnumber)+"/products/*.jpg "+str(cardnumber)+"/"+str(cardnu
system("mv "+str(cardnumber)+"/products/*.jpg "+str(cardnumber)+"/
```