



a HOPEFULLY-SMART NEWS AGGREGATOR



Data Mining Project
Filippo Scotto

INTRODUCTION

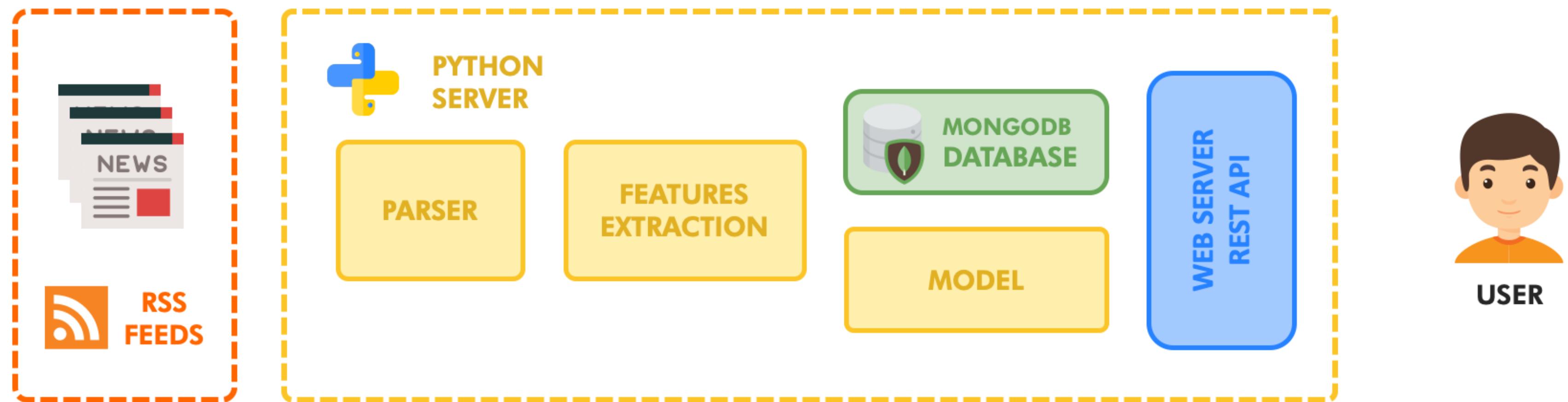
There are thousands of news sites, but most of their articles are *not for everybody*. Can we build a *machine learning based* system capable to **filter out** what we are not interesting in?



We need a system capable to classify the **category** of the articles and correctly predict their **likability**.

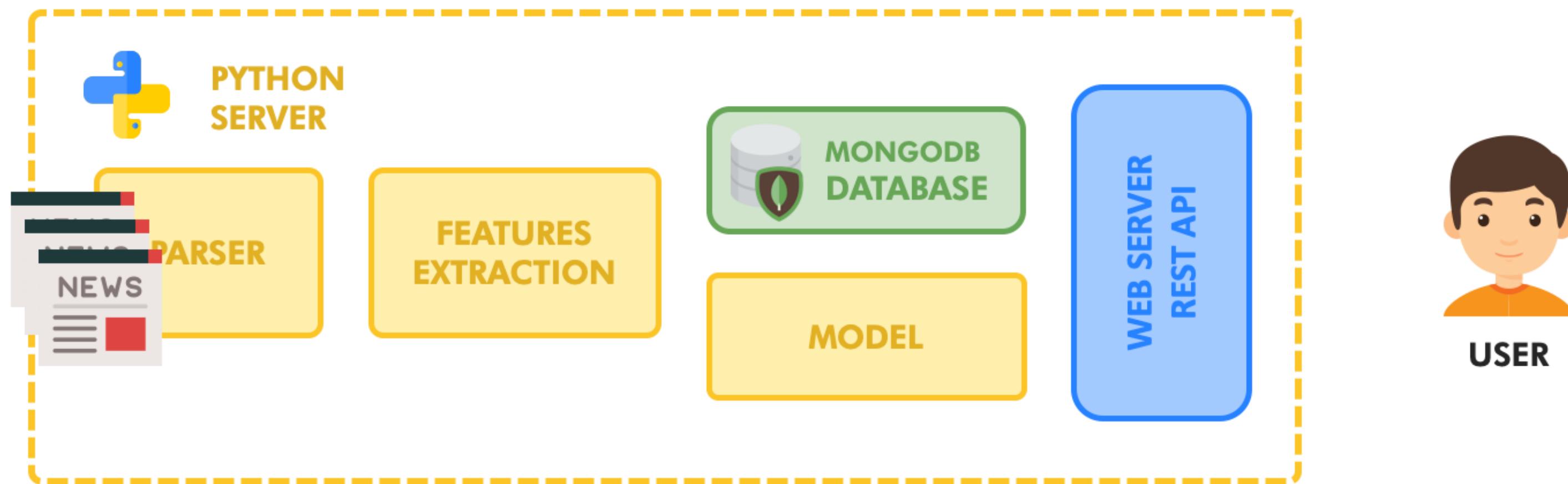
- The designed system will parse the **RSS feeds** coming from the most popular italian news sites;
- It will **preprocess** the data and **extract some features**;
- It will **classify** the news and predict wheter the user may like it or not.

THE SYSTEM



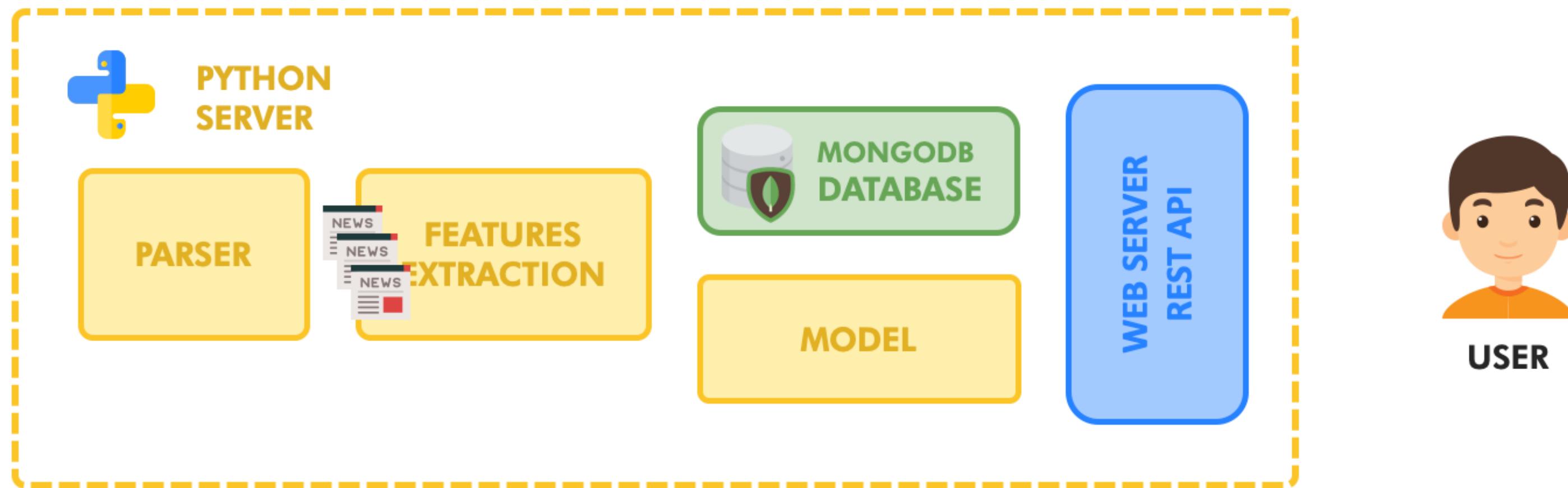
1 The articles are downloaded from the RSS Feeds

THE SYSTEM



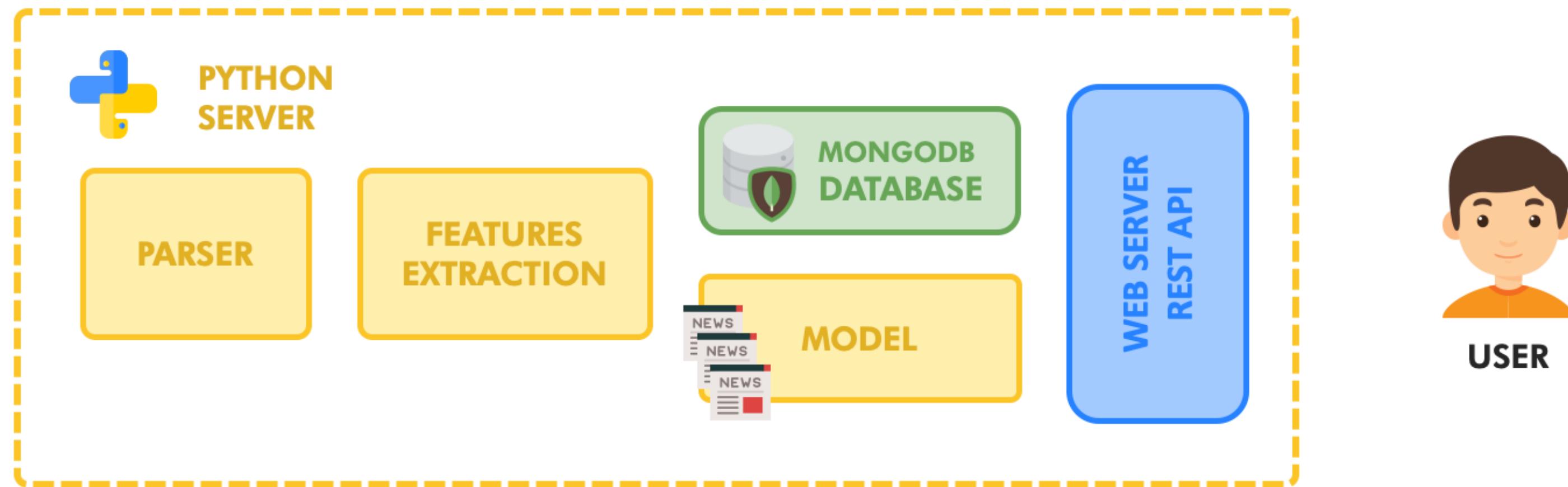
2 The articles are parsed by a Python Script

THE SYSTEM



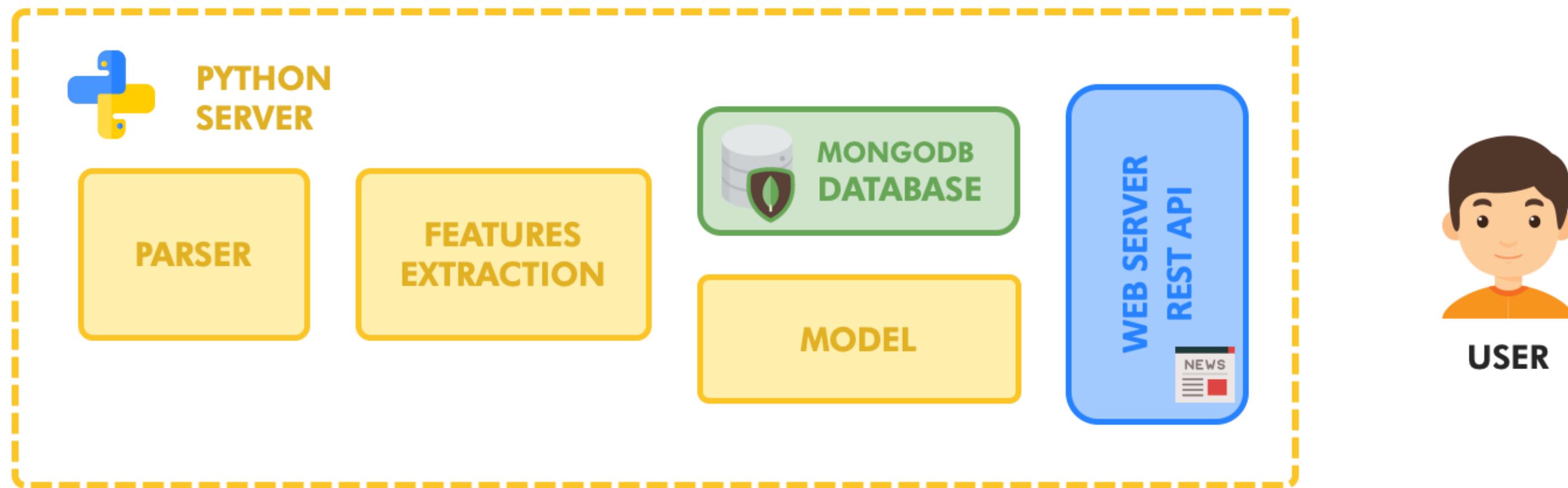
3 The parsed articles are processed to extract some features

THE SYSTEM



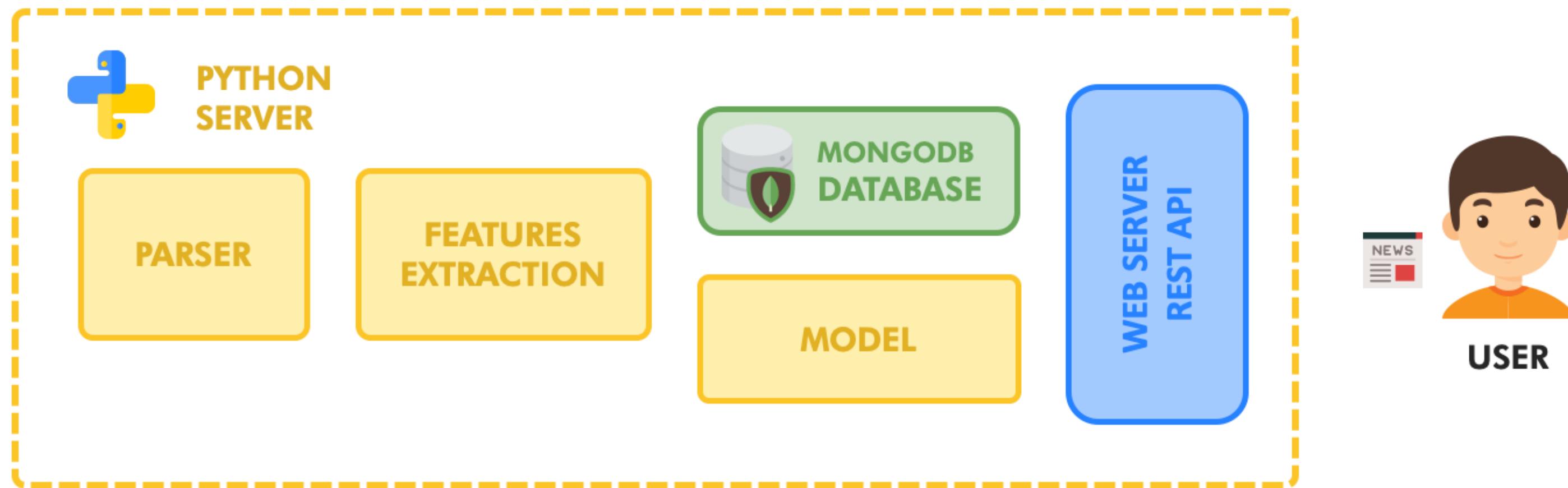
4 The features are given as input to a Machine Learning Model

THE SYSTEM



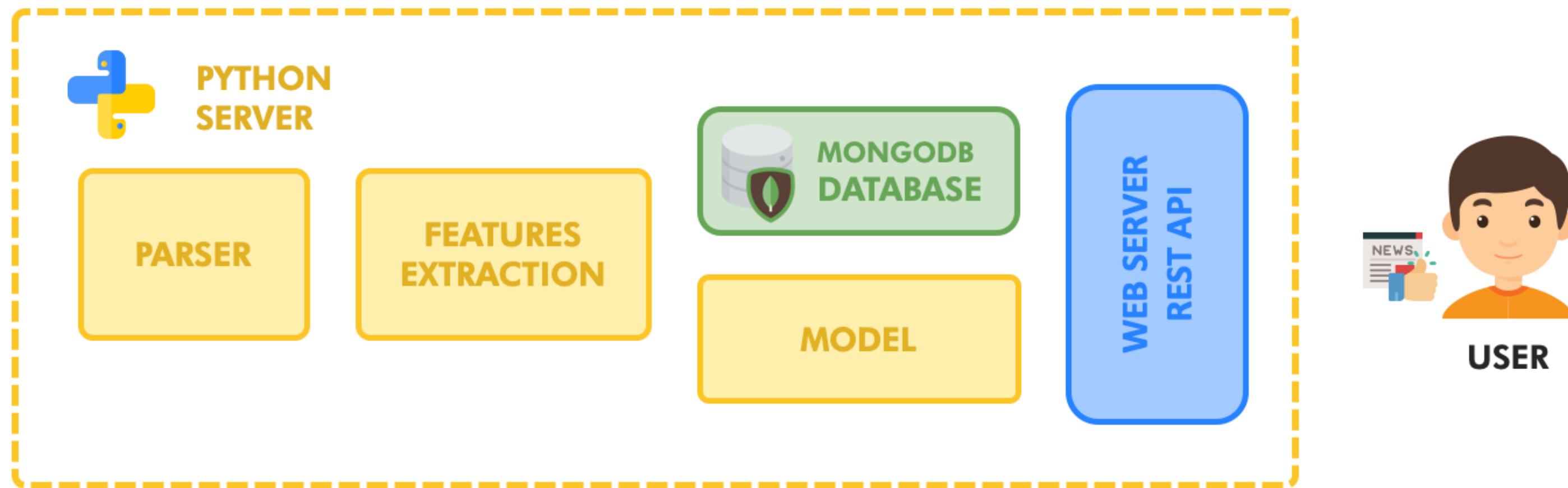
5 The filtered news are passed to the webserver

THE SYSTEM



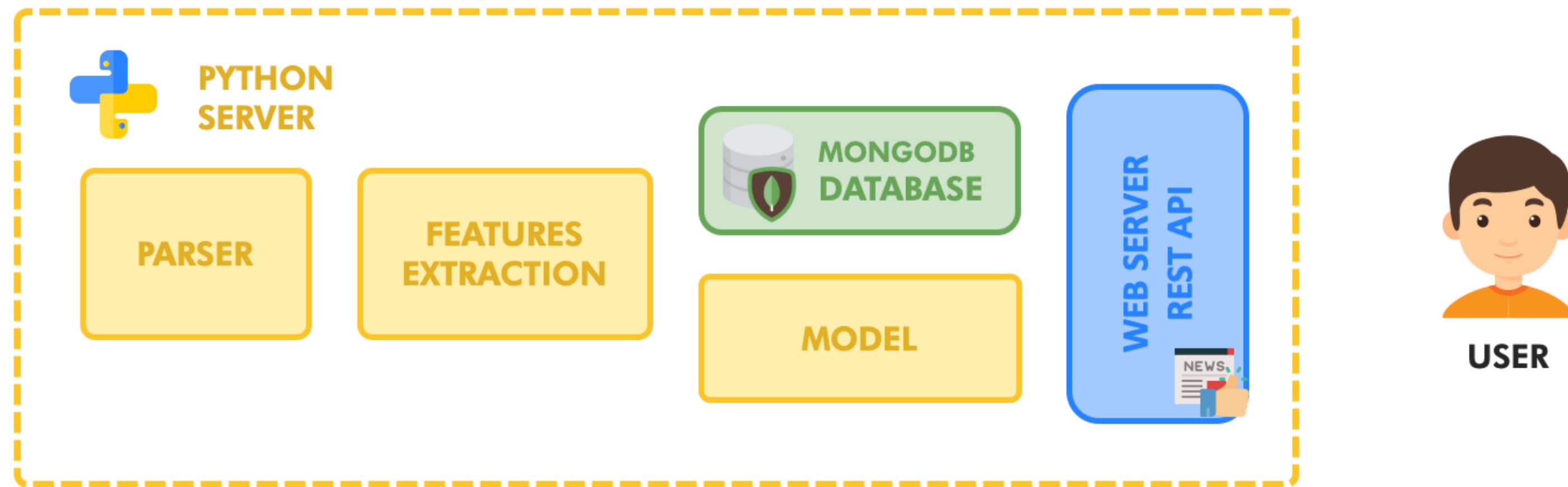
6 The user can request its personalized feed using a REST API

THE SYSTEM



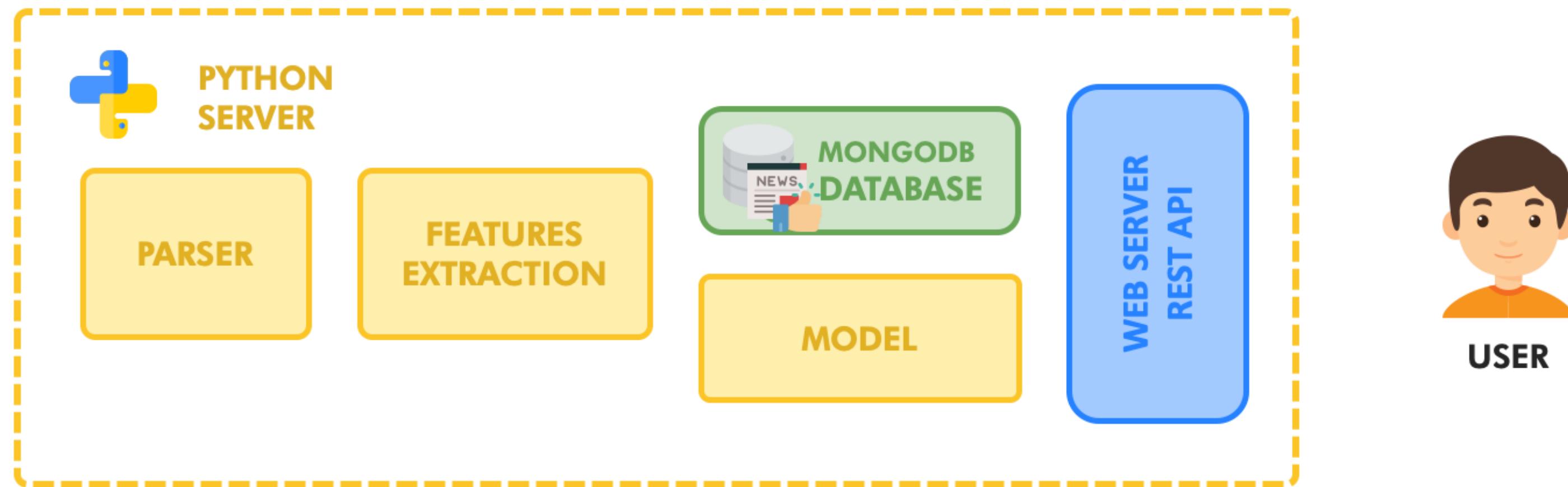
7 The user can “like” the article using the WebApp

THE SYSTEM



8 The like request is processed by the WebServer

THE SYSTEM



9 Finally the database entry for the liked article is updated

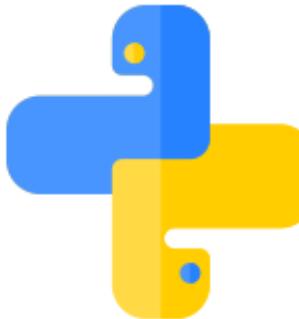
THE PARSER MODULE

News are collected from the RSS feeds from the most popular italian news sites. Unfortunately RSS feeds are not as structured as they were meant to be...

- Missing values and different tag names
- Article descriptions containing images or raw HTML
- We don't need everything!



NEWS FEED



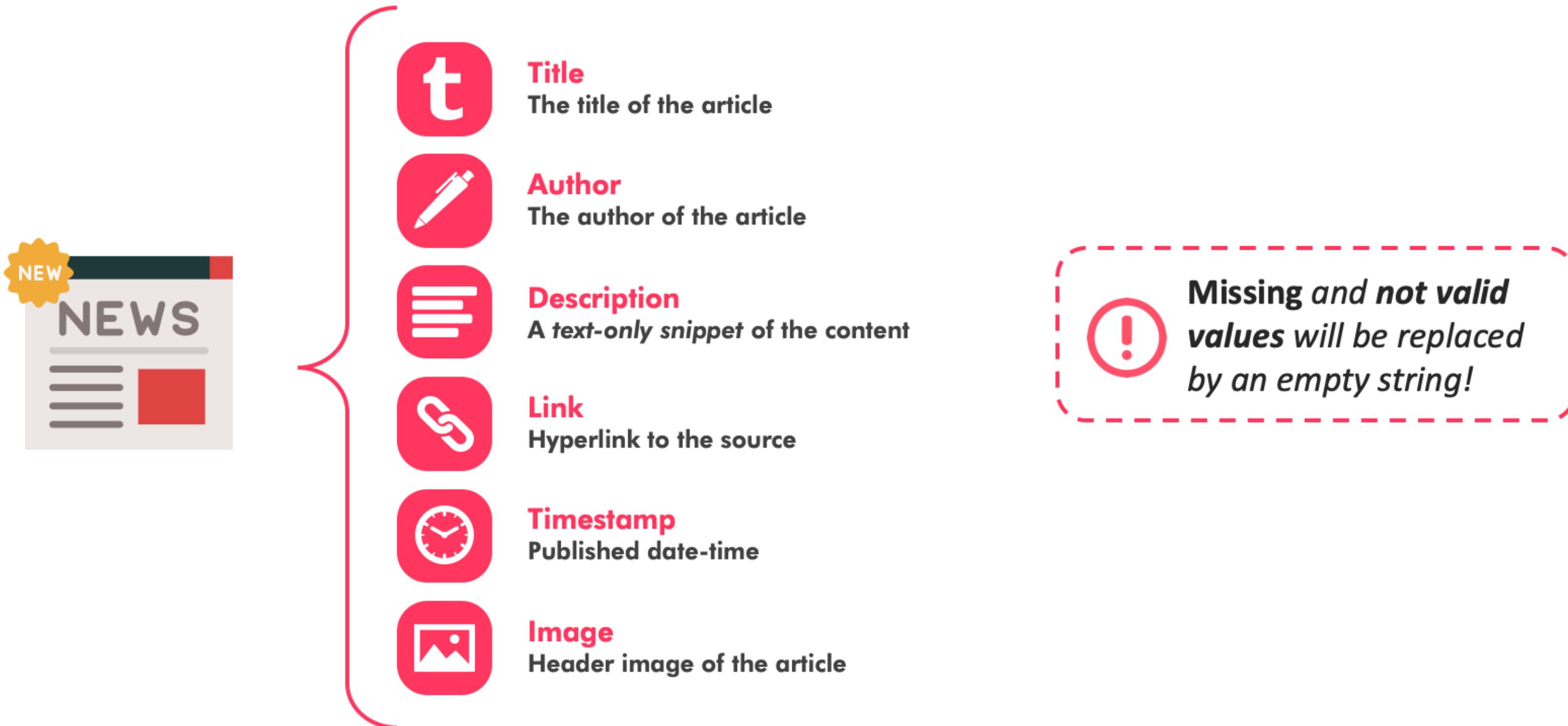
PYTHON SCRIPT



ADJUSTED FEED

THE PARSER MODULE: AN EXAMPLE

In the end what we obtain as output from the parser is a list of articles with the following structure:



BUILDING THE DATASET

Once the feeds are parsed, we have to build a dataset for our model. In particular what we need to do is:

- 1. Manually tag the articles:** each will be labelled with a category (nine different categories available)
- 2. Manually Like/Dislike the articles**

To do this a **WebApp** was built: the app will show all the articles coming from the feed and then:

1. The article must be liked/disliked



Like/Dislike buttons

2. The liked/disliked article will be inserted in the database and can be tagged



Form to tag an article

This article is about...

Politica

skip

tag

Button to skip current article and tag the next one

BUILDING THE DATASET: THE WEBAPP

Learning mode

hopefully-smart news aggregator

> learn

learning mode

Whole parsed feed

Gazzetta dello Sport
Lecce promosso in Serie A, esplode la festa per le vie della città
© 2019-05-12T09:03:13
Lecce promosso in Serie A, esplode la festa per le vie della città Dopo la vittoria sullo Spezia e il secondo posto che vale la promozione tutta la gioia dei salentini si è riversata per le strade
Corriere della Sera
Riforme a metà, rissa continua M5S-Lega
Enrico Marro e Dino Martirano
© 2019-05-12T09:02:17
A quasi un anno dalla nascita dell'esecutivo giallorosso reddito di cittadinanza e quota 100 alla prova dei risultati. Scontro sulle infrastrutture. Il nodo della prossima manovra
Corriere della Sera
Gp di Spagna: tre grandi rimonte del passato che fanno sperare ancora la Ferrari
Daniele Sparisci, inviato Barcellona
© 2019-05-12T09:02:10
Dopo l'ennesima prima fila dominata dalla Mercedes, alla Rossa non resta che confidare nella rivalità interna tra Hamilton e Bottas. L'inglese infatti se vuole vincere dovrà passare il compagno di squadra nelle prime curve
Corriere della Sera
Palazzo Chigi teme la crisi, i sospetti di Conte e M5S: il decreto-sicurezza è una mina
Antonio Gattulli
© 2019-05-12T09:01:31
L'idea che il piano sicurezza di Salvini serva a causare la crisi. E si spera nello stop del Colle
Corriere della Sera
F2 Montmelò, Latifi mette la terza. Shwartzman ok in F3
Antonio Gattulli
© 2019-05-12T09:01:31
F2 Montmelò, Latifi mette la terza. Shwartzman ok in F3 Il canadese centra il terzo successo in cinque gare. Mick Schumacher forza e chiude dietro. In Formula 3 il russo vince grazie alla penalizzazione di Lundgaard
Corriere della Sera
Salvini e Di Maio: 15 appuntamenti (uno è saltato) nel weekend
Redazione Politica
© 2019-05-12T09:01:22
I due vicepremier, divisi su molti fronti in una delle settimane più tese per la tenuta del governo,

Tagging mode

hopefully-smart news aggregator

> tag

tag an article

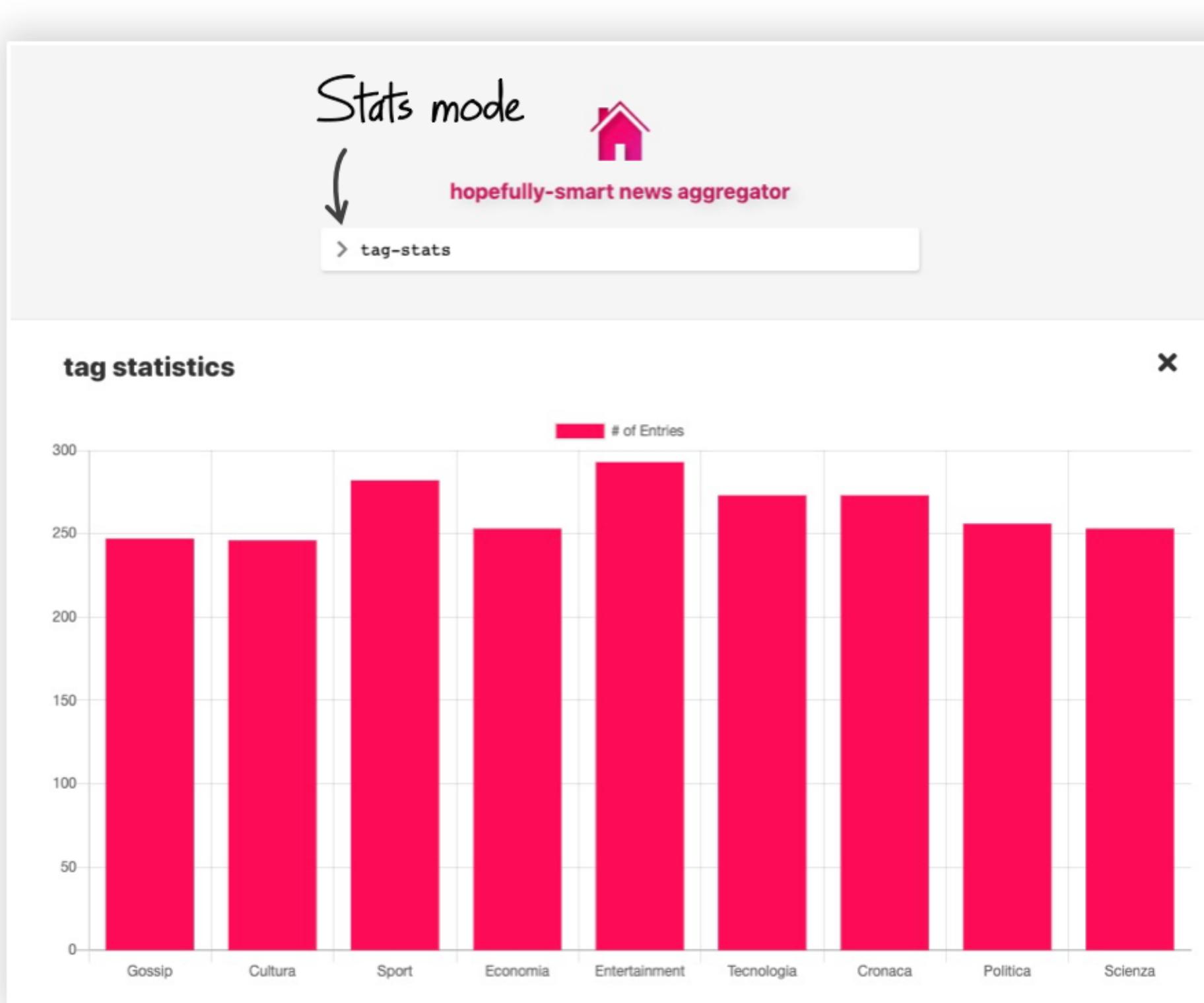
This article is about...

Pollica

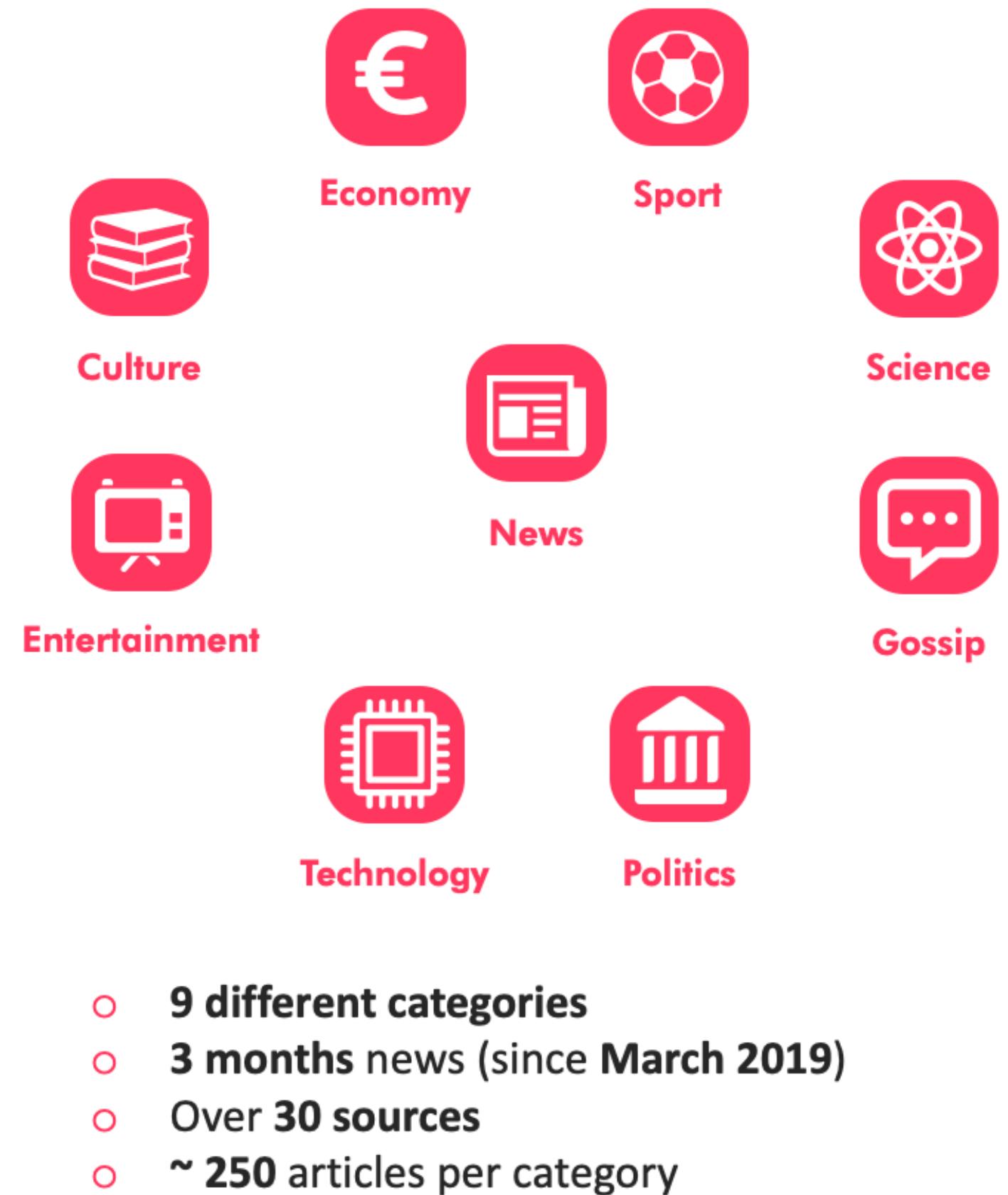
skip tag

Corriere della Sera
Palazzo Chigi teme la crisi, i sospetti di Conte e M5S: il decreto-sicurezza è una mina
© 2019-05-12 09:01:54
L'idea che il piano sicurezza di Salvini serva a causare la crisi. E si spera nello stop del Colle

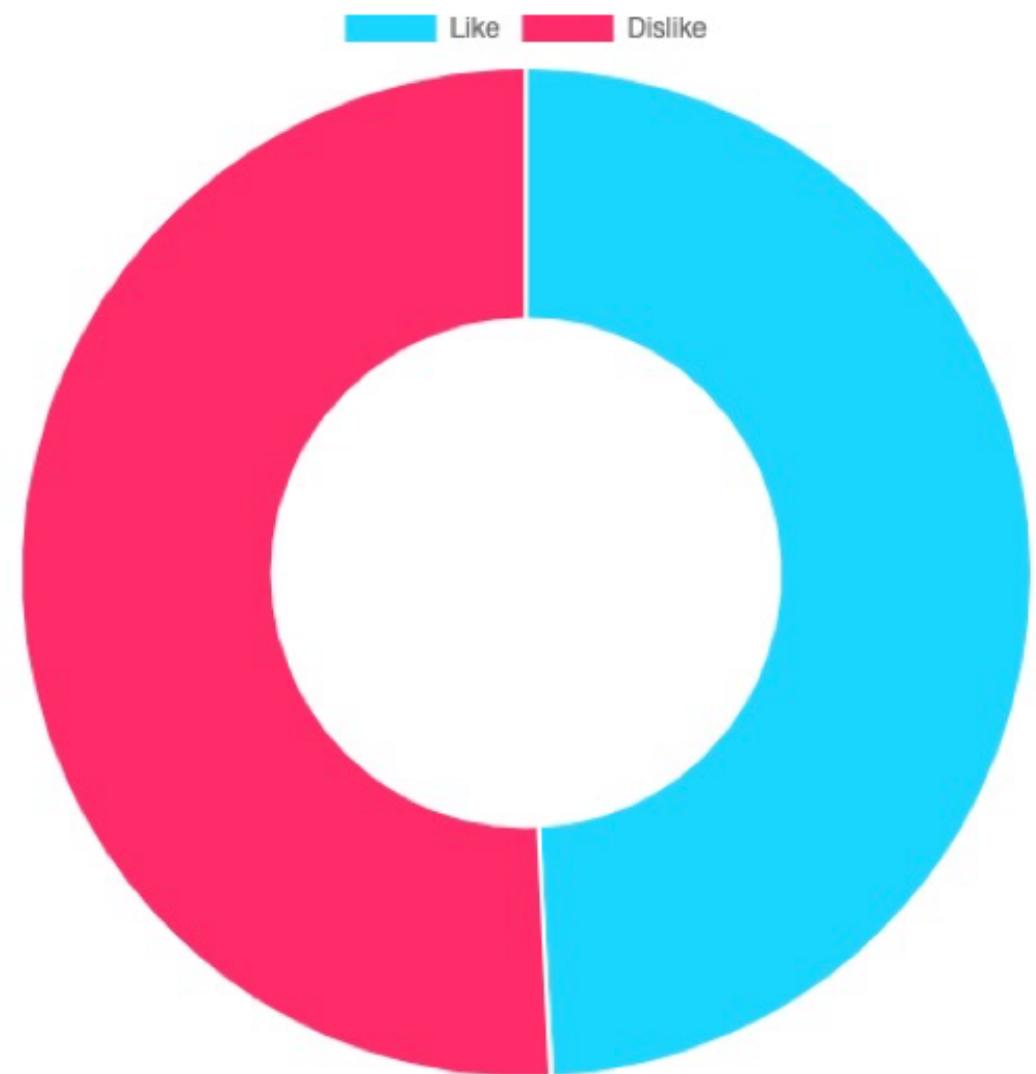
BUILDING THE DATASET: THE NEWS CATEGORIES



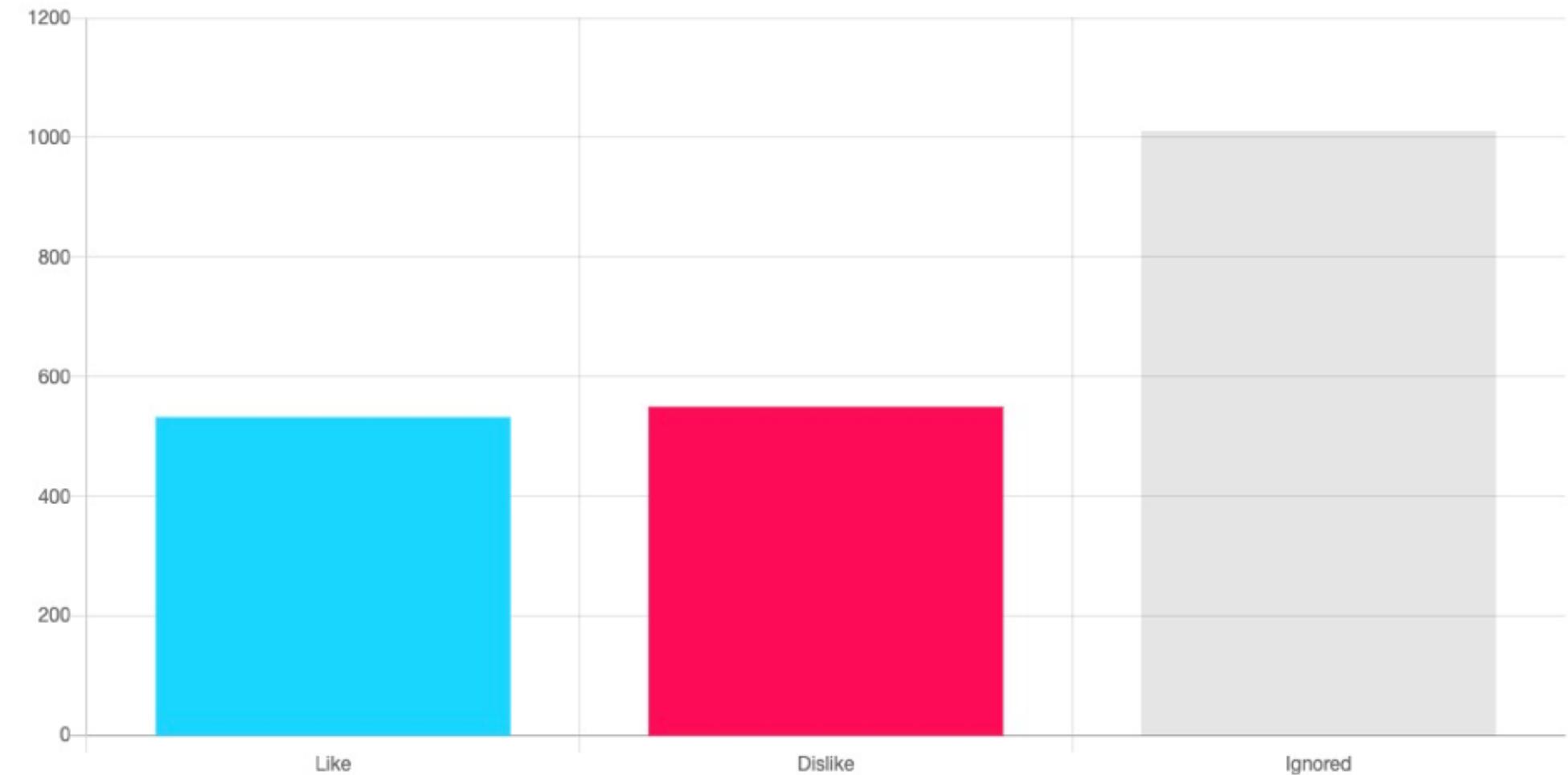
Tag distribution



BUILDING THE DATASET: THE LIKE/DISLIKE DISTRIBUTION



Like/Dislike
distribution



Extra articles, useful to improve the
category classification

EXTRACTING THE FEATURES



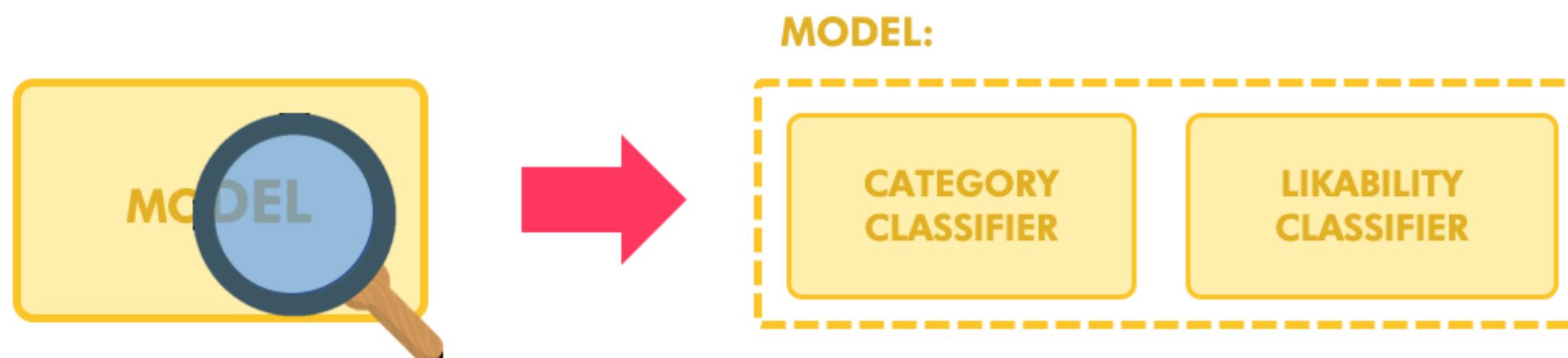
Now we need to mine the news in order to extract some useful features.

1. Natural Language Processing:
 - a. Normalization
 - b. Tokenization
 - c. Stemming
 - d. Ignoring Stopwords
2. Vectorizer

Input: Article title + article description

Output: Sparse matrix containing the TF-IDF scores for the article's terms

BUILDING THE MODEL



The machine learning model that we are going to use it's actually composed by **two classifiers**:



CATEGORY CLASSIFIER

Predict the category for the article knowing the matrix of TF-IDF scores.



LIKABILITY CLASSIFIER

Predict whether the user will like the article or not knowing the TF-IDF matrix and the category.

THE CATEGORY CLASSIFIER: BASE ESTIMATORS

Here are reported the **scores** and **performance** of some **simple classifiers** trying to predict the **category** of an article:

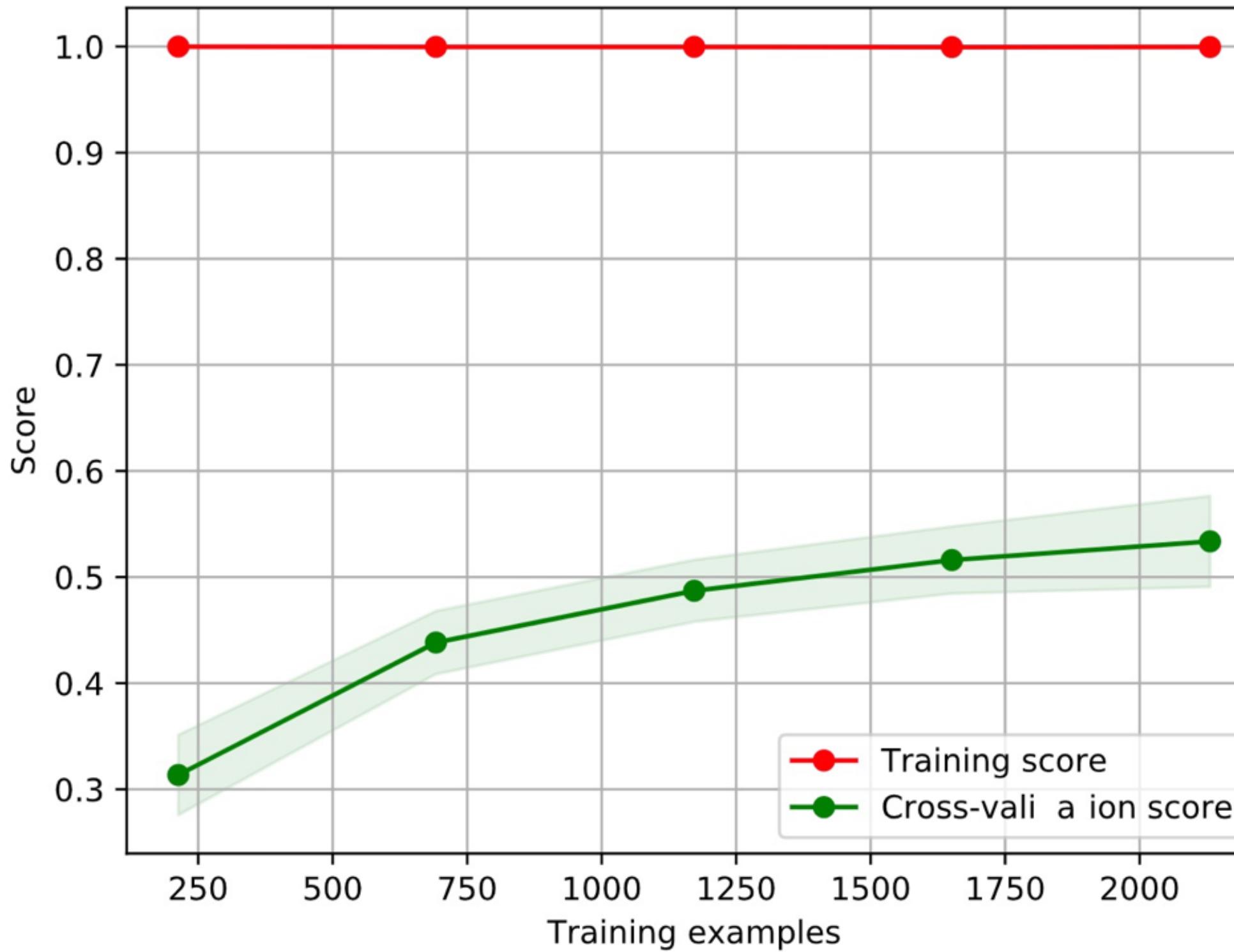
CLASSIFIER	F1-SCORE		ACCURACY		PRECISION		RECALL	
	MEAN	STD DEV	MEAN	STD DEV	MEAN	STD DEV	MEAN	STD DEV
Decision Tree	0.5276	0.0327	0.5267	0.0333	0.5447	0.0333	0.5267	0.0333
MN-Bayes	0.7704	0.0142	0.7729	0.0152	0.7899	0.0169	0.7729	0.0152
LinearSVC	0.8002	0.0203	0.8015	0.0195	0.8055	0.0211	0.8015	0.0195
LogisticRegression	0.7862	0.0209	0.7878	0.0209	0.7932	0.0223	0.7878	0.0209

Best score!

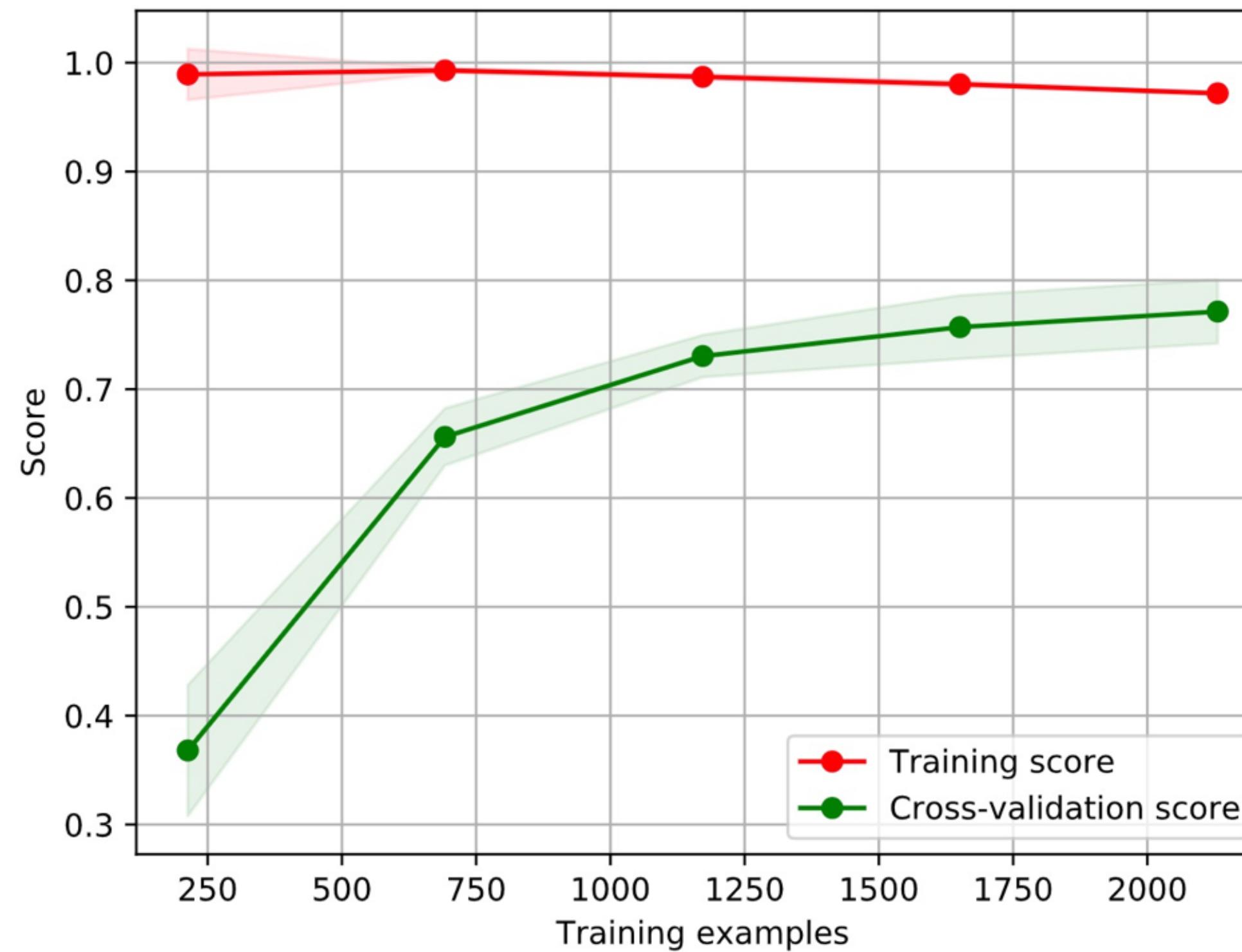


The following classifiers were **tuned** using a GridSearch algorithm! The testing and validation were performed using **10 Folds Cross Validation**

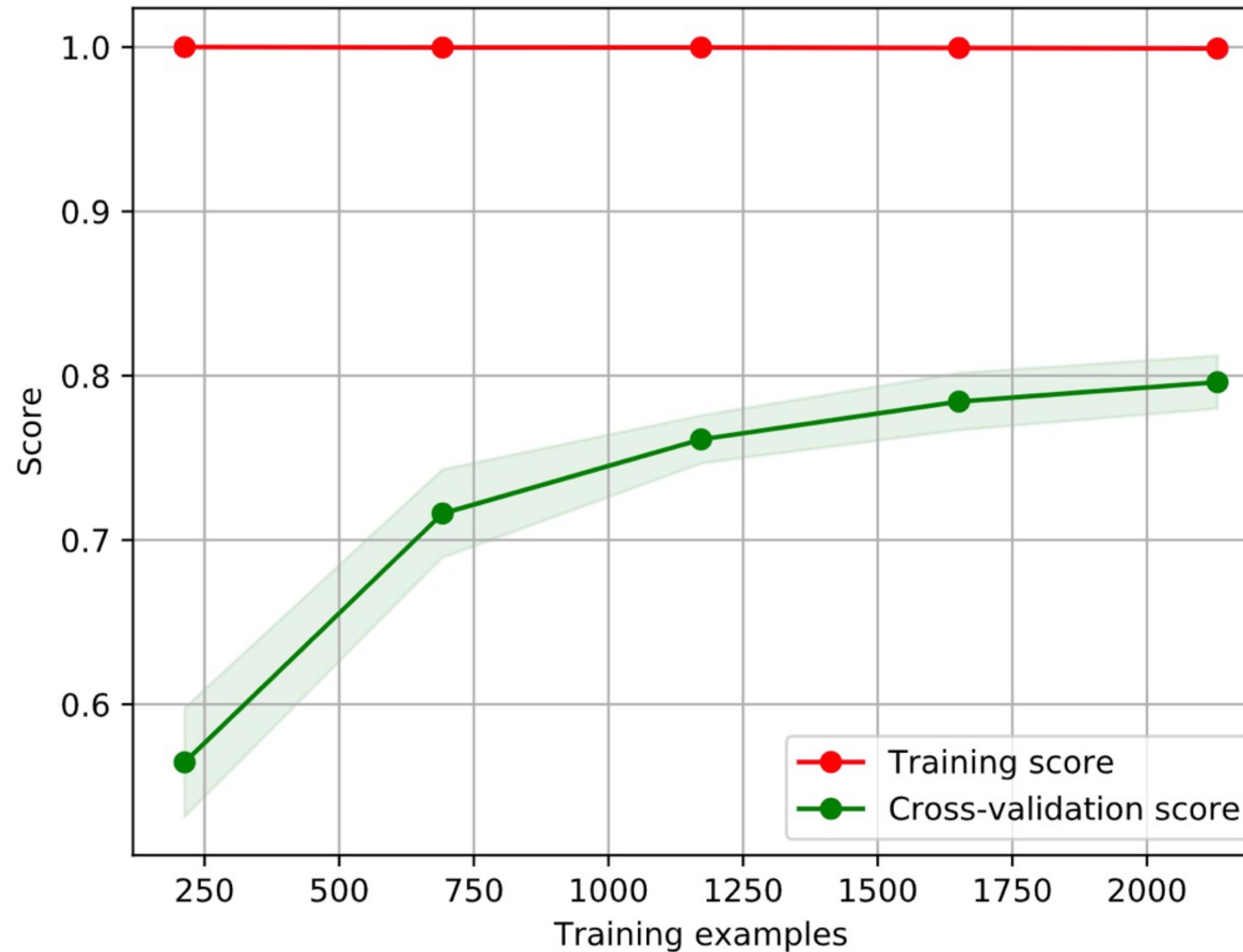
THE CATEGORY CLASSIFIER: DECISION TREE LEARNING CURVE



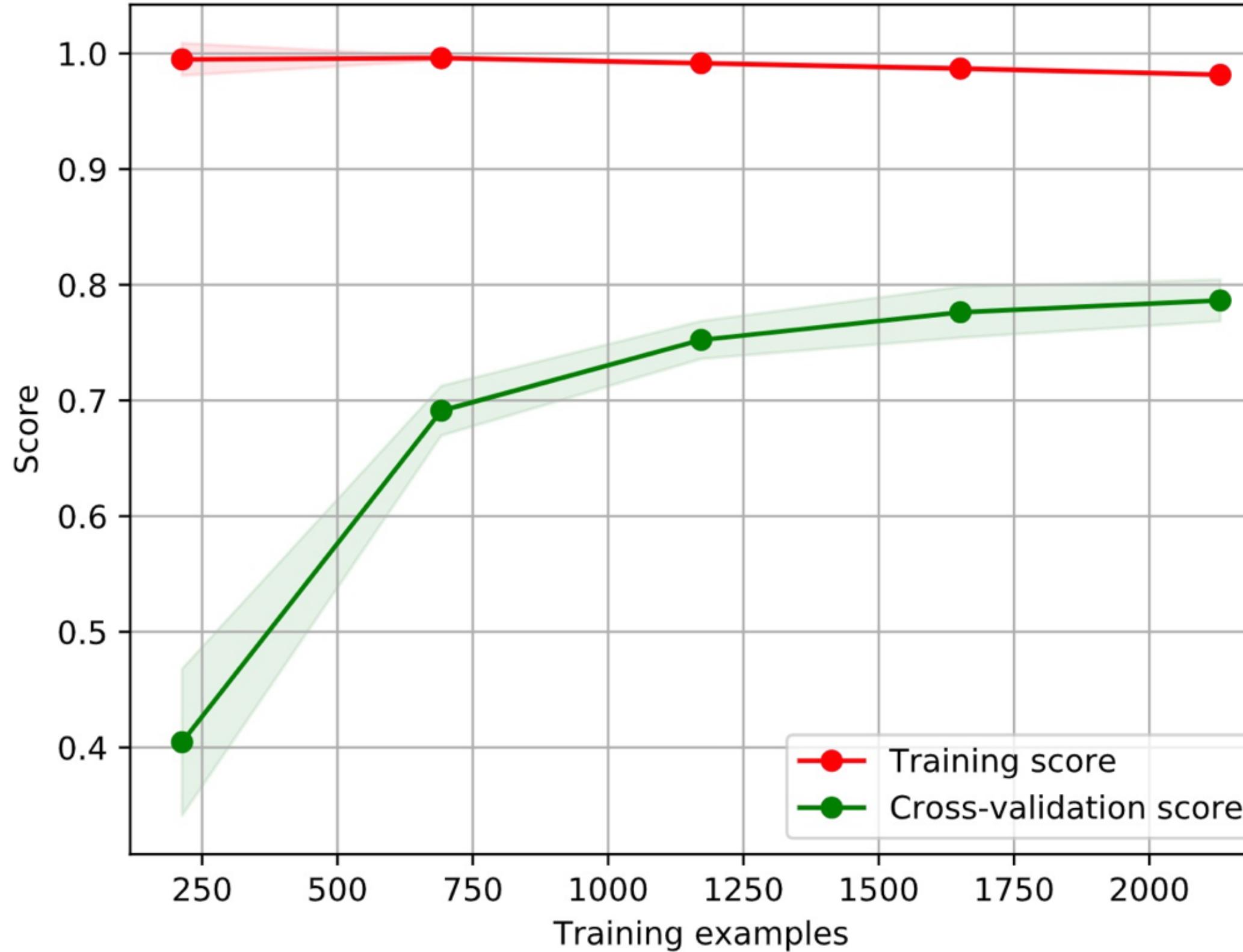
THE CATEGORY CLASSIFIER: MULTINOMIAL BAYES LEARNING CURVE



THE CATEGORY CLASSIFIER: LINEAR SVC LEARNING CURVE



THE CATEGORY CLASSIFIER: LOGISTIC REGRESSION LEARNING CURVE



THE CATEGORY CLASSIFIER: ENSEMBLES AND METACLASSIFIERS

Here are reported the **scores** and **performance** of some **ensemble classifiers** trying to predict the **category** of an article:

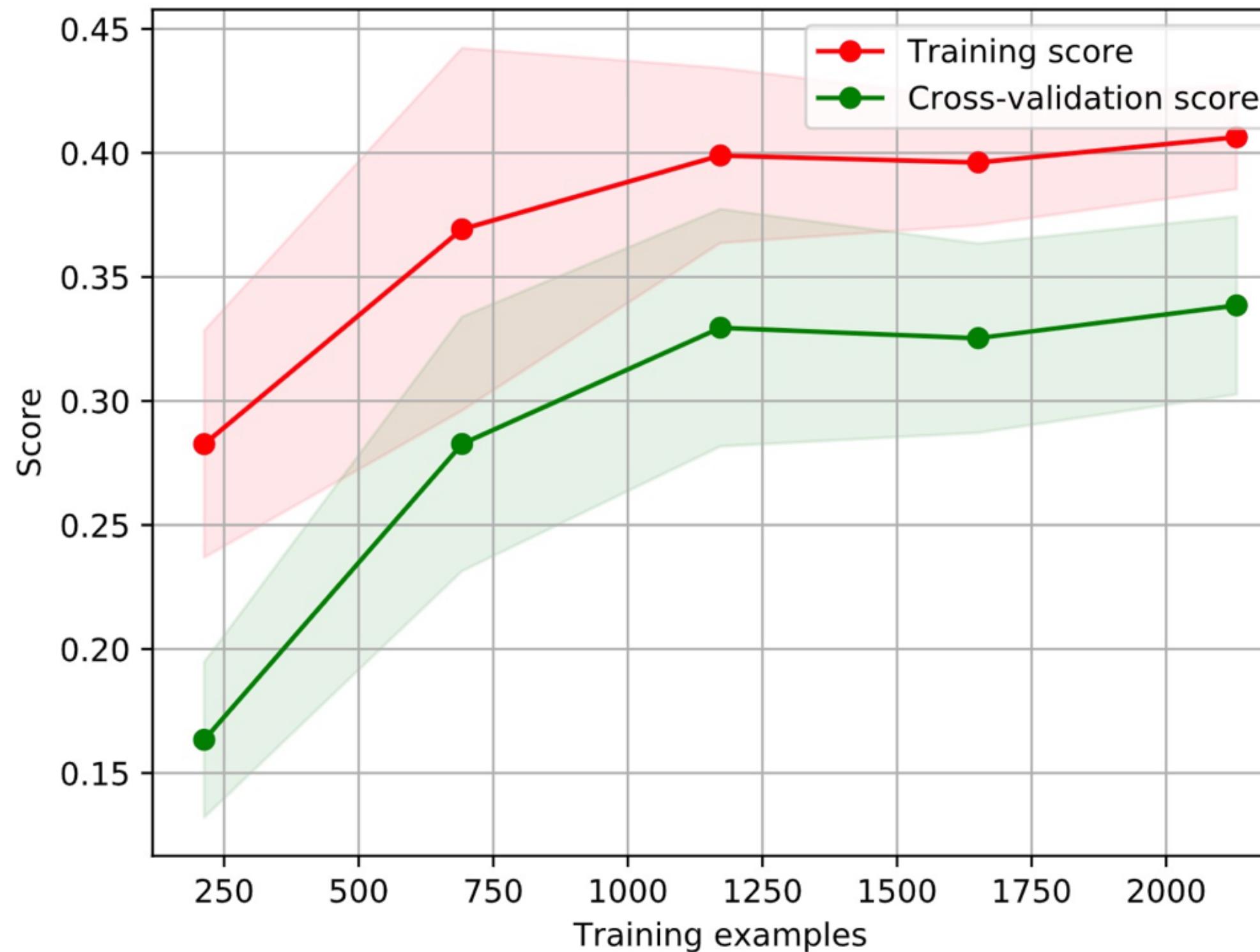
CLASSIFIER	F1-SCORE		ACCURACY		PRECISION		RECALL	
	MEAN	STD DEV	MEAN	STD DEV	MEAN	STD DEV	MEAN	STD DEV
D. Tree + ADABoost	0.3617	0.0347	0.3597	0.0248	0.5272	0.0493	0.3597	0.0248
Random Forest	0.6852	0.0286	0.6869	0.0274	0.6949	0.0285	0.6869	0.0274
Voting	0.7911	0.0208	0.7924	0.0207	0.7989	0.0228	0.7924	0.0207
Bagging	0.7960	0.0203	0.7970	0.0195	0.8016	0.0211	0.7970	0.0195
XGBClassifier	0.6649	0.0224	0.6631	0.0228	0.6753	0.0209	0.6631	0.0228
Stacking	0.7541	0.0147	0.7557	0.0147	0.7686	0.0127	0.7557	0.0147

Best score
but worse
than SVC



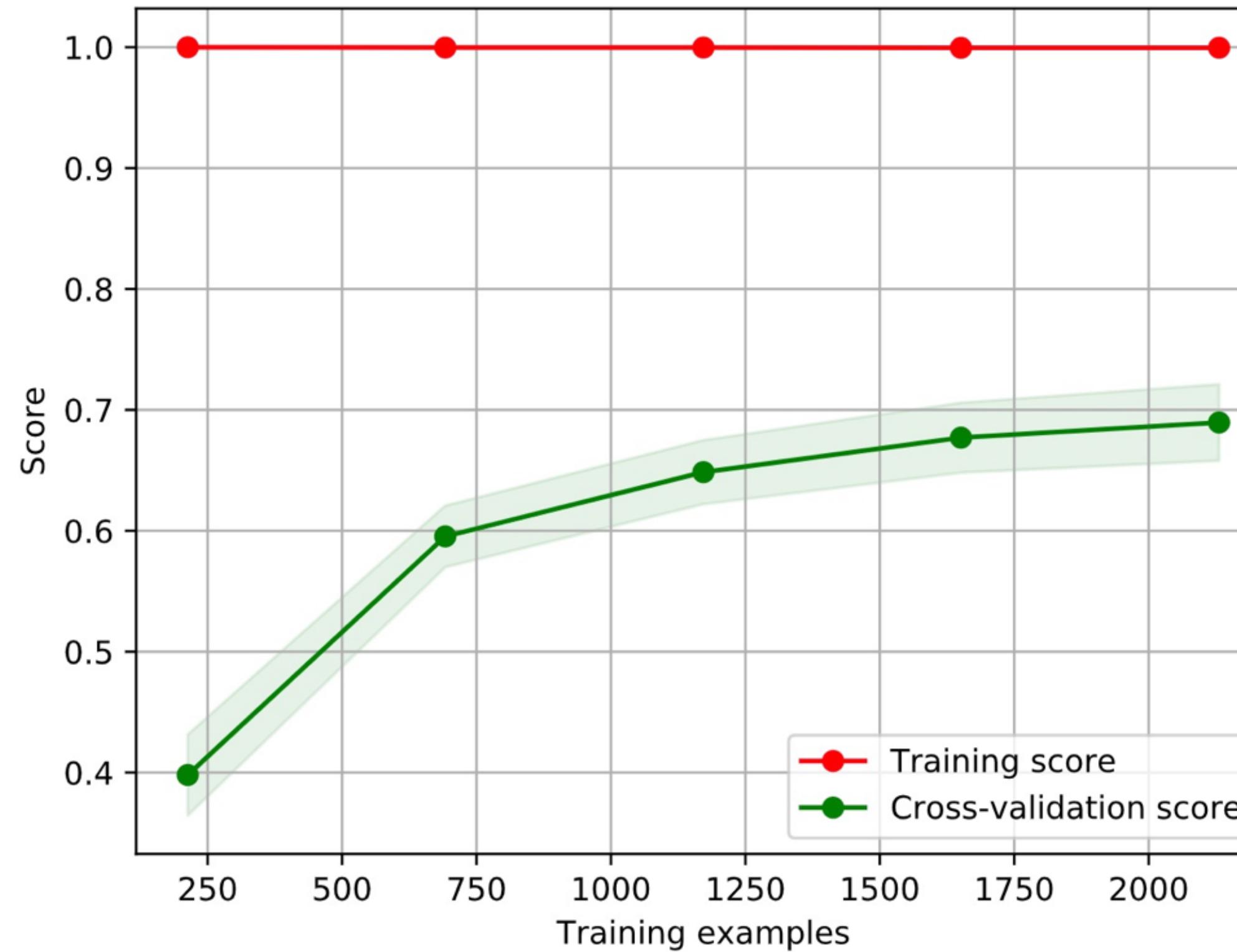
The following classifiers were **tuned** using a GridSearch algorithm! The testing and validation were performed using **10 Folds Cross Validation**

THE CATEGORY CLASSIFIER: ADABoost LEARNING CURVE



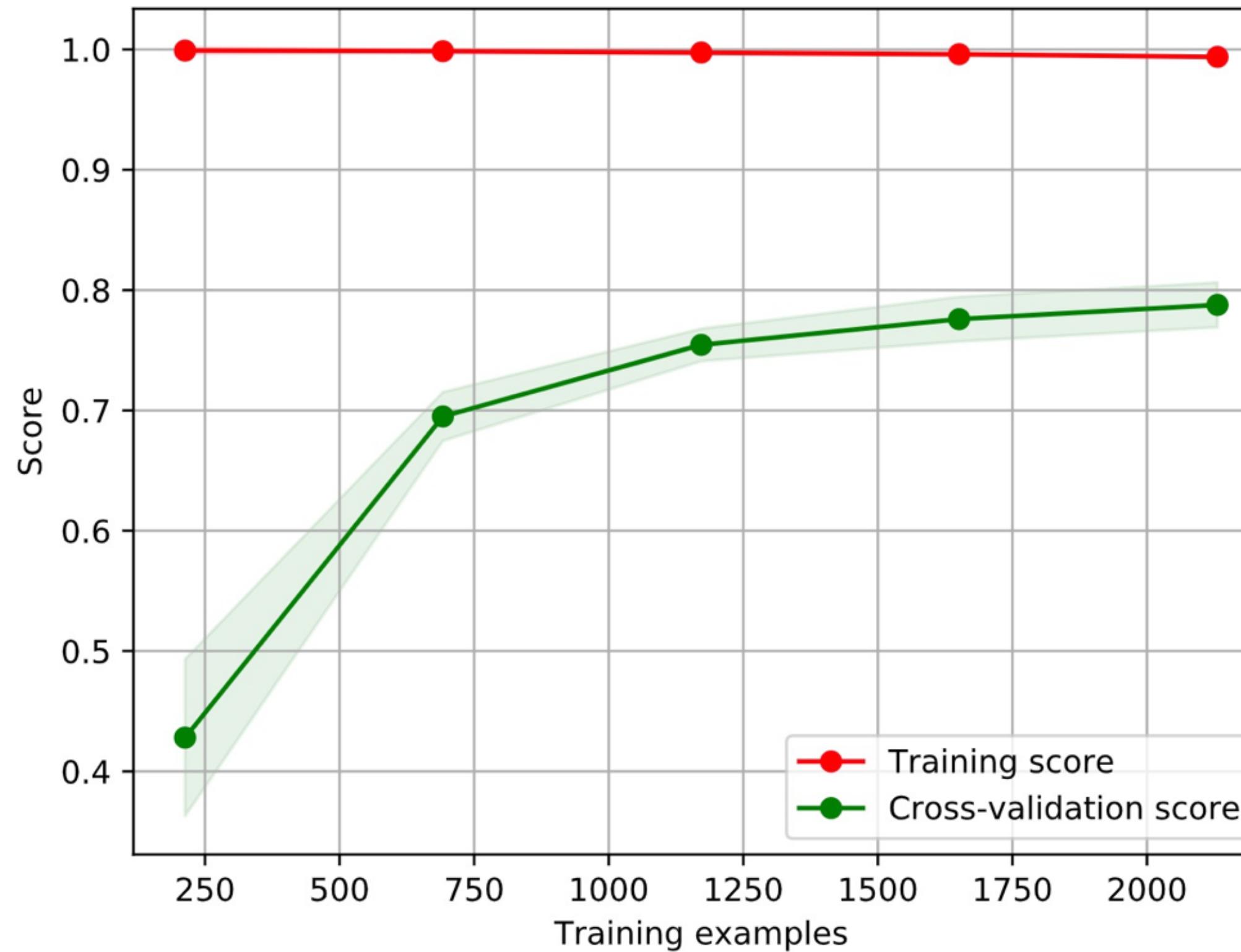
```
parameters {  
    'base': DecisionTree(),  
    'n_estimators': 100  
}
```

THE CATEGORY CLASSIFIER: RANDOM FOREST LEARNING CURVE

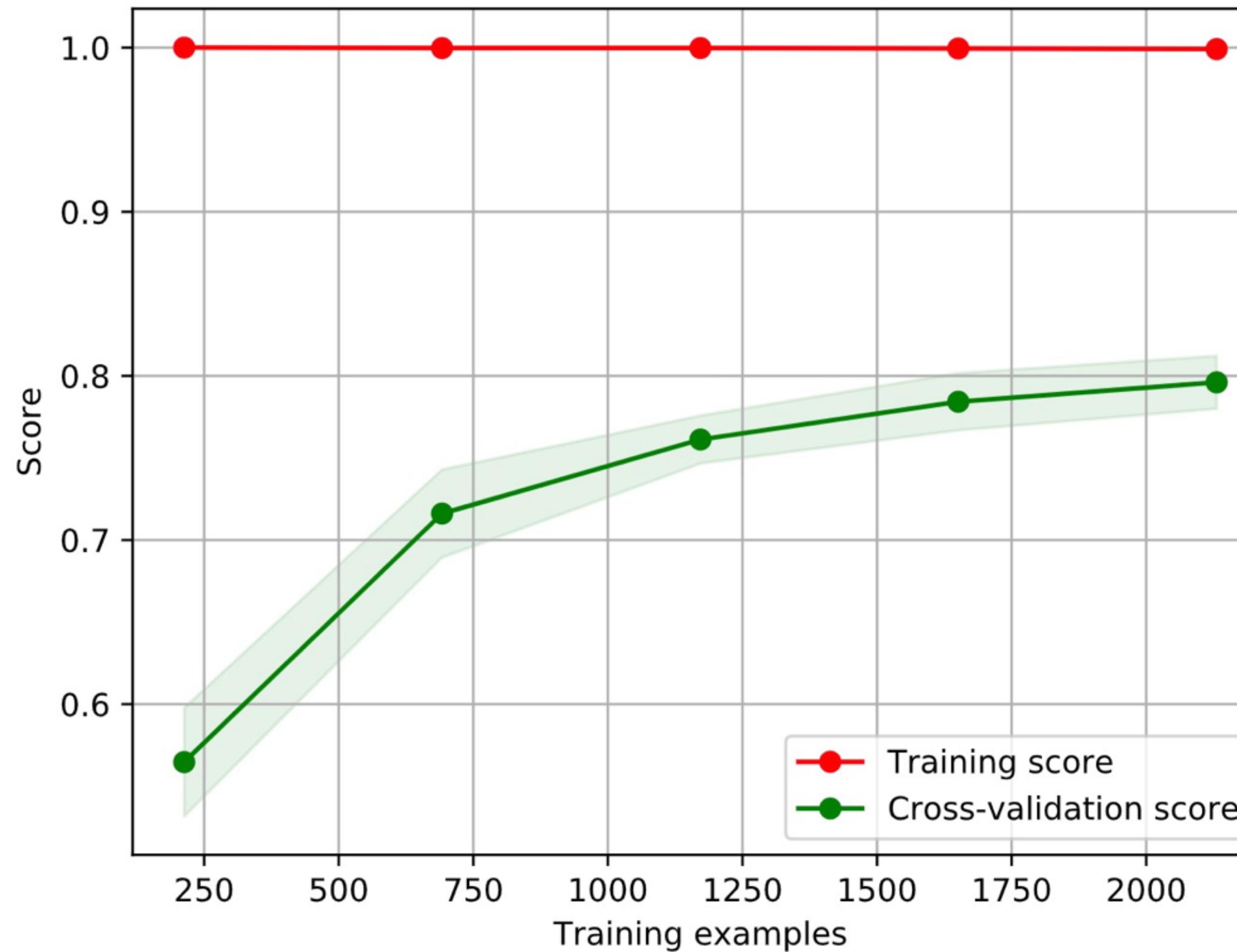


```
parameters {  
    'n_estimators': 100  
}
```

THE CATEGORY CLASSIFIER: VOTING LEARNING CURVE

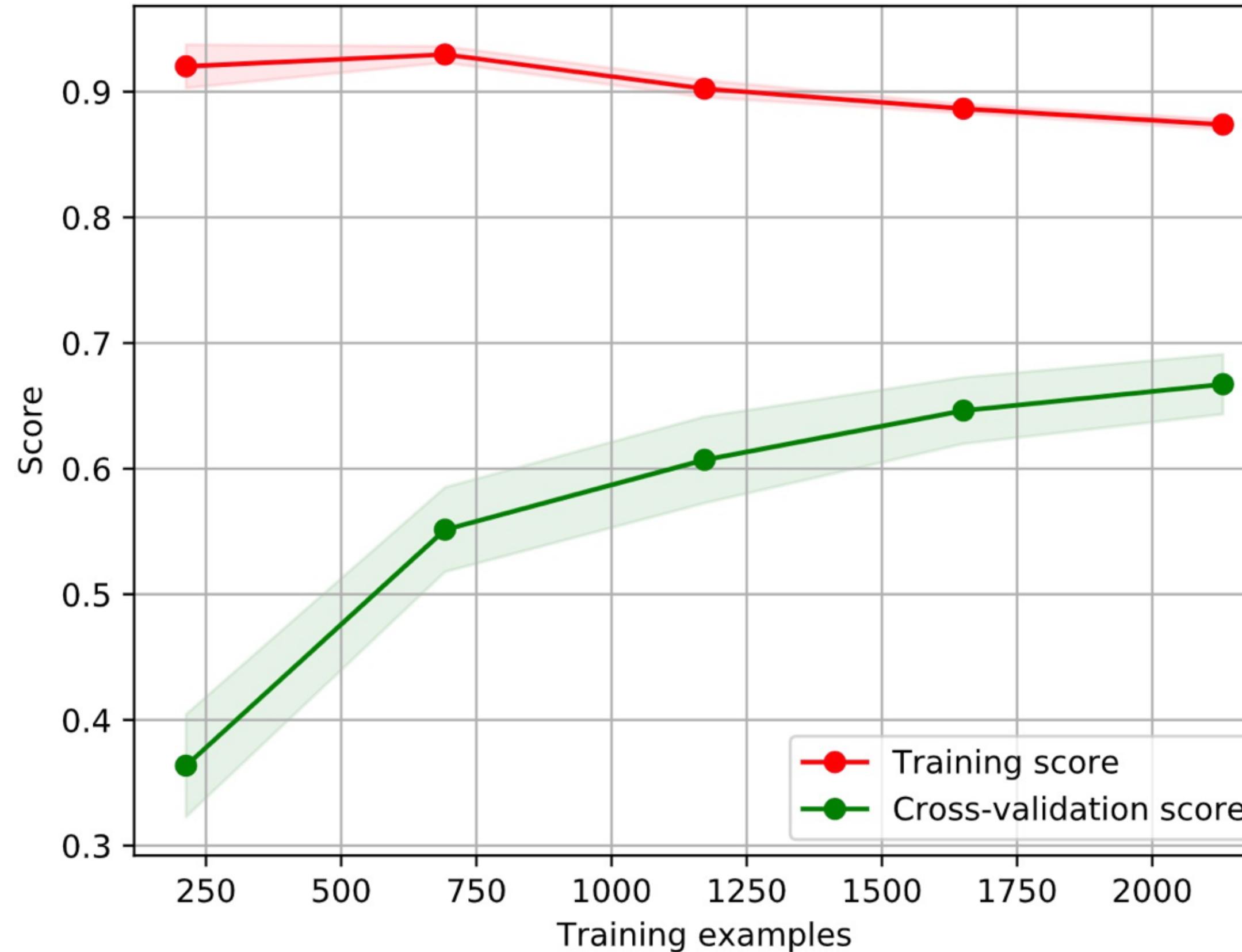


THE CATEGORY CLASSIFIER: BAGGING LEARNING CURVE



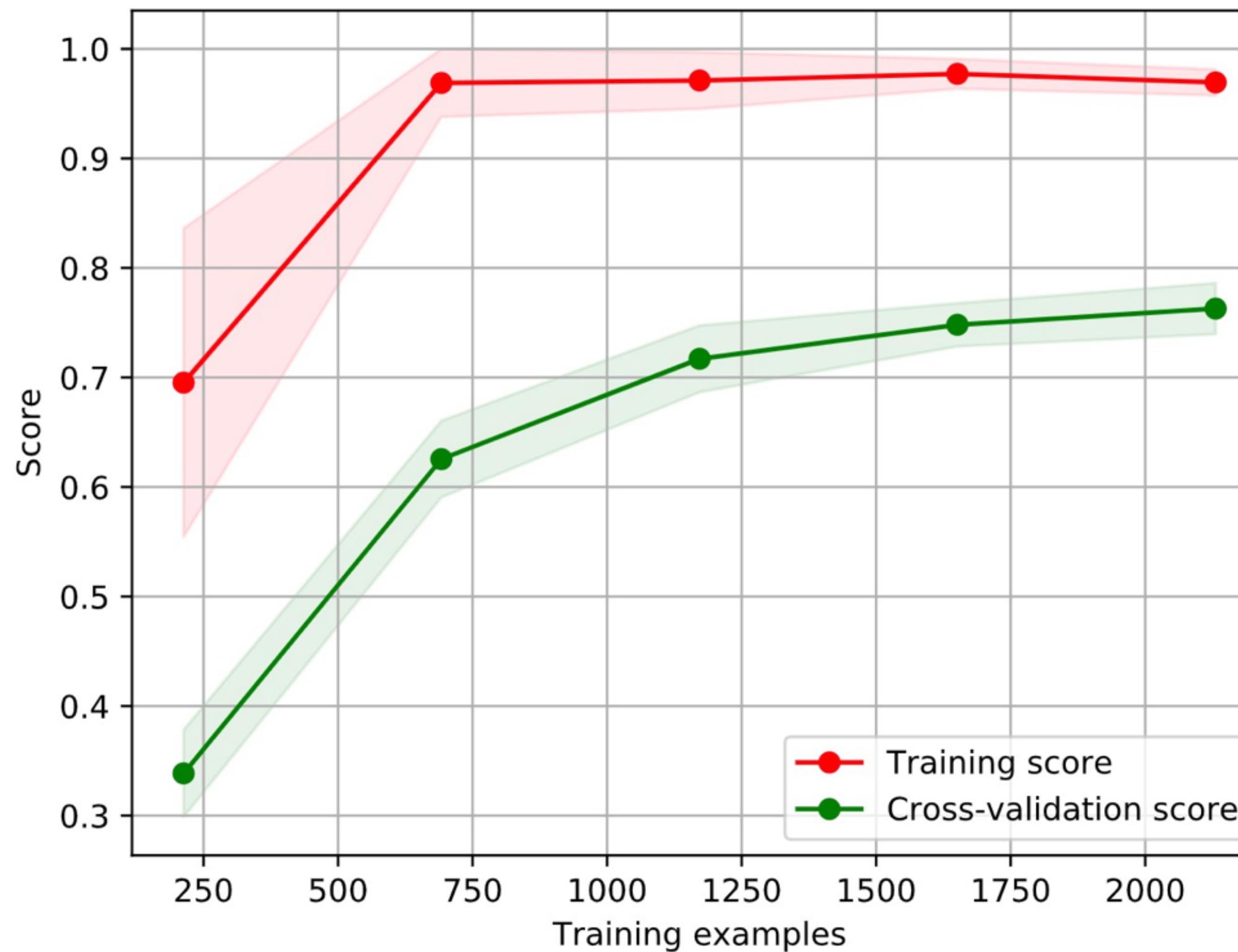
```
parameters {  
    'base': LinearSVC(),  
    'n_estimators': 100  
}
```

THE CATEGORY CLASSIFIER: XGB LEARNING CURVE



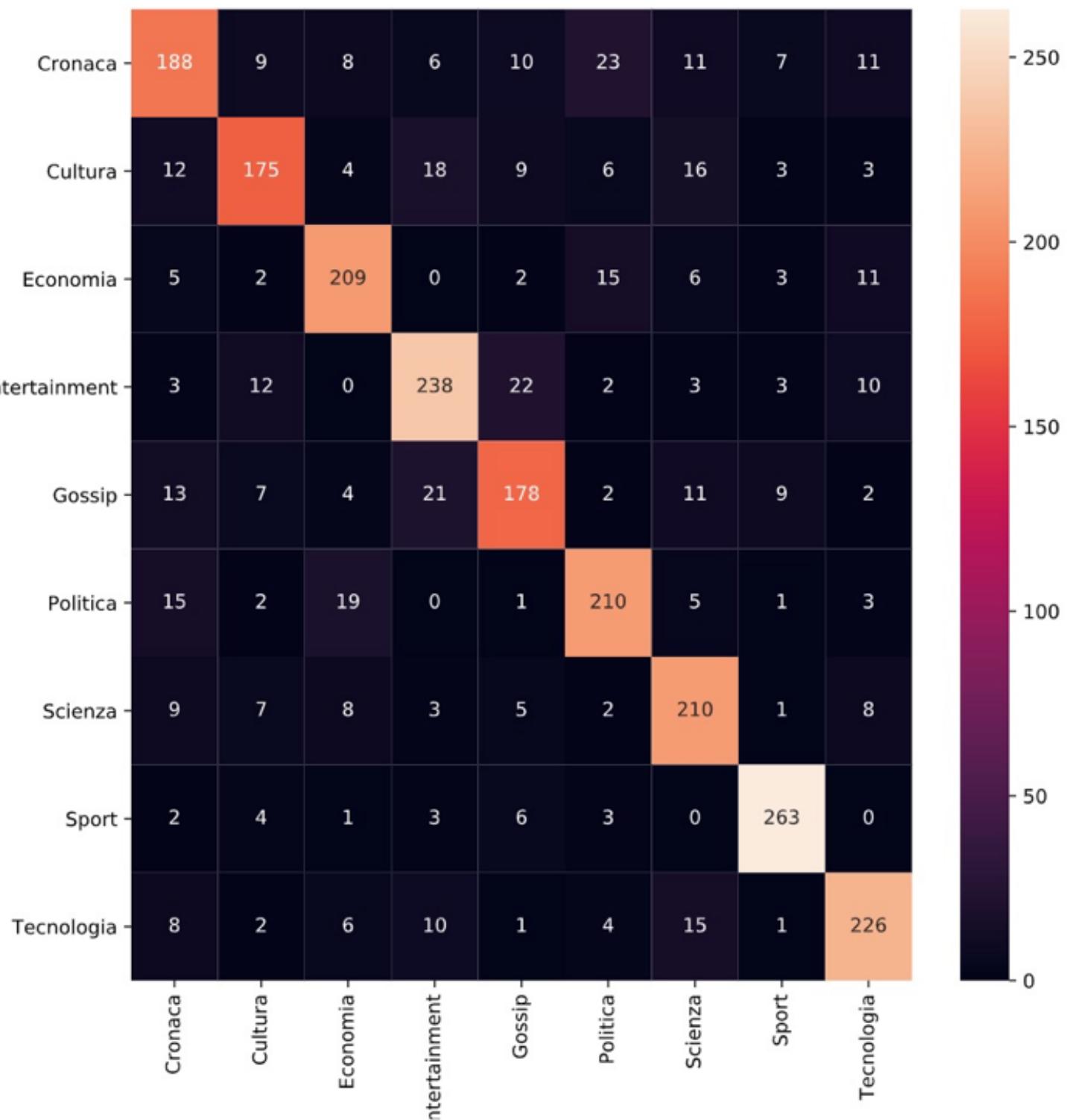
```
parameters {  
    'n_estimators': 100,  
    'max_depth': 4,  
    'learning_rate': 0.9  
}
```

THE CATEGORY CLASSIFIER: STACKING LEARNING CURVE



THE CATEGORY CLASSIFIER

In the end, no classifier did better than the base **LinearSVC**, so it was chosen as the classifier that should predict the article category. Here is shown its **confusion matrix**:



The chosen classifier has an 80% accuracy, is this good?

Indeed it could be better, however predicting the category of an article is not an easy task (not even for a human), so in the end we can say that it is fair enough.

THE LIKABILITY CLASSIFIER: BASE ESTIMATORS

Here are reported the **scores** and **performance** of some **simple classifiers** trying to predict the **likability** of an article:

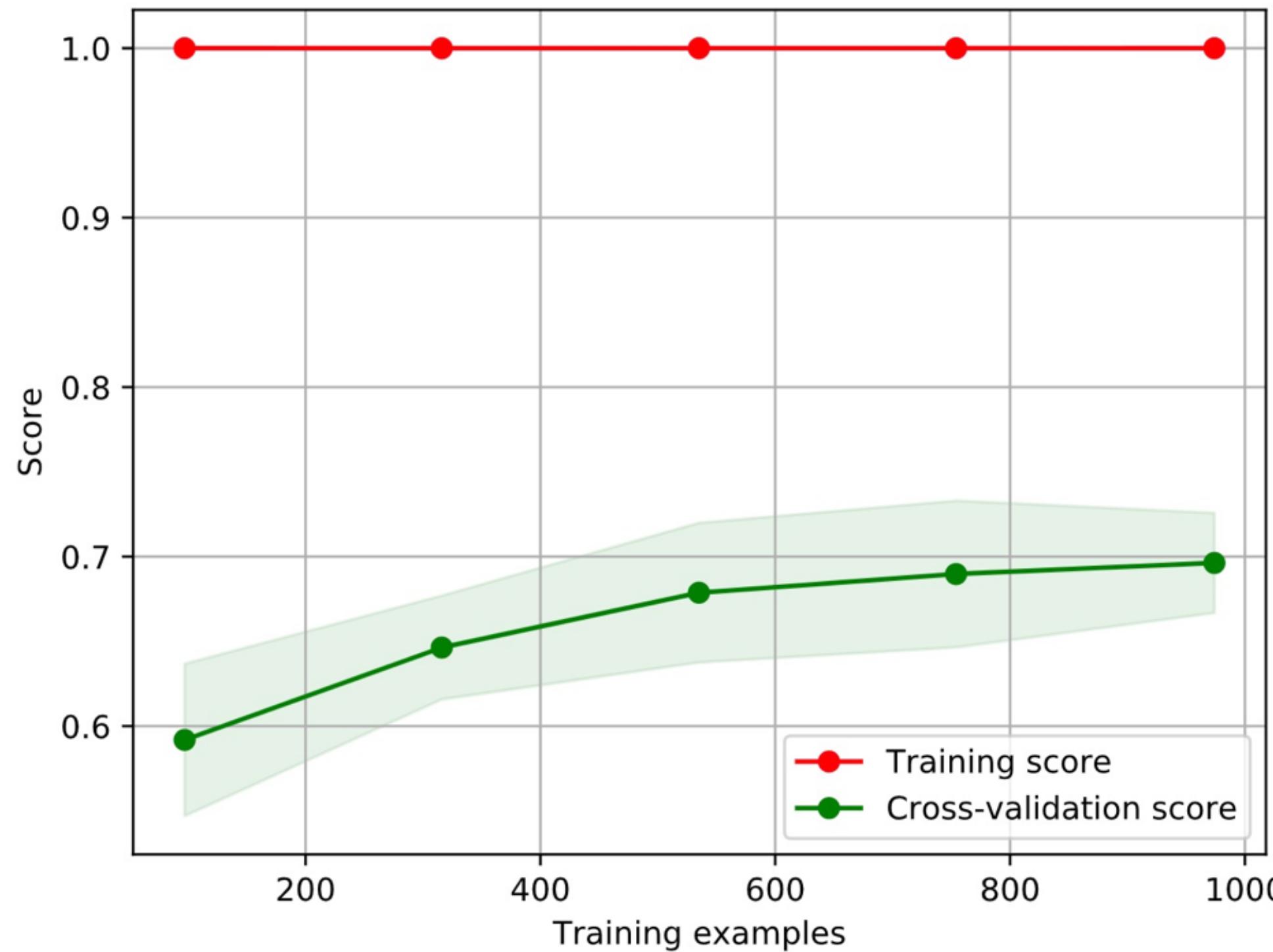
CLASSIFIER	F1-SCORE		ACCURACY		PRECISION		RECALL	
	MEAN	STD DEV	MEAN	STD DEV	MEAN	STD DEV	MEAN	STD DEV
Decision Tree	0.6967	0.0275	0.6972	0.0275	0.6983	0.0276	0.6972	0.0275
MN-Bayes	0.8607	0.0244	0.8615	0.0241	0.8680	0.0242	0.8615	0.0241
LinearSVC	0.8669	0.0286	0.8670	0.0286	0.8690	0.0287	0.8670	0.0286
LogisticRegression	0.8725	0.0312	0.8726	0.0312	0.8739	0.0316	0.8726	0.0312

Best score!

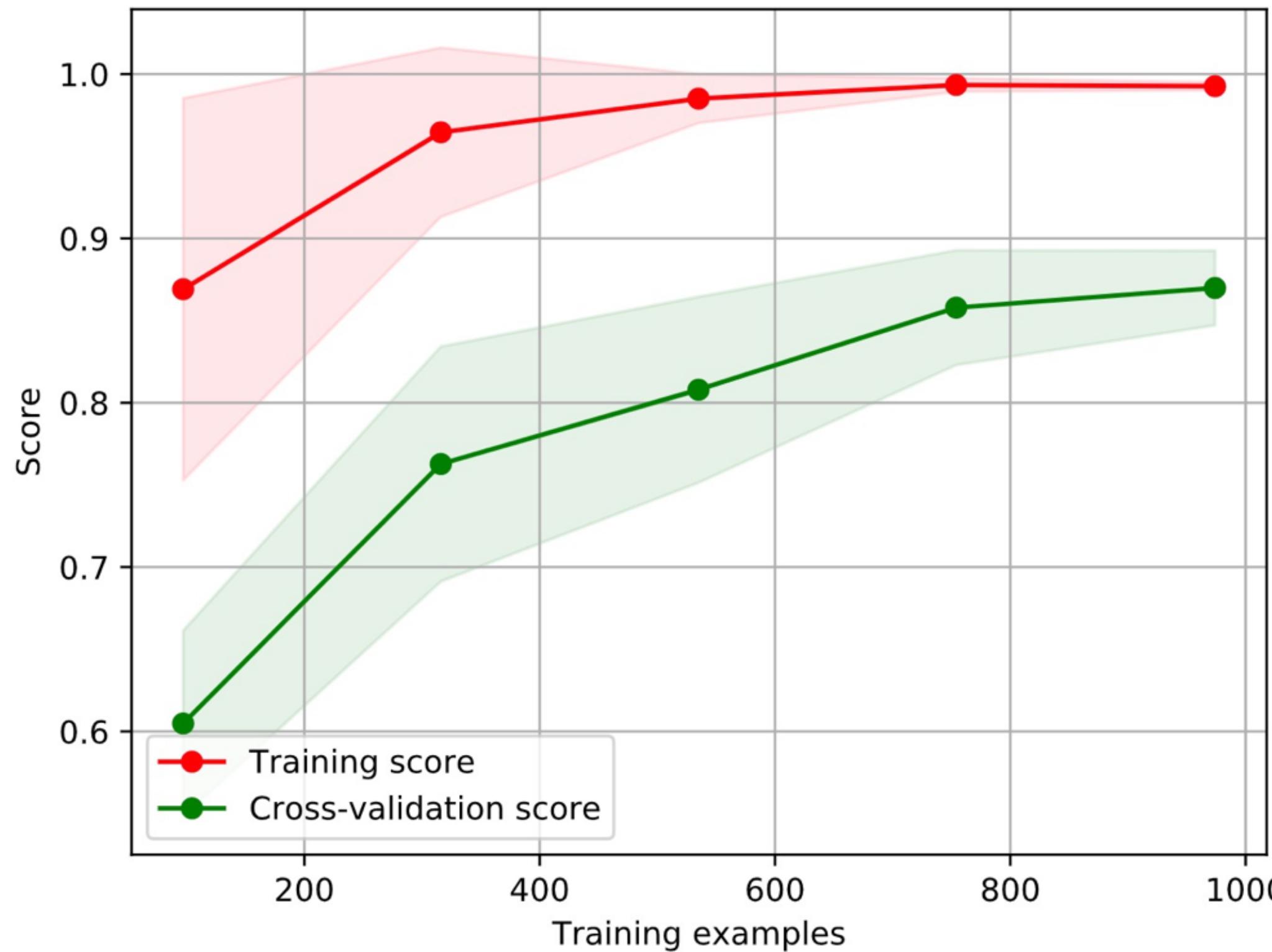


The following classifiers were **tuned** using a GridSearch algorithm! The testing and validation were performed using **10 Folds Cross Validation**

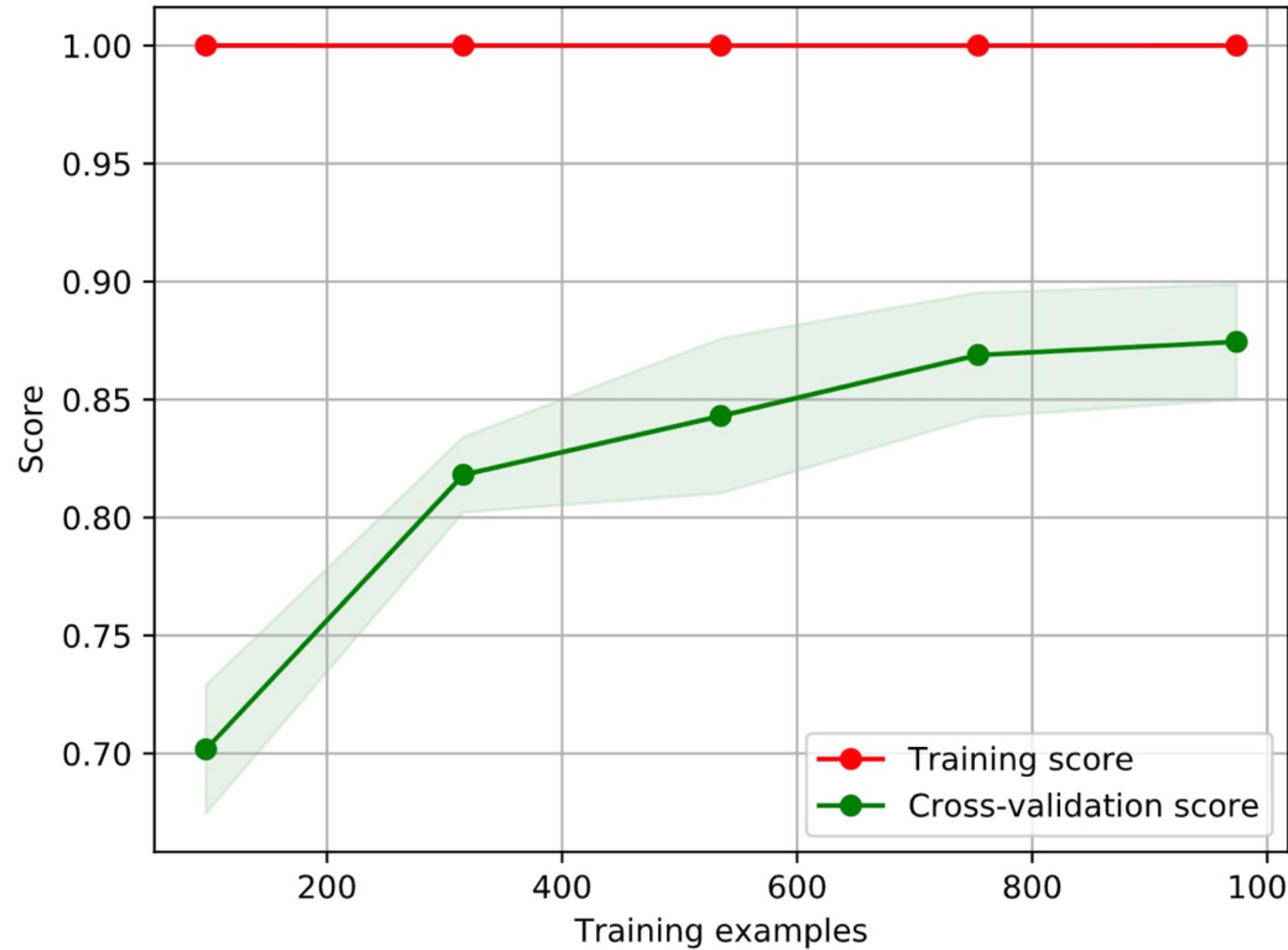
THE LIKABILITY CLASSIFIER: DECISION TREE LEARNING CURVE



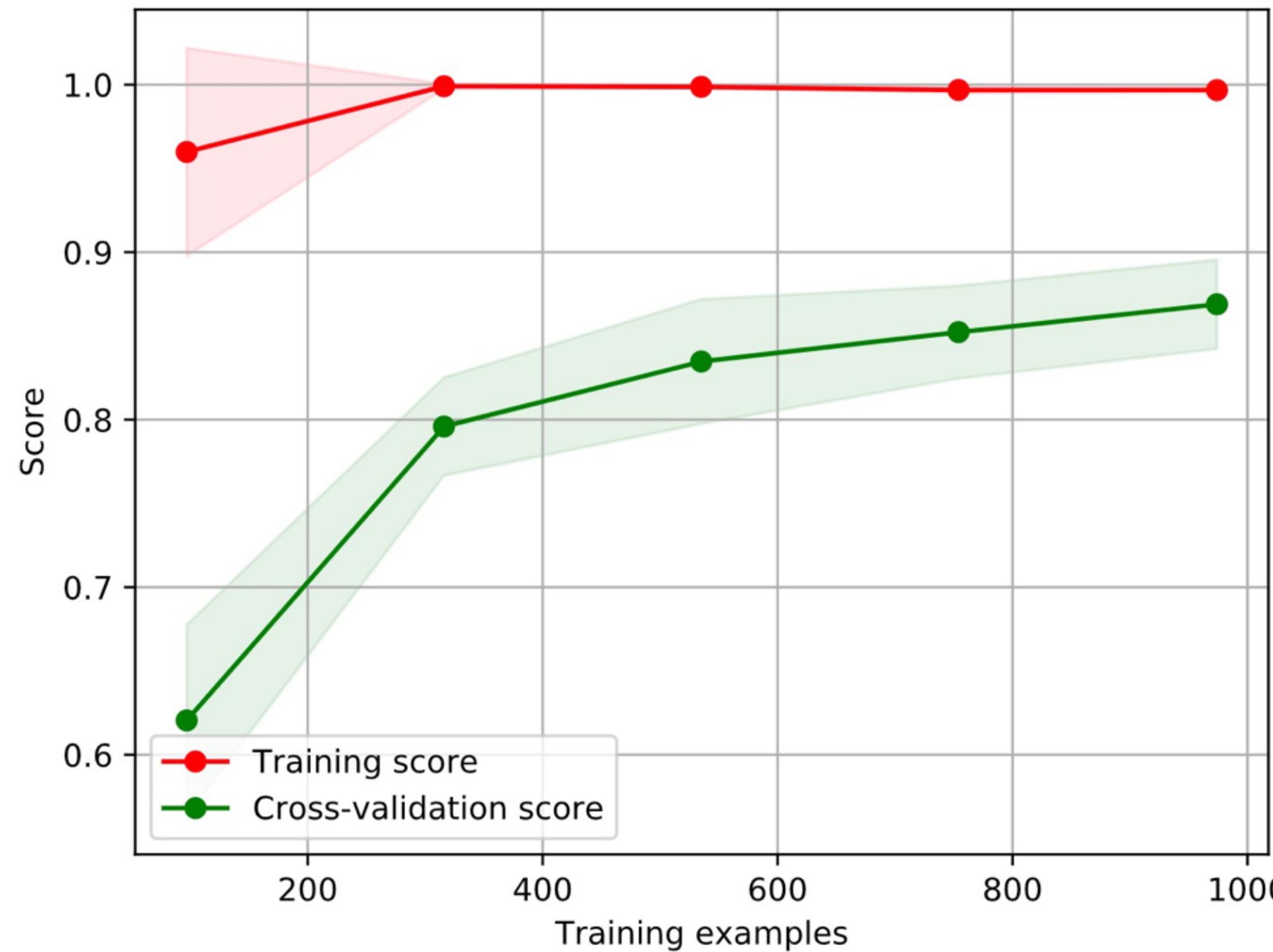
THE LIKABILITY CLASSIFIER: MULTINOMIAL BAYES LEARNING CURVE



THE LIKABILITY CLASSIFIER: LINEAR SVC LEARNING CURVE



THE LIKABILITY CLASSIFIER: LOGISTIC REGRESSION LEARNING CURVE



THE LIKABILITY CLASSIFIER: ENSEMBLES AND METACLASSIFIERS

Here are reported the **scores** and **performance** of some **ensemble classifiers** trying to predict the **category** of an article:

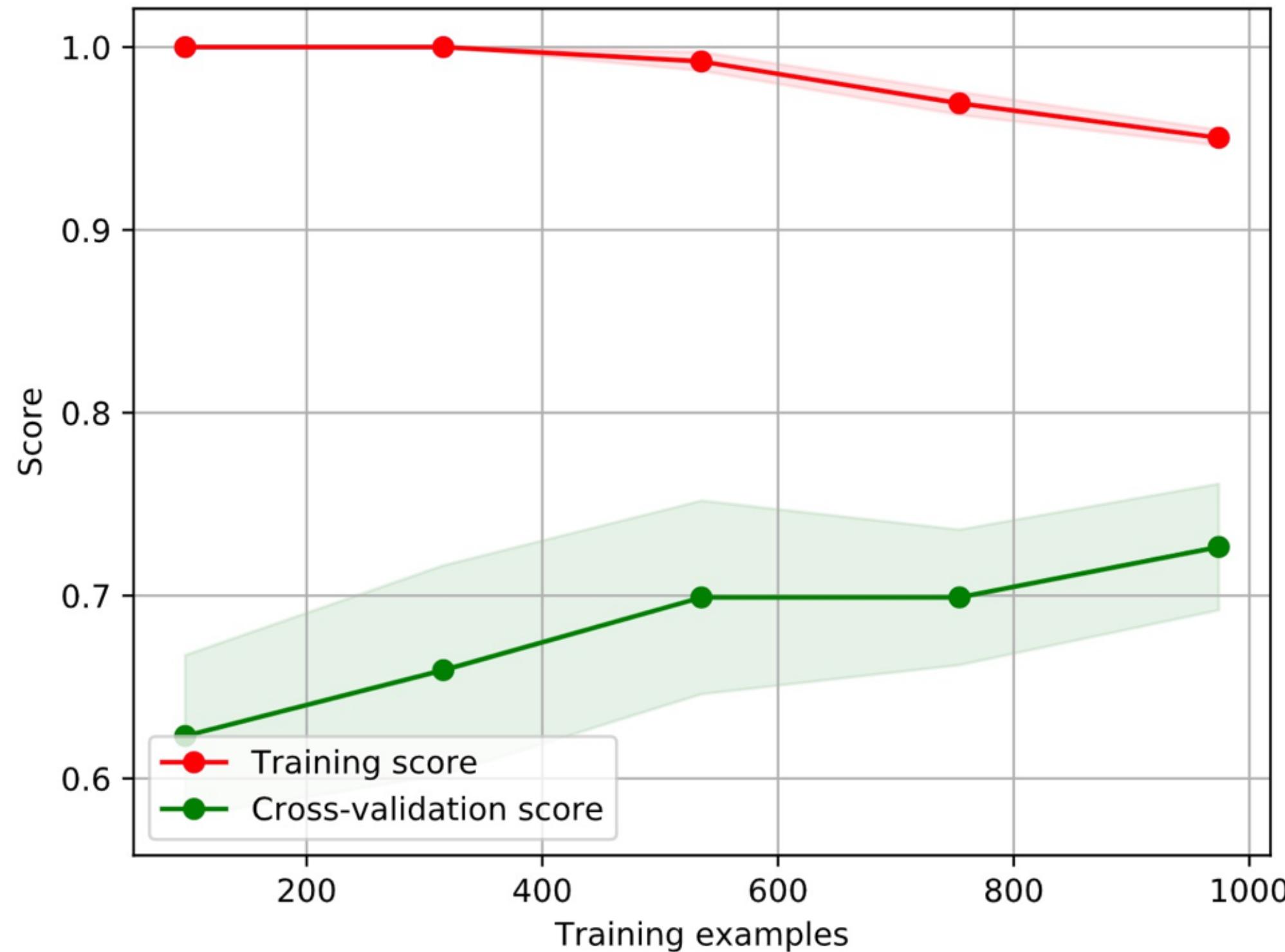
CLASSIFIER	F1-SCORE		ACCURACY		PRECISION		RECALL	
	MEAN	STD DEV	MEAN	STD DEV	MEAN	STD DEV	MEAN	STD DEV
D. Tree + ADABoost	0.7358	0.0324	0.7369	0.0322	0.7397	0.0326	0.7369	0.0322
Random Forest	0.7976	0.0312	0.7996	0.0302	0.8087	0.0271	0.7996	0.0302
Voting	0.8714	0.0339	0.8717	0.0338	0.8739	0.0341	0.8717	0.0338
Bagging	0.8669	0.0286	0.8670	0.0286	0.8690	0.0287	0.8670	0.0286
XGBClassifier	0.7234	0.0410	0.7267	0.0402	0.7356	0.423	0.7267	0.0402

Best score
but not
better than
LogReg



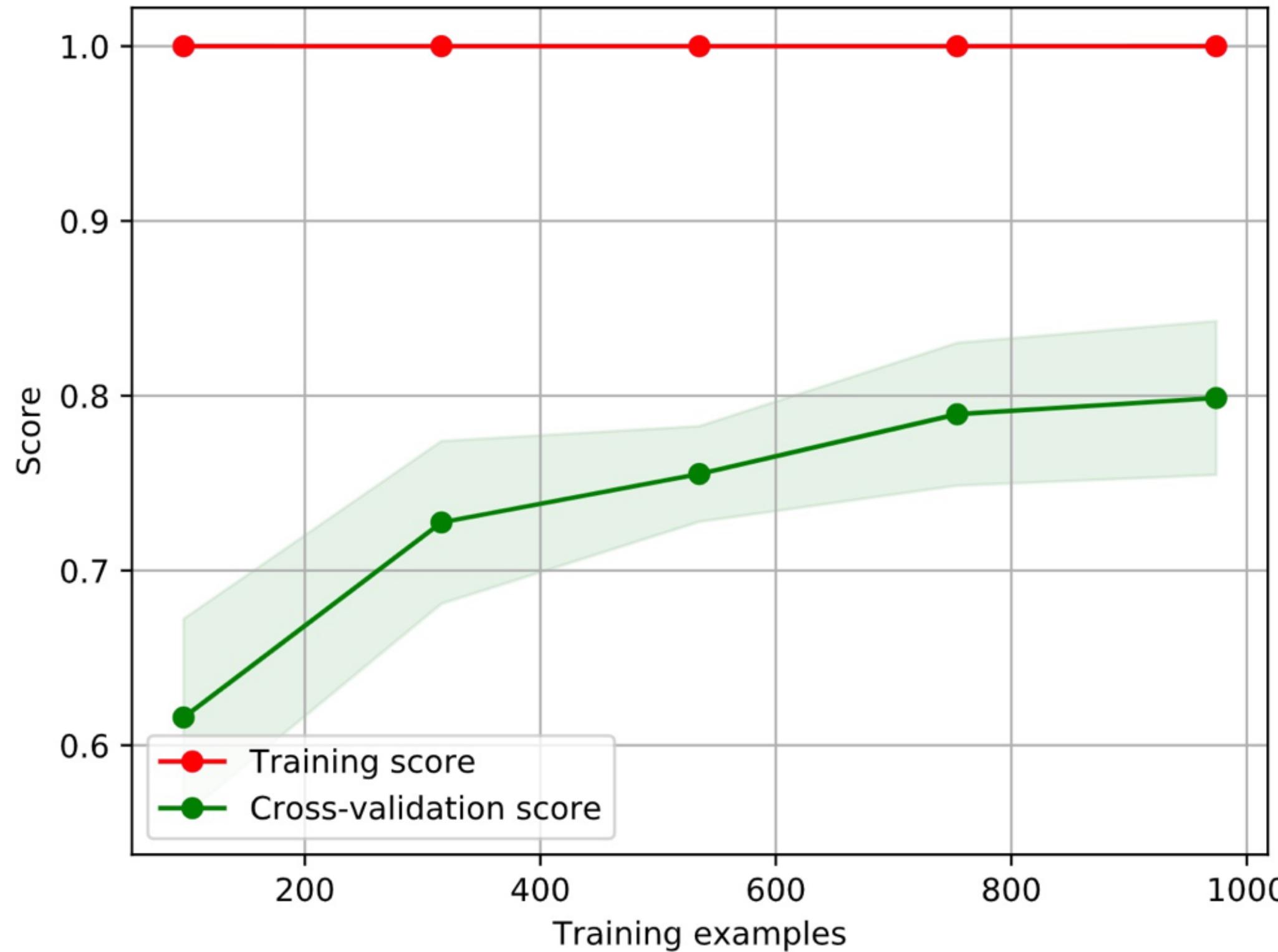
The following classifiers were **tuned** using a GridSearch algorithm! The testing and validation were performed using **10 Folds Cross Validation**

THE LIKABILITY CLASSIFIER: ADABOOST LEARNING CURVE



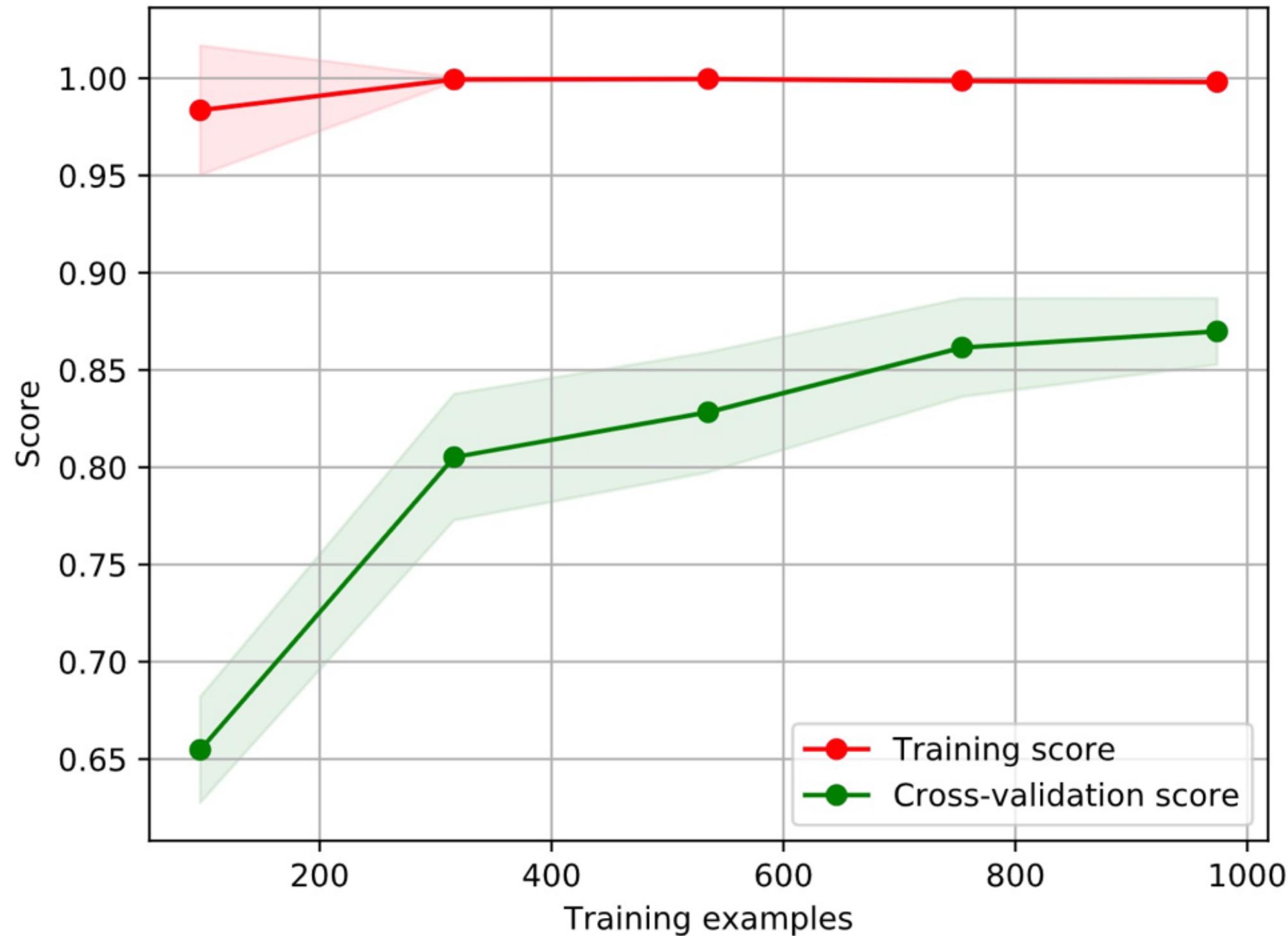
```
parameters {  
    'base': DecisionTree(),  
    'n_estimators': 100  
}
```

THE LIKABILITY CLASSIFIER: RANDOM FOREST LEARNING CURVE

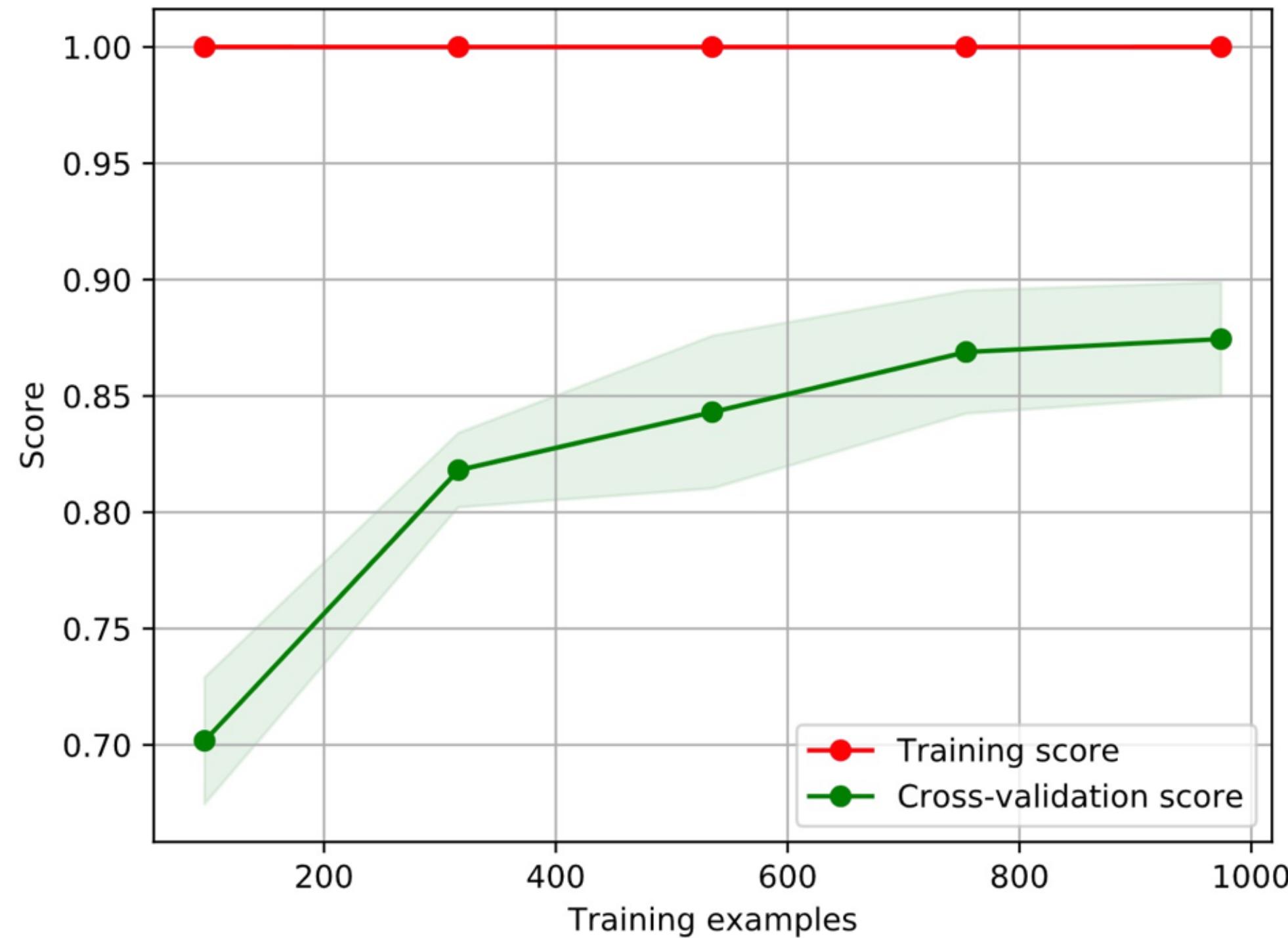


```
parameters {  
    'n_estimators': 100  
}
```

THE LIKABILITY CLASSIFIER: VOTING LEARNING CURVE

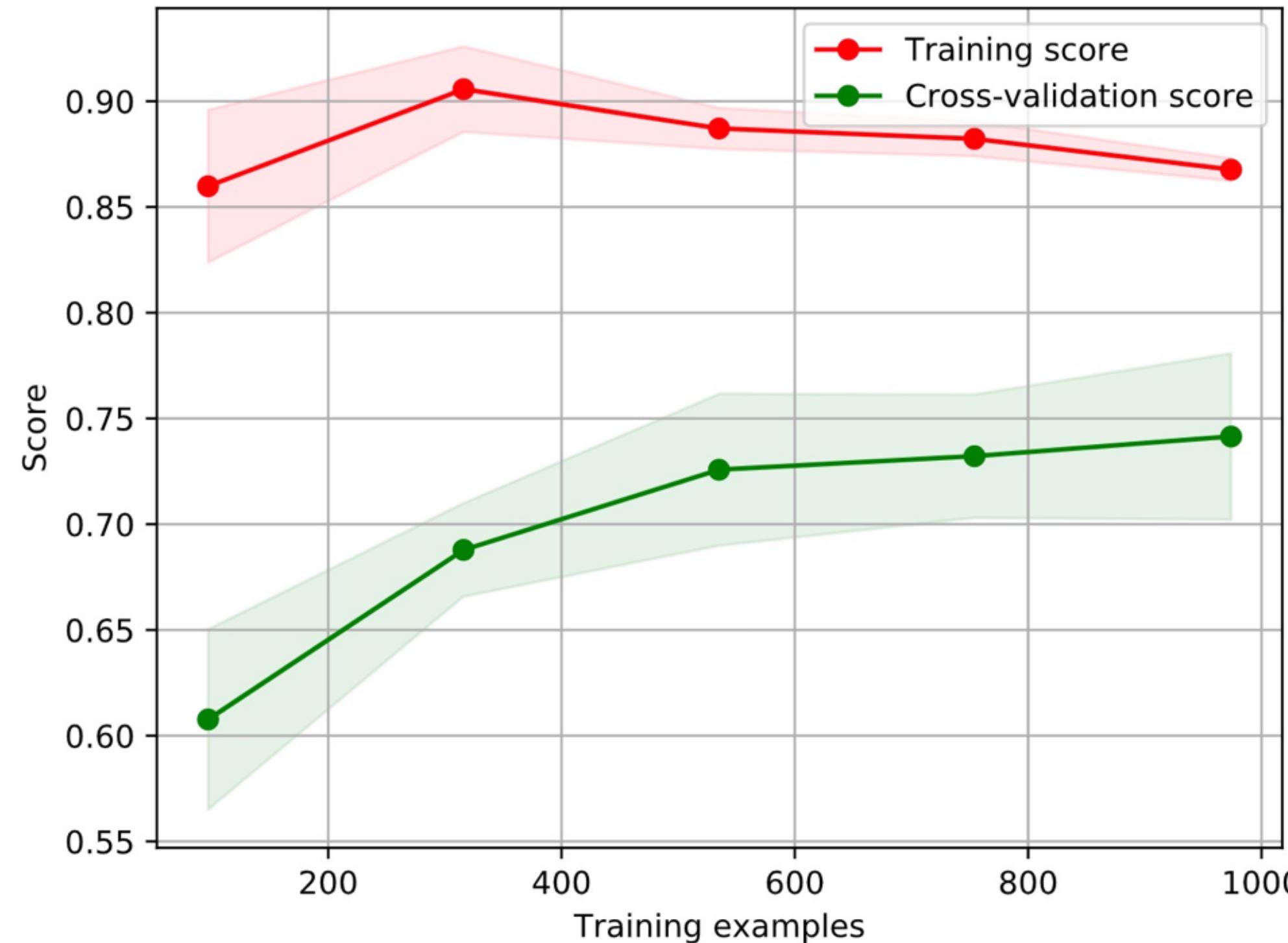


THE LIKABILITY CLASSIFIER: BAGGING LEARNING CURVE



```
parameters {  
    'base': LinearSVC(),  
    'n_estimators': 100  
}
```

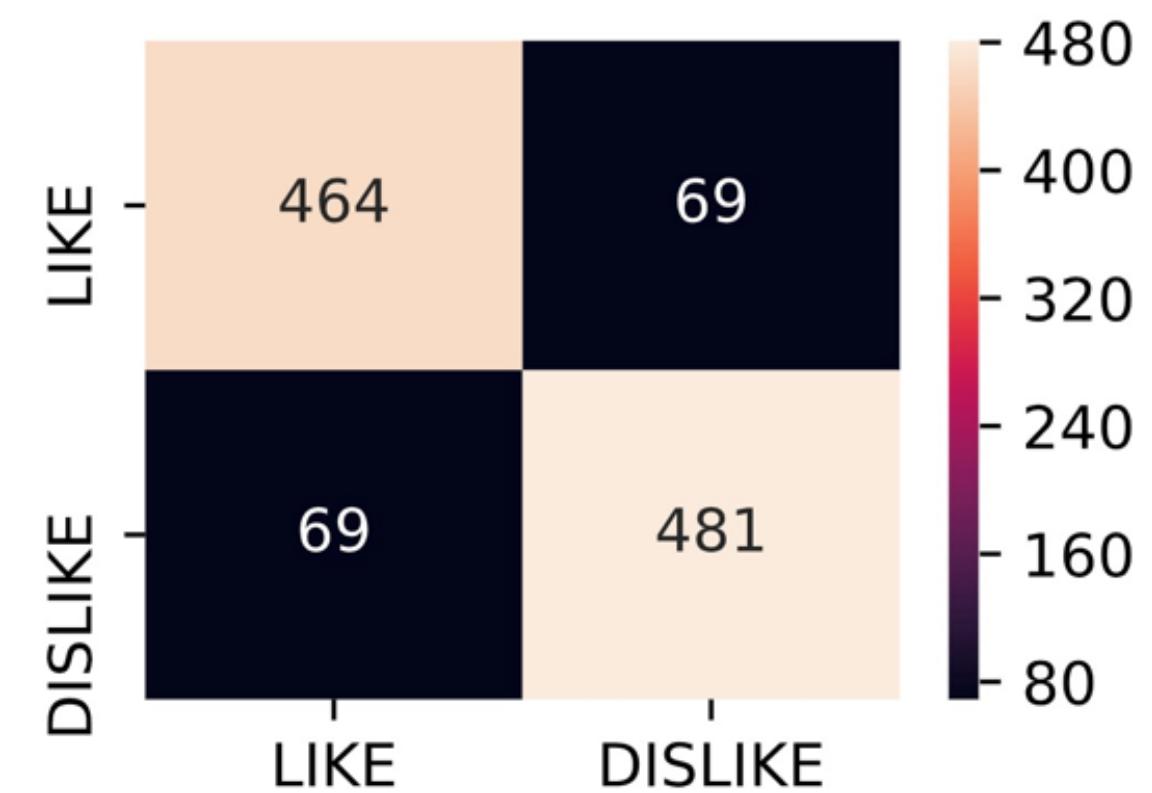
THE LIKABILITY CLASSIFIER: XGB LEARNING CURVE



```
parameters {  
    'n_estimators': 100,  
    'max_depth': 3,  
    'learning_rate': 1  
}
```

THE CATEGORY CLASSIFIER

In the end, no classifier did better than the base **LogisticRegression**, so it was chosen as the classifier that should predict the article likability. Here is shown its **confusion matrix**:



Please notice that this result was achieved under the assumption that the category of the article was correctly predicted !

THE WEB APPLICATION: A SCREENSHOT OF THE NEWSFEED

Personalized feed

Predicted category

Maybe an error?

The screenshot shows a web application interface titled "hopefully-smart news aggregator". At the top, there is a search bar with the placeholder "type a command...". Below the search bar, the title "your newsfeed" is displayed. The news feed consists of several news items and social media posts:

- Entertainment - Fazio:** Che sporri tempo che fa chiude in anticipo, cancellate tre puntate del lunedì (Corriere della Sera)
- Redazione online:** 2019-05-12T19:21:09
L'annuncio in diretta su RaiUno
- facebook:** Post from Lega Salvini about the closure of 23 pages on Facebook.
- Tecnologia - 'Diffondono fake news':** Facebook chiude 23 pagine (metà a favore di M5S e Lega) (Corriere della Sera)
Valentina Santarpia 2019-05-12T18:31:59
Difondevano disinformazione e violavano la policy del social network in tema di autenticità: questi i motivi per cui 23 pagine italiane sono state chiuse. Tra queste, molte pagine non ufficiali a sostegno dei partiti di governo
- Cronaca - A Roma due hamburger e due cappuccini:** 81 euro. Lo scontrino diventa un caso social (Corriere della Sera)
Valeria Costantini 2019-05-12T16:52:01
Ennesimo caso di «turisti spennati» nella Capitale. In un locale a due passi da piazza San Pietro il conto è da ristorante di lusso
- Politica - Salvini:** «Le Europee, un referendum tra la vita e la morte» e Di Maio ricorda: «A Renzi non andò bene» (Corriere della Sera)
Redazione Online 2019-05-12T18:48:08
Il ministro dell'Interno lancia
- Entertainment - Anche i gorilla odiano la pioggia:** la loro reazione al diluvio è «umana» (Corriere della Sera)
Gino Pagliuca

A table at the bottom right lists bank names and their interest rates:

Banca	Tasso nominale	Rata mensile	Tasso effettivo
Credem	1,30%	568,08	1,59%
Widiba	1,55%	581,93	1,63%
Webank.it	1,60%	584,59	1,64%
IBBank	1,55%	581,93	1,67%
Hello Bank!	1,60%	584,59	1,74%
Credit Agricole - Cariparma	1,62%	585,7	1,75%
Sella	1,60%	584,59	1,75%
Banco Desio	1,55%	581,93	1,75%
CheBanca!	1,55%	581,93	1,76%
UbiBanca	1,55%	581,93	1,76%
Banca	1,56%	582,37	1,76%