The Python Way: Predicting Entity Popularity on Twitter

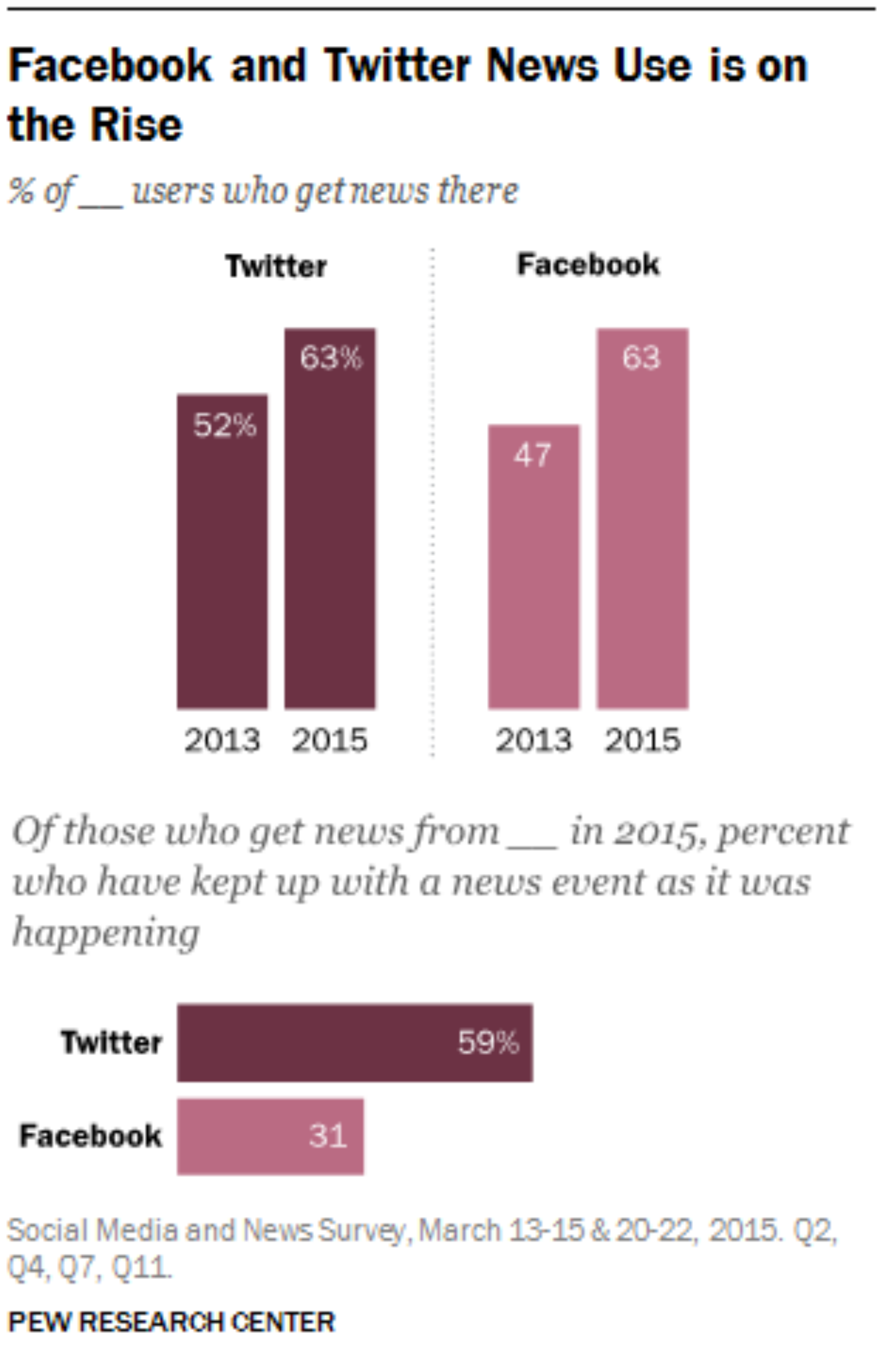
Pedro Saleiro

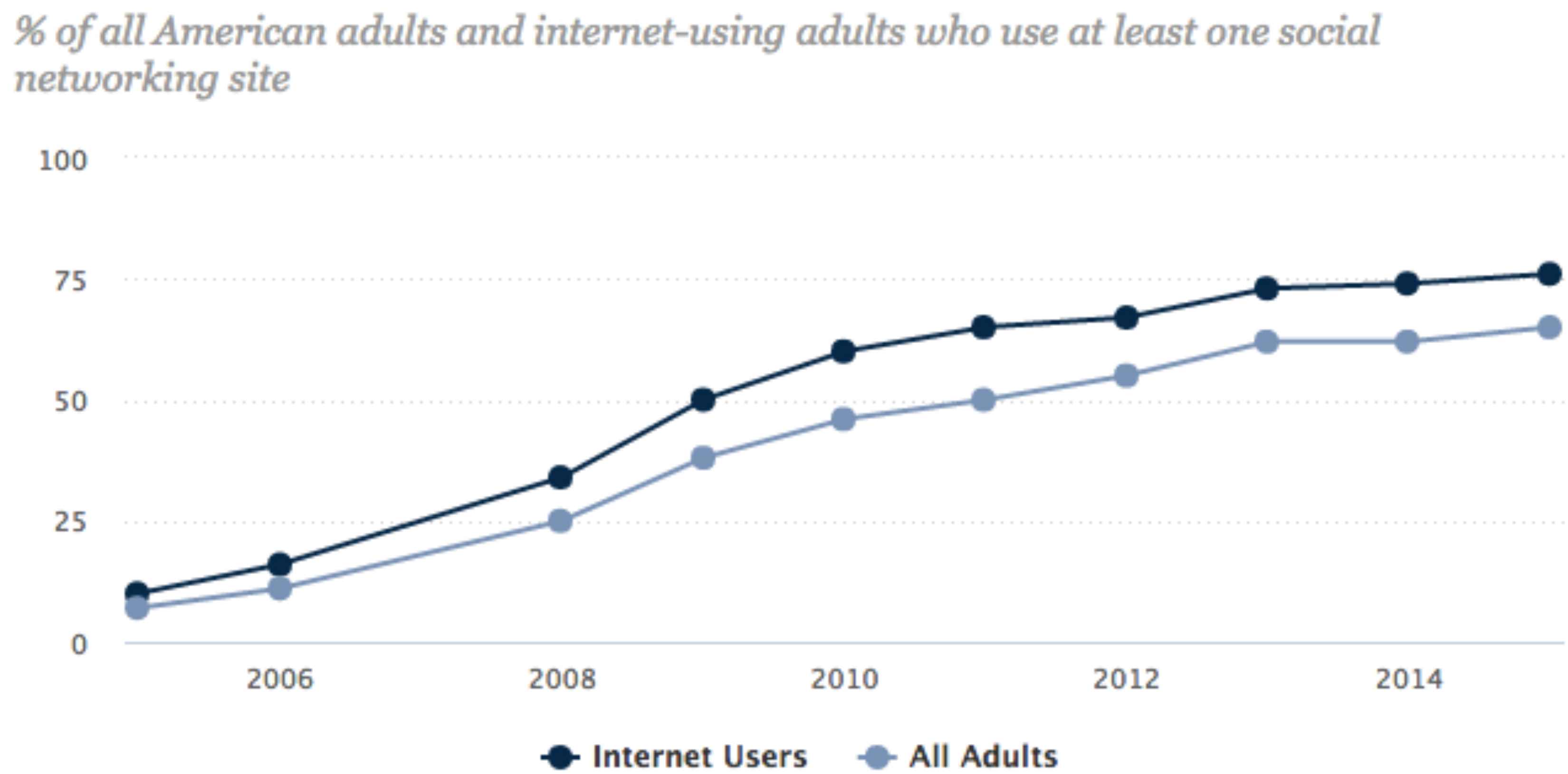
DSPT#2
Porto, 2016

Reputation

[Van Riel et al., 2007] define reputation as "overall assessments of organizations by their stakeholders"

Social Media & Online News





Online Reputation Monitoring

Tracking what is said about a given entity on Social Media

Early ORM systems focused on counting entity mentions on Social Media

Implies collecting, cleaning, filtering, mining, exploring and analysing large streams of unstructured text data

Current Systems focus in NER, NED, Polarity Classification and Visualisation (Social Media Analytics)

POPmine

Framework for ORM

Data

Tweets

100K “portuguese” users panel

24/7 since 2012

228M tweets by today

News

Sapo news crawler

55 online news outlets since 2010

6M news articles

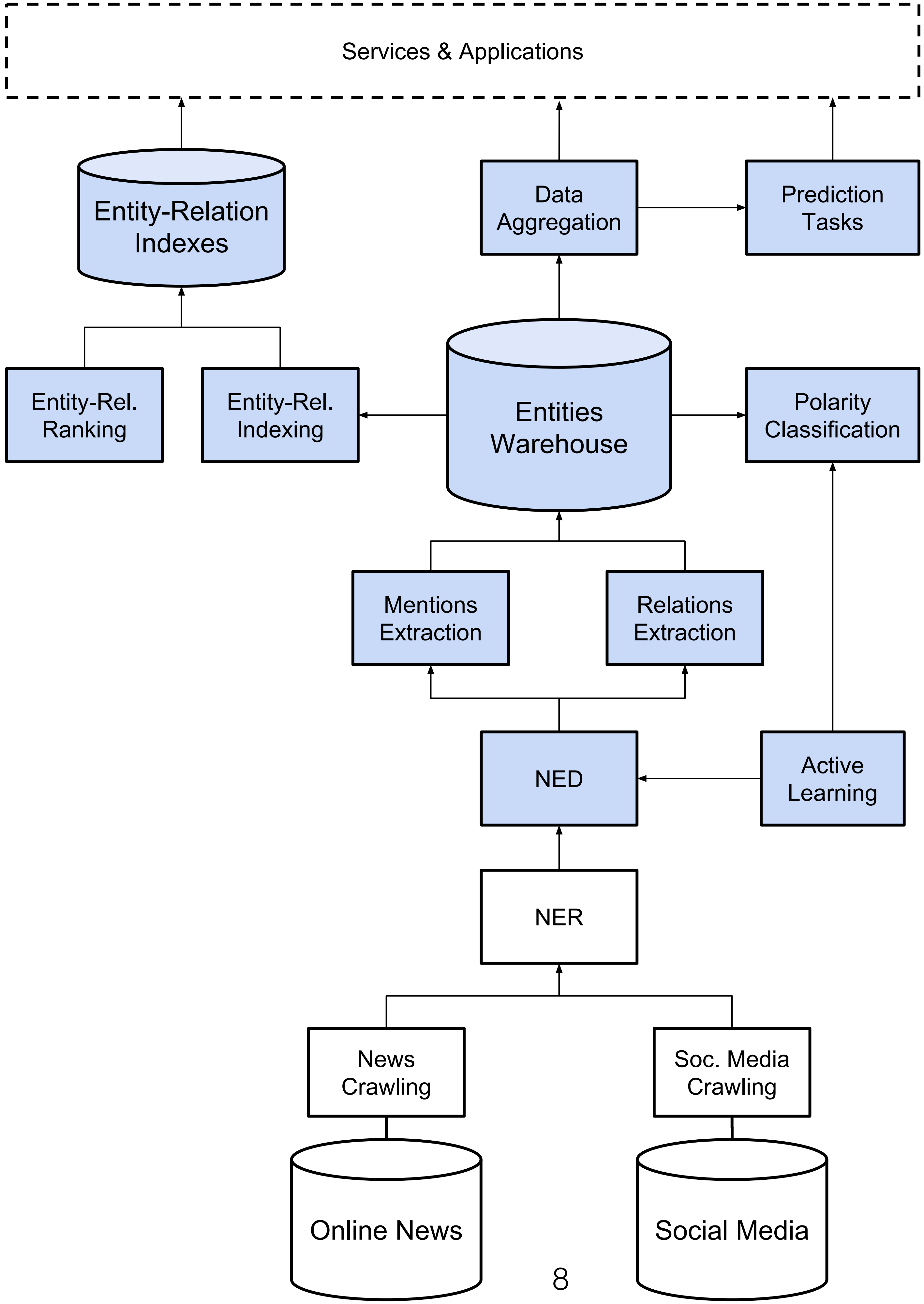
Named Entities

Filters using Natural Language Processing, Information Retrieval and Machine Learning.

“Passos” is Passos Coelho based on remaining words, hashtags, links.

Use of SapoLabs Verbetes (names of personalities and their lexical variations).

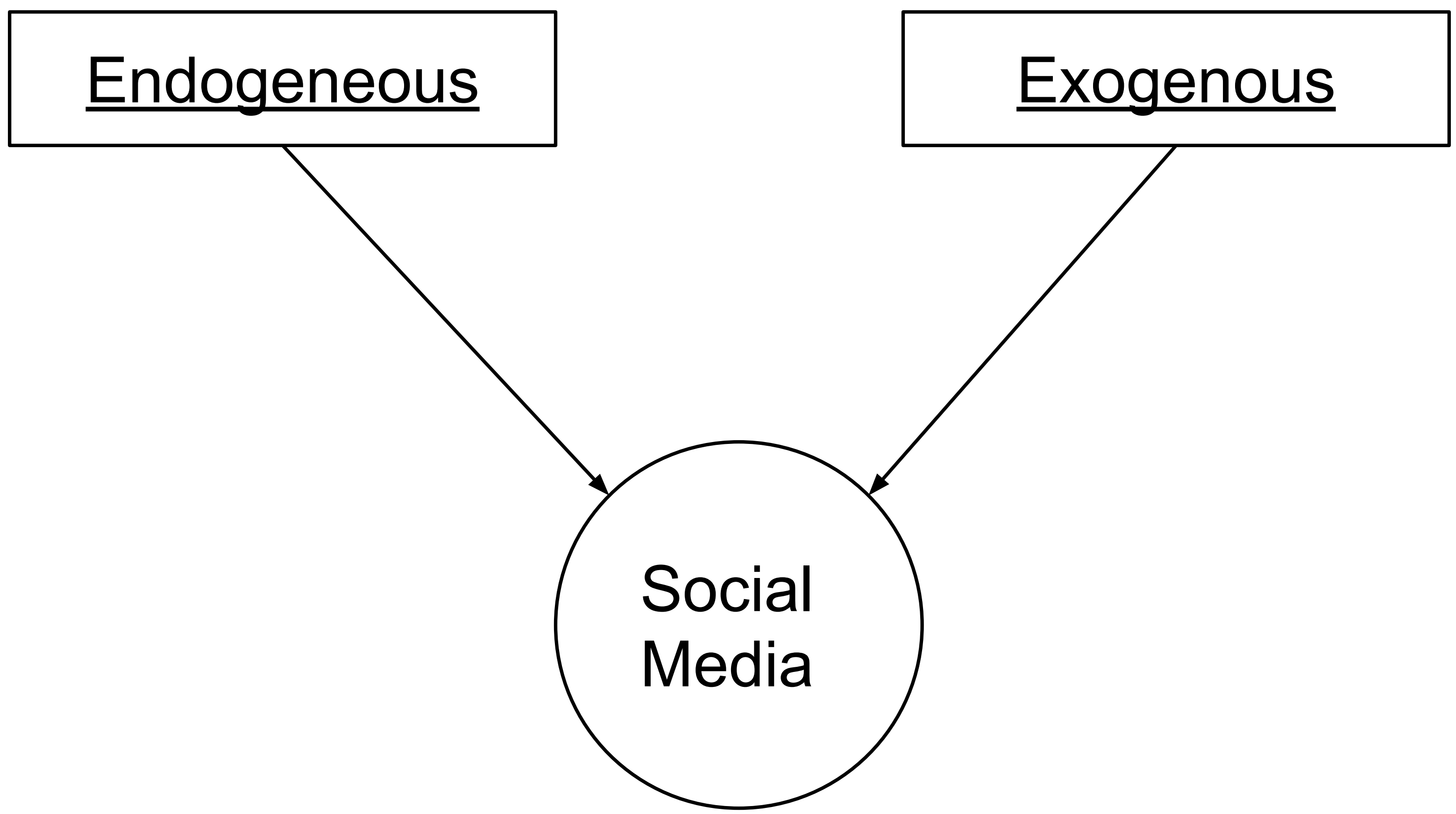
RepLab Filtering competition: 91% accuracy.


Services & Applications
Entity-Relation Indexes
Data Aggregation
Prediction Tasks
Entity-Rel. Ranking
Entity-Rel. Indexing
Entities Warehouse
Polarity Classification
Mentions Extraction
Relations Extraction
NED
Active Learning
NER
News Crawling
Soc. Media Crawling
Online News
Social Media

Prediction Task

Predict entity popularity on Twitter?

Social Media Attention



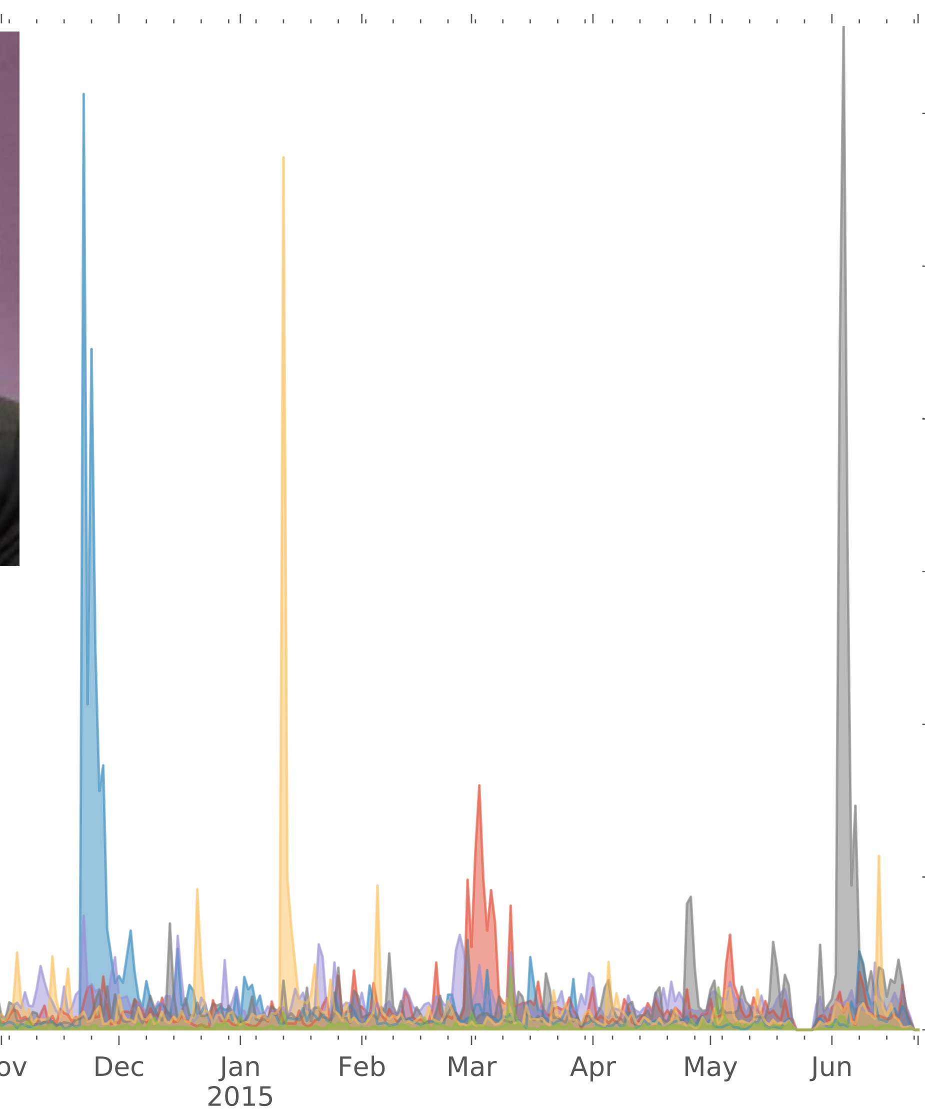
Social Media Attention

Endogenous: Followers network, Influence, Hashtags,

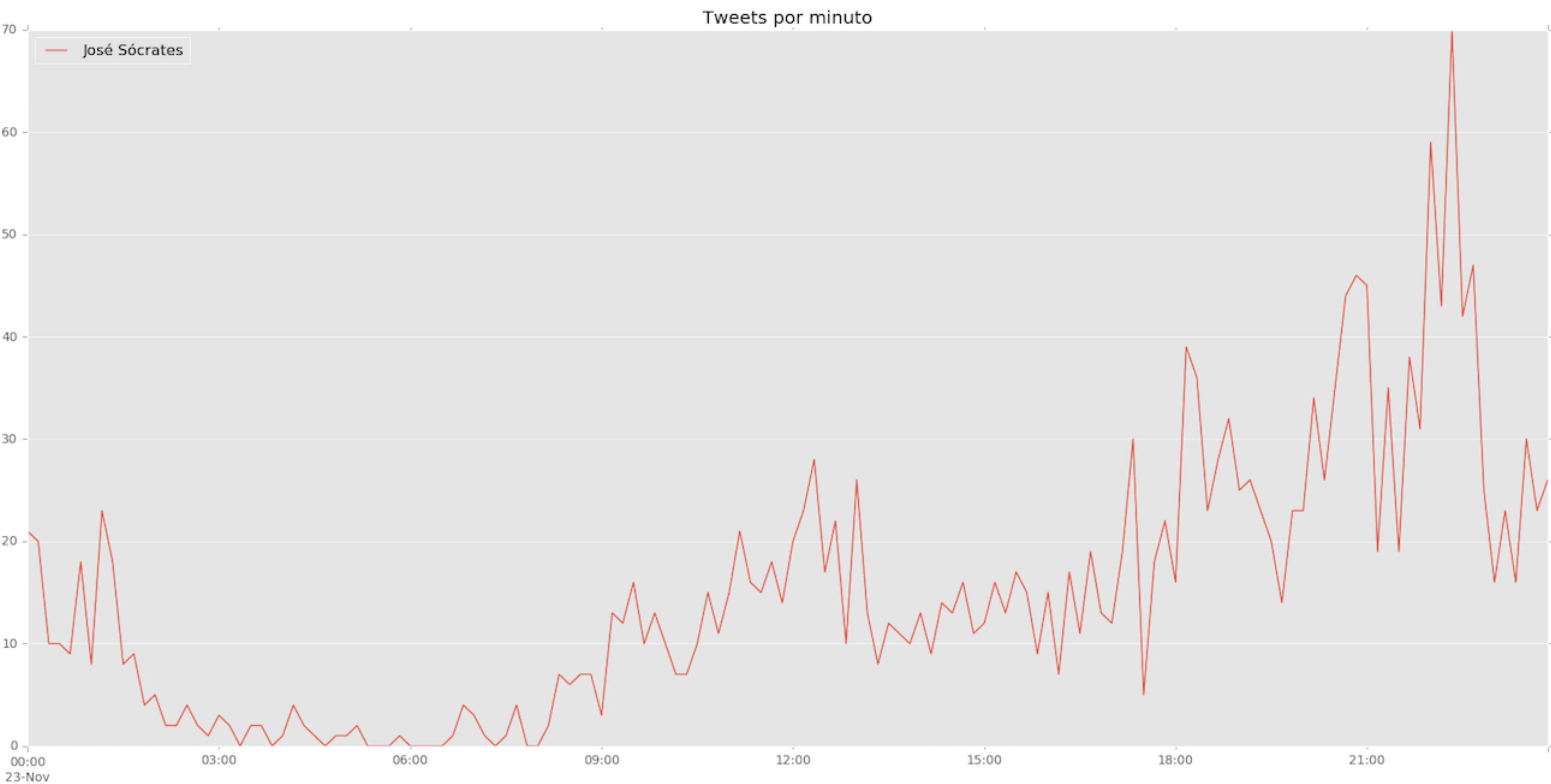
Exogenous: Online Media, TV, Traditional Media, Friends, Family, Personality, Macroeconomy,…

Goal

Predict if an entity (e.g. politician) will be *frequently mentioned on Twitter* in the hours *following appearing in the news*

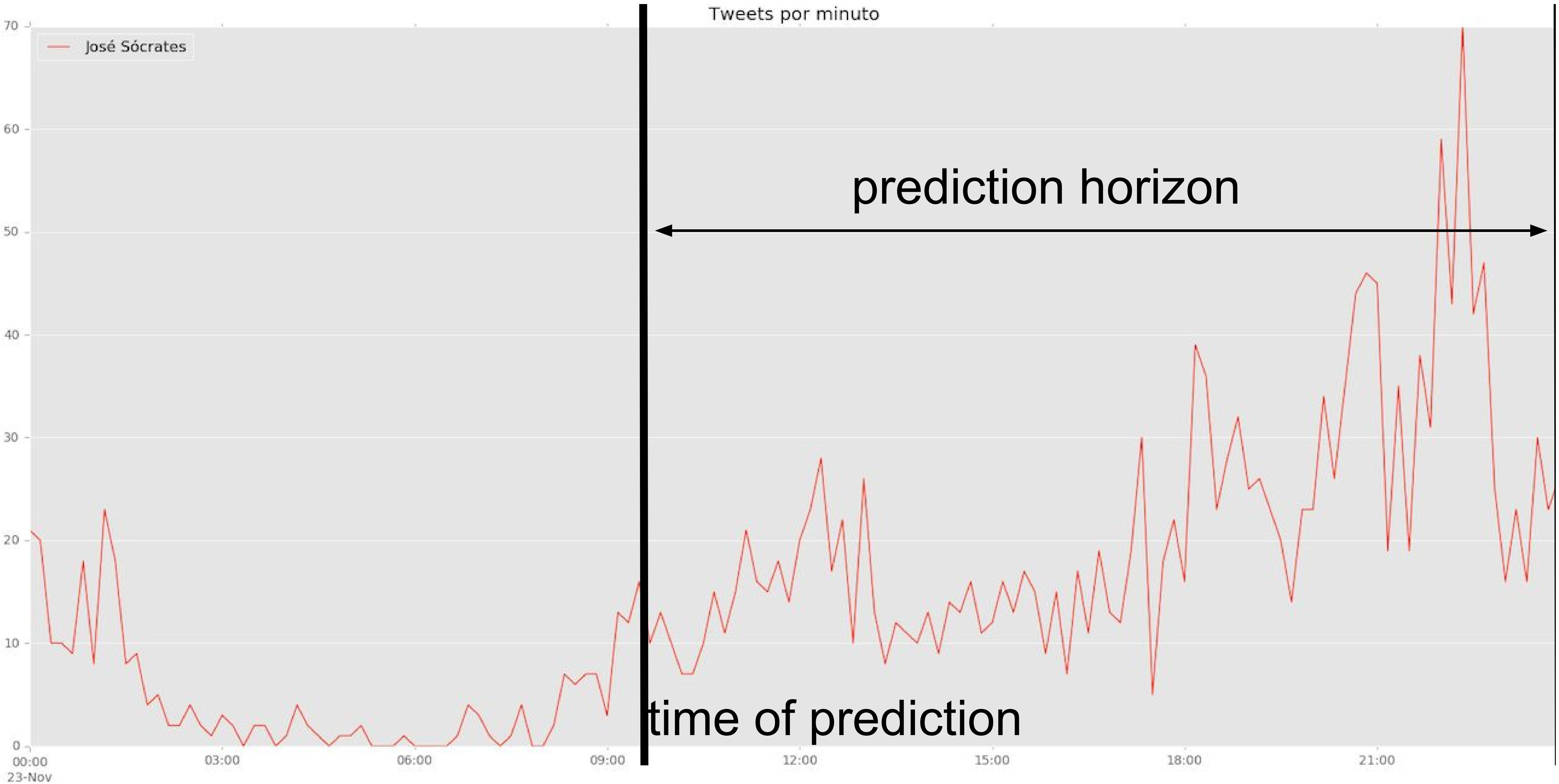


Example



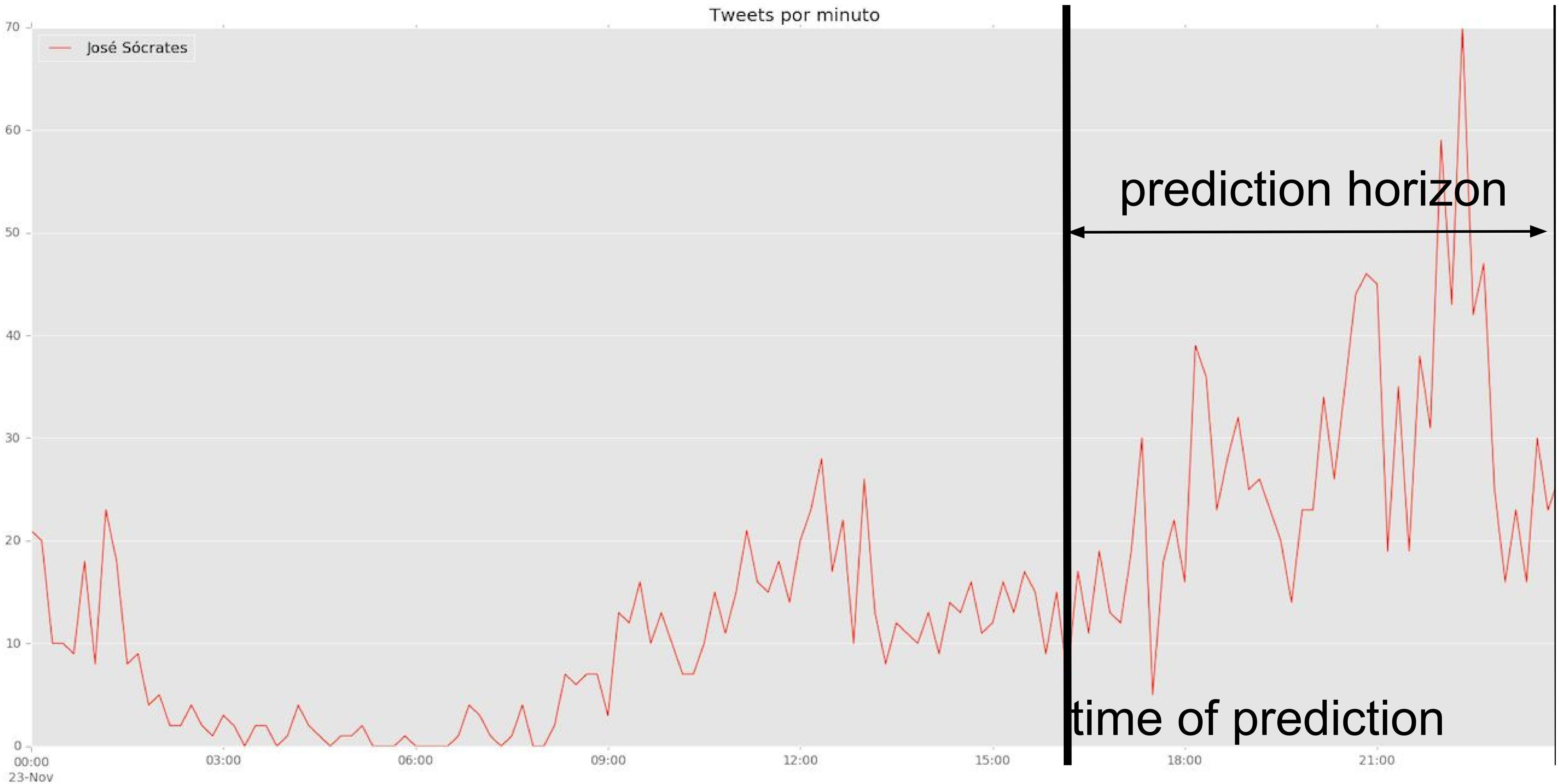
Tweets por minuto
José Sócrates
70
60
50
40
30
20
10
0
00:00
23-Nov
03:00
06:00
09:00
12:00
15:00
18:00
21:00

Example



Tweets por minuto
José Sócrates
prediction horizon
time of prediction
70
60
50
40
30
20
10
0
00:00
23-Nov
03:00
06:00
09:00
12:00
15:00
18:00
21:00

Example



Tweets por minuto
José Sócrates
70
60
50
40
30
20
10
0
00:00
23-Nov
03:00
06:00
09:00
12:00
15:00
18:00
21:00
prediction horizon
time of prediction

Formalisation

Given a set of entities $E = \{e_1, e_2, ..., e_i, ...\}$,

a daily stream of social media messages $S = \{s_1, s_2, ..., s_i, ...\}$,

a daily stream of online news articles $N = \{n_1, n_2, ..., n_i, ...\}$,

a discrete function $f_m(e_i, S)$ representing mentions of an entity e_i on the social media stream S,

a daily time frame $T = [t_p, t_{p+h}]$, where the time t_p is the time of prediction and t_{p+h} is the prediction horizon time.

We want to learn a target popularity function

$$f_p(e_i, N, T) = \sum_{t=t_p}^{t=t_{p+h}} f_m(e_i, S)$$

Approach

Supervised learning approach

News features:

- *signal*
- *textual*
- *semantic*
- *sentiment*

Multiple prediction horizons

- *e.g. impact until 24:00 of news published at 8am*

Features - Signal

- total mentions of entity in the news
    - between midnight and time of prediction
    - same for previous day (lagged)
- total mentions of entity in news titles
- average news length
- number of different news outlets mentioning entity
- weekday/weekend

Features - Textual

Create a meta-document of news titles mentioning entity between midnight and time of prediction

Everyday

Calculate TF-IDF for each day (compare with others)

Create a topic model using Latent Dirichlet Allocation

assign topic probability for each day

Features - Sentiment

Use a sentiment lexicon

Count occurrences in news titles mentioning entity

Count positives, negatives and neutrals

Calculate aggregate functions (e.g. pos/neg)

Calculate a TF-IDF score of adjectives in the news

Features - Semantic

Take advantage of journalists tags

Semantic categories (e.g. Internal Politics)

Calculate TF-IDF of semantic tags

Create a BOW representation of named entities

Calculate a TF-IDF of named entities

Prediction

Regression or classification?

Implementation

The Python Way

db
entity
horizon
train period
test period
features list

model
predictions
evaluation

Tweets & News DB

```
db.tweets.find_one()
```


```
{u'_id': u'144424904090062849',
 u'datasource_s': u'mysql',
 u'description': u'Jornalista redactor da RTP',
 u'followers_count': 89,
 u'friends_count': 269,
 u'language_s': u'pt',
 u'location': u'',
 u'name': u'Carlos Santos Neves',
 u'profile_image_url': u'http://a0.twimg.com/profile_images/1206504715/IMG_0639_normal.JPG',
 u'retweet_count': 0,
 u'screen_name': u'csantosneves',
 u'source': u'web',
 u'status_in_reply_to_status_id': u'0',
 u'status_in_reply_to_user_id': u'0',
 u'text': u'A\xed est\xe1 Paulo Portas: "A Europa, em todo o caso, n\xe3o se faz a dois, faz-se a 27" - http://t.c
o/aa4XZYGk',
 u'time_zone': u'Lisbon',
 u'tokenized_text': u'A\xed est\xe1 Paulo Portas : " A Europa , em todo o caso , n\xe3o se faz a dois , faz-se a 27
" - http://t.co/aa4XZYGk',
 u'tweet_date': datetime.datetime(2011, 12, 7, 14, 35, 55),
 u'urls': [u'http://t.co/aa4XZYGk'],
 u'user_created_at': datetime.datetime(2010, 10, 26, 9, 54, 22),
 u'user_id': u'207933284',
 u'version': 10}
```

Tweets & News DB

```
db.news.find_one()
```


```
{u'_id': u'6066313',
 u'content': u"O avan\xe7ado Karim Benzema mostrou pontaria afinada no triunfo sobre os LA Galaxy (3-1), terminando
o encontro com dois golos na folha de marcadores. A partida est\xe1 inserida na Guiness Cup, competi\xe7\xe3o amig
\xe1vel realizada na digress\xe3o dos merengues pelos Estados Unidos.\nO \xfanico tento da primeira parte foi apont
ado por Angel Di Maria. O extremo argentino n\xe3o se fez rogado e inaugurou a contenda, logo aos 15 minutos.\nA se
gunda metade tamb\xe9m come\xe7ou de fei\xe7\xe3o para os merengues, que aumentaram vantagens por interm\xe9dio de
Karim Benzema, aos 51'. Jose Vilarreal ainda reduziu para a antiga forma\xe7\xe3o de David Beckham (63'), mas o ava
n\xe7ado franc\xeas matou definitivamente o encontro, com um bis (75').\nNa mesma competi\xe7\xe3o est\xe1 inserido
o Chelsea de Jos\xe9 Mourinho que, esta madrugada, com golos de Oscar e Eden Hazard derrotou o Inter de Mil\xe3o. O
s italianos jogaram com menos um elemento desde a expuls\xe3o de Campagnaro, aos 58'.",
 u'link': u'http://www.futebol365.pt/noticias/artigo.asp?id=90615&utm_source=rss&utm_medium=feed&utm_campaign=notic
ias_xml',
 u'numComments': u'0',
 u'occurrences': [u'Jos\xe9 Mourinho'],
 u'pubdate': datetime.datetime(2013, 8, 2, 9, 30, 37),
 u'source': u'Futebol 365',
 u'tags': u'',
 u'title': u'Jogos Amig\xe1veis: Benzema em destaque na vit\xf3ria sobre os LA Galaxy'}
```

Tweets & News DB

```
db.tweets_mentions.find_one({'target_name':'Pedro Passos Coelho'})
```


```
{u'_id': u'290892810243162112 50b344a0e9e57312c88b8bfc',
 u'mention_confidence': 1,
 u'mention_size': 3,
 u'polarity': {u'PopstarOpinionizer_2': 0},
 u'random': 0.5188359553449241,
 u'target_id': ObjectId('50b344a0e9e57312c88b8bfc'),
 u'target_mention': u'Pedro Passos Coelho',
 u'target_name': u'Pedro Passos Coelho',
 u'tweet_date': datetime.datetime(2013, 1, 14, 18, 47, 5),
 u'tweet_id': u'290892810243162112',
 u'tweet_user': u'18186509'}
```


```
db.news_mentions.find_one({'target_name':'Marcelo Rebelo de Sousa'})
```


```
{u'_id': u'3746192 50ec6ca8e9e573741a23a73c',
 u'news_date': datetime.datetime(2011, 6, 21, 10, 29),
 u'news_id': u'3746192',
 u'news_link': u'http://aeiou.expresso.pt/assuncao-esteves-candidata-a-liderar-a-ar=f656920',
 u'news_source': u'Expresso',
 u'target_id': ObjectId('50ec6ca8e9e573741a23a73c'),
 u'target_name': u'Marcelo Rebelo de Sousa'}
```

Tweets & News DB

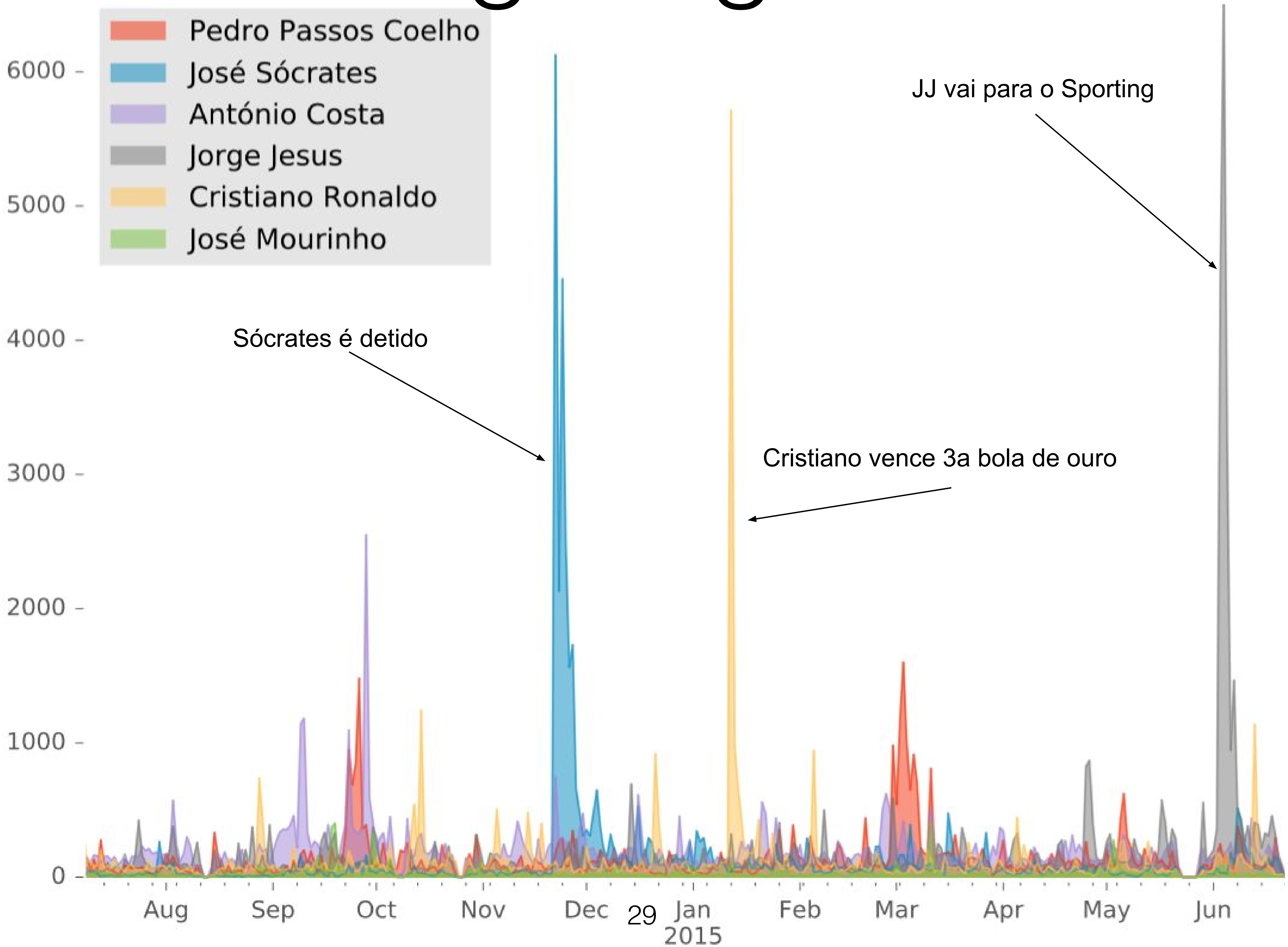
```
db.tweets_hourly_sentiment.find_one({'_id.target_name':'Jorge Jesus'})
```

```
{u'_id': {u'hour': u'2011-06-21T00', u'target_name': u'Jorge Jesus'},
 u'value': {u'negative_mentions': 3.0,
  u'neutral_mentions': 0.0,
  u'positive_mentions': 0.0,
  u'total_mentions': 3.0}}
```

```
db.news_hourly_buzz.find_one({'_id.target_name':'Luís Filipe Vieira'})
```

```
{u'_id': {u'hour': u'2011-06-22T06', u'target_name': u'Lu\xeds Filipe Vieira'},
 u'value': {u'total_mentions': 1.0}}
```

Generating Target Variable



Generating Target Variable

```
def getSeries(db, entity, start_date, end_date):
    rng_date = pandas.date_range(start_date, end_date, freq='h')
    tweets_hour = {}
    for each in db.tweets_hourly_sentiment.find({'_id.target_name': entity}):
        tweets_hour[each['_id']['hour']] = each['value']['total_mentions']
    ts = pandas.Series(tweets_hour)
    ts = ts.reindex(rng_date)
    ts.fillna(ts.median(),inplace=True) 
    # or we can interpolate: ts = ts.interpolate()
    return ts


def reSample(ts, start_horizon, end_horizon)
    ts_hour = ts.index.hour
    selector = ((ts_hour >= start_horizon) & (ts_hour <= end_horizon))
    series = ts[selector].resample('D', how = 'sum')
    return series
```

Extracting Features

```
    dataset = [{'date': ..., 'titles': ..., ...},
                {'date': ..., 'titles': ..., ...},
                ... ... ...
            ]
    dataset[date]["semantic_entities_num"] = len(set(entities.split()))
```

Extracting Features

```python
from sklearn.decomposition import TruncatedSVD
from sklearn.feature_extraction.text import TfidfVectorizer


def getTFIDF(training_titles):
    #training_titles = [list of concatenated titles per day]
    vec_tfidf = TfidfVectorizer(min_df=5, max_df=0.9, ngram_range=(
        1, 2), smooth_idf=True, sublinear_tf=True, use_idf=True, max_features=10000)
    tfidf_matrix = vec_tfidf.fit_transform(training_titles)
    lsa_vec = TruncatedSVD(n_components=10, random_state=0)
    lsa_matrix = lsa_vec.fit_transform(tfidf_matrix)
    return vec_tfidf, tfidf_matrix, lsa_vec, lsa_matrix
```

```python
    # TF_IDF TITLES
    if 'titles_tfidf' in features:
        X_test_titles_tfidf = vec_tfidf.transform(each[date]['titles'])
        X_test_titles = lsa_vec.transform(X_test_titles_tfidf)
```

Extracting Features

```
import lda

def getLDA(training_titles):
    cvec_titles = CountVectorizer(ngram_range=(1, 1), binary = True, min_df = 2, max_df=0.9)
    cvec_matrix = cvec_titles.fit_transform(training_data)
    lda_model = lda.LDA(n_topics=10, n_iter=500, random_state=1)
    lda_model.fit(cvec_matrix)
    topic_word = lda_model.topic_word_  # model.components_ also works
    doc_topic = lda_model.doc_topic_
    return doc_topic, cvec_titles, lda_model
```

```
    if 'titles_lda' in features:
        cvec_matrix = cvec_titles.transform(each[date]['titles'])
        doc_topic = lda_model.transform(cvec_matrix)
```

Features

Number	Feature	Description	Type
Signal			
1	*news*	number of news mentions of e_i in $[0, t_p]$ in d_i	*Int*
2	*news* d_{i-1}	number of news mentions of e_i in $[0, t_p]$ in d_{i-1}	*Int*
3	*news total* d_{i-1}	number of news mentions of e_i in $[0, 24[$ in d_{i-1}	*Int*
4	*news titles*	number of title mentions in news of e_i in $[0, t_p]$ in d_i	*Int*
5	*avg content*	average content length of news of e_i in $[0, t_p]$ in d_i	*Float*
6	*sources*	number of different news sources of e_i in $[0, t_p]$ in d_i	*Int*
7	*weekday*	day of week	*Categ*
8	*is weekend*	true if weekend, false otherwise	*Bool*

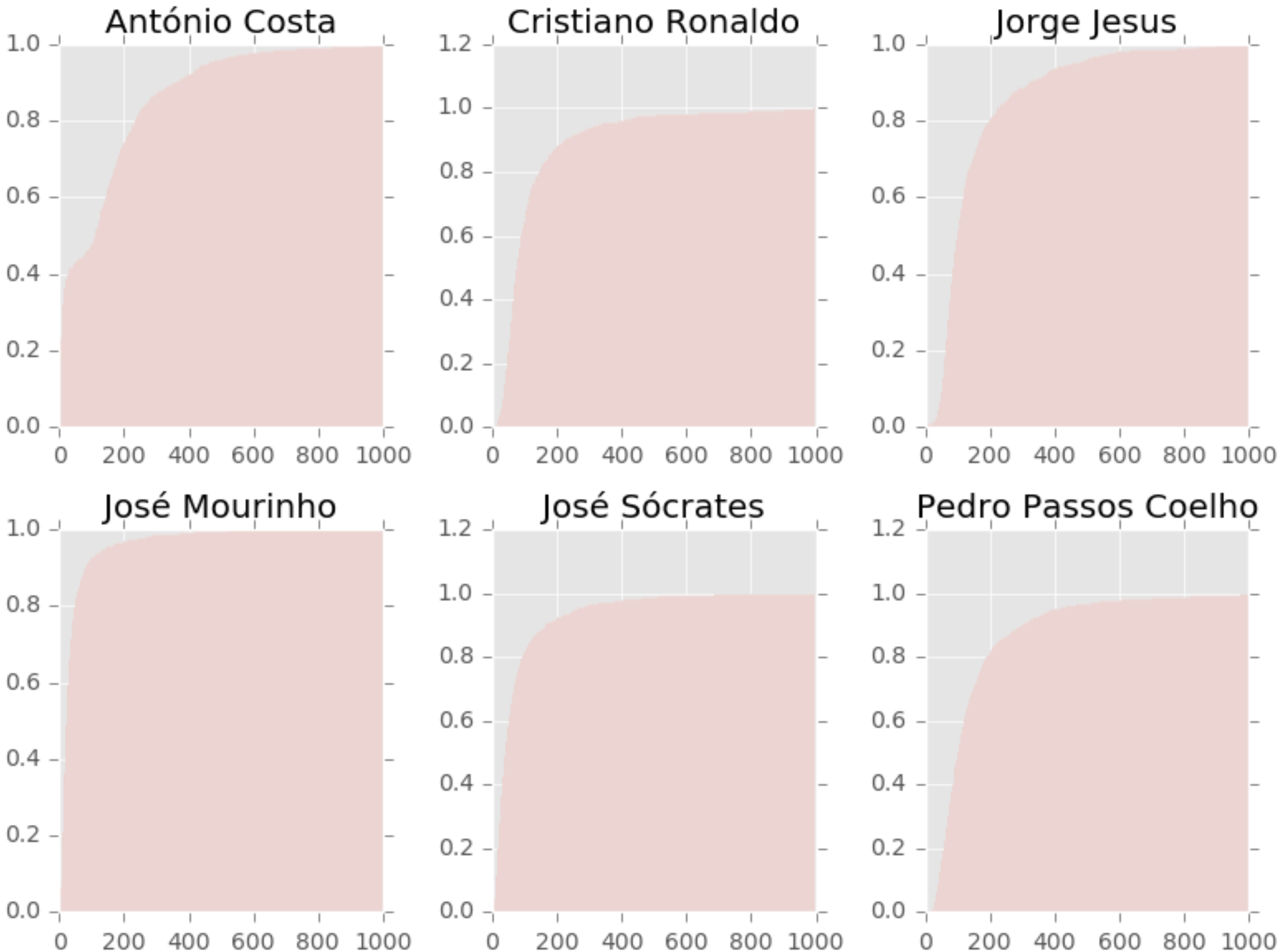
Features

Number	Feature	Description	Type
Textual			
9–18	*tfidf titles*	TF-IDF of news titles $[0, t_p]$ in d_i	*Float*
19–28	*LDA titles*	LDA-10 of news titles $[0, t_p]$ in d_i	*Float*
Sentiment			
29	*pos*	number of positive words in news titles $[0, t_p]$ in d_i	*Int*
30	*neg*	number of negative words in news titles $[0, t_p]$ in d_i	*Int*
31	*neu*	number of neutral words in news titles $[0, t_p]$ in d_i	*Int*
32	*ratio*	$positive/negative$	*Float*
33	*diff*	$positive - negative$	*Int*
34	*subjectivity*	$(positive + negative + neutral)/\sum words$	*Float*
35–44	*tfidf subj*	TF-IDF of subjective words (pos, neg and neu)	*Float*
Semantic			
45	*entities*	number of entities in news $[0, t_p]$ in d_i	*Int*
46	*tags*	number of tags in news $[0, t_p]$ in d_i	*Int*
47–56	*tfidf entities*	TF-IDF of entities in news $[0, t_p]$ in d_i	*Float*
57–66	*tfidf tags*	TF-IDF of news tags $[0, t_p]$ in d_i	*Float*

Prediction

$$\hat{f}_p = \begin{cases} 0(low), & \text{if } P(f_p(e_i, N, T) \leq \delta) = k \\ 1(high), & \text{if } P(f_p(e_i, N, T) > \delta) = 1 - k \end{cases}$$

δ is the inverse of cumulative distribution function at k of $f_p(e_i, N, T)$


António Costa
Cristiano Ronaldo
Jorge Jesus
José Mourinho
José Sócrates
Pedro Passos Coelho
0.0
0.2
0.4
0.6
0.8
1.0
1.2
0
200
400
600
800
1000

Combining stuff

```
    X_train = np.hstack([X_train_signal, X_train_textual, X_train_sentiment, X_train_semantic])


    y_train = np.array(training_tweets[:,vec_feat.index("tweets_series")])
    threshold_65 = np.percentile(y_train, 65.0)
    y_train[y_train <= threshold_65] = 0
    y_train[y_train > threshold_65] = 1
```

Train - Predict

```
def fitModel(X_train, y_train, classifier):
    classifier.fit(X_train, y_train)
    model = pickle.dumps(classifier)
    return classifier

def predictModel(classifier, X_test):
    #classifier = pickle.loads(model_path)
    y_predicted = classifier.predict(X_test)
    return y_predicted
```

Predictors

`linear_model.ARDRegression` ([n_iter, tol, ...])	`ensemble.AdaBoostClassifier` ([...])
`linear_model.BayesianRidge` ([n_iter, tol, ...])	`ensemble.AdaBoostRegressor` ([base_estimator, ...])
`linear_model.ElasticNet` ([alpha, l1_ratio, ...])	`ensemble.BaggingClassifier` ([base_estimator, ...])
`linear_model.ElasticNetCV` ([l1_ratio, eps, ...])	`ensemble.BaggingRegressor` ([base_estimator, ...])
`linear_model.HuberRegressor` ([epsilon, ...])	`ensemble.ExtraTreesClassifier` ([...])
`linear_model.Lars` ([fit_intercept, verbose, ...])	`ensemble.ExtraTreesRegressor` ([n_estimators, ...])
`linear_model.LarsCV` ([fit_intercept, ...])	`ensemble.GradientBoostingClassifier` ([loss, ...])
`linear_model.Lasso` ([alpha, fit_intercept, ...])	`ensemble.GradientBoostingRegressor` ([loss, ...])
`linear_model.LassoCV` ([eps, n_alphas, ...])	`ensemble.IsolationForest` ([n_estimators, ...])
`linear_model.LassoLars` ([alpha, ...])	`ensemble.RandomForestClassifier` ([...])
`linear_model.LassoLarsCV` ([fit_intercept, ...])	`ensemble.RandomTreesEmbedding` ([...])
`linear_model.LassoLarsIC` ([criterion, ...])	`ensemble.RandomForestRegressor` ([...])
`linear_model.LinearRegression` ([...])	`ensemble.VotingClassifier` (estimators[, ...])
`linear_model.LogisticRegression` ([penalty, ...])	`svm.SVC` ([C, kernel, degree, gamma, coef0, .
`linear_model.LogisticRegressionCV` ([Cs, ...])	`svm.LinearSVC` ([penalty, loss, dual, tol, C, ...
`linear_model.MultiTaskLasso` ([alpha, ...])	`svm.NuSVC` ([nu, kernel, degree, gamma, ...])
	`svm.SVR` ([kernel, degree, gamma, coef0, tol,
	`svm.LinearSVR` ([epsilon, tol, C, loss, ...])
	`svm.NuSVR` ([nu, C, kernel, degree, gamma, ..
	`svm.OneClassSVM` ([kernel, degree, gamma, .
	`svm.l1_min_c` (X, y[, loss, fit_intercept, ...])

Experimental Setup

Entity-Specific Models

2 years training set (+- 720 examples)

Monthly sliding window

Iteration 1

Training				Test	
Jan 2013	Feb 2013	...	Dec 2014	Jan 2015	Feb 2015

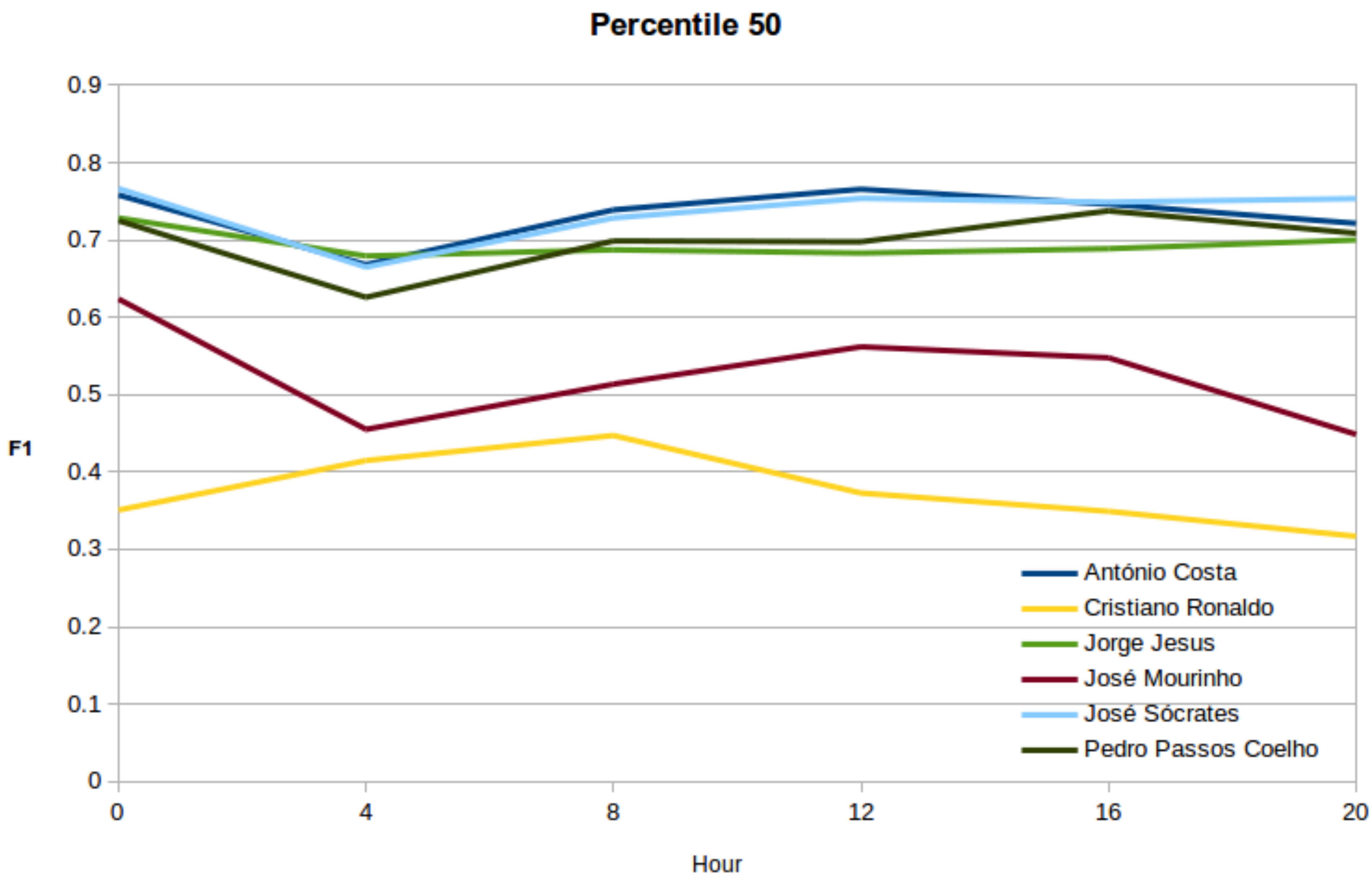
Iteration 2

	Training				Test
Jan 2013	Feb 2013	Mar 2013	...	Jan 2015	Feb 2015

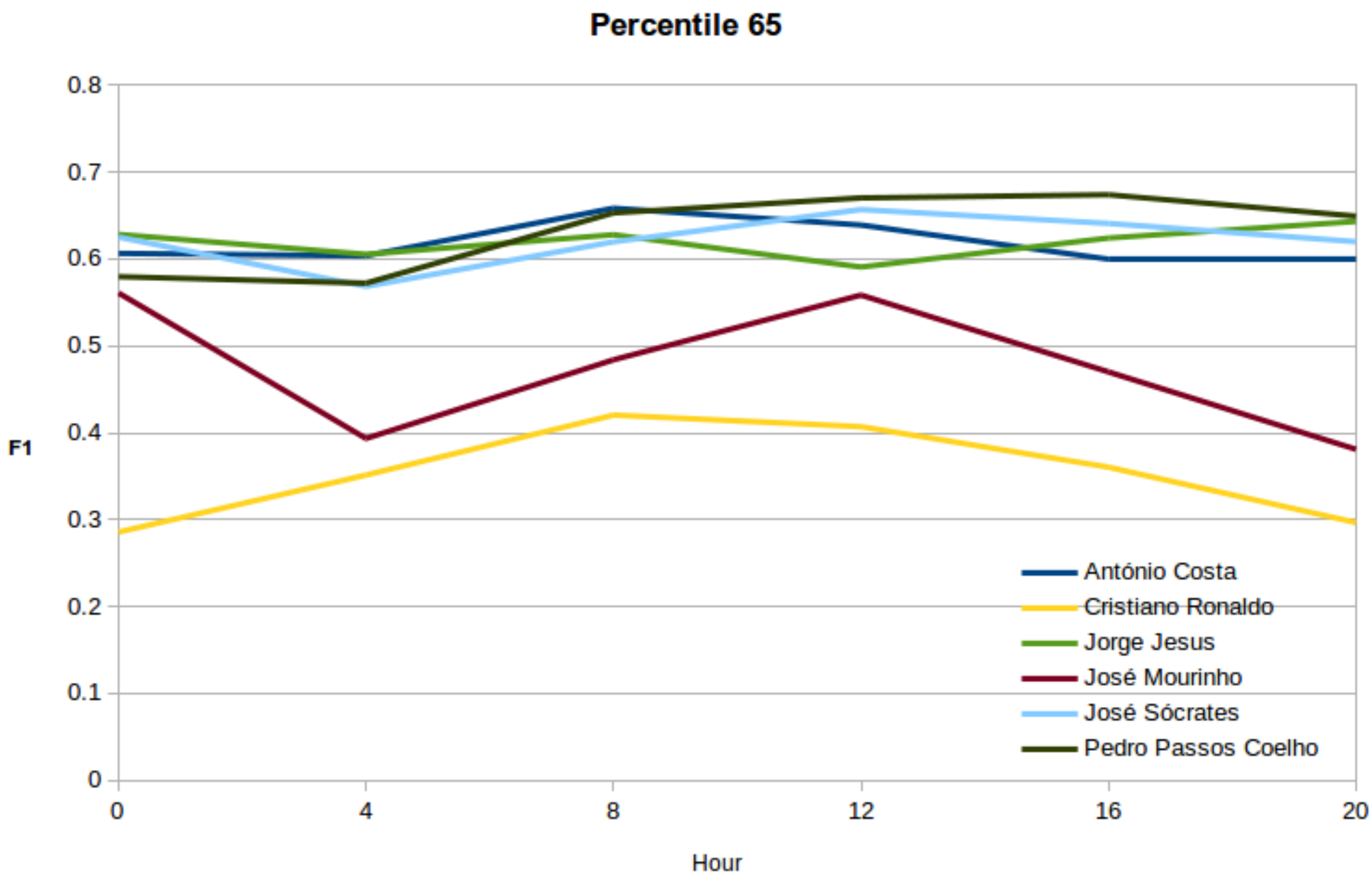
Evaluation

Scoring	Function
Classification	
'accuracy'	`metrics.accuracy_score`
'average_precision'	`metrics.average_precision_score`
'f1'	`metrics.f1_score`
'f1_micro'	`metrics.f1_score`
'f1_macro'	`metrics.f1_score`
'f1_weighted'	`metrics.f1_score`
'f1_samples'	`metrics.f1_score`
'neg_log_loss'	`metrics.log_loss`
'precision' etc.	`metrics.precision_score`
'recall' etc.	`metrics.recall_score`
'roc_auc'	`metrics.roc_auc_score`
Clustering	
'adjusted_rand_score'	`metrics.adjusted_rand_score`
Regression	
'neg_mean_absolute_error'	`metrics.mean_absolute_error`
'neg_mean_squared_error'	`metrics.mean_squared_error`
'neg_median_absolute_error'	`metrics.median_absolute_error`
'r2'	`metrics.r2_score`

Results

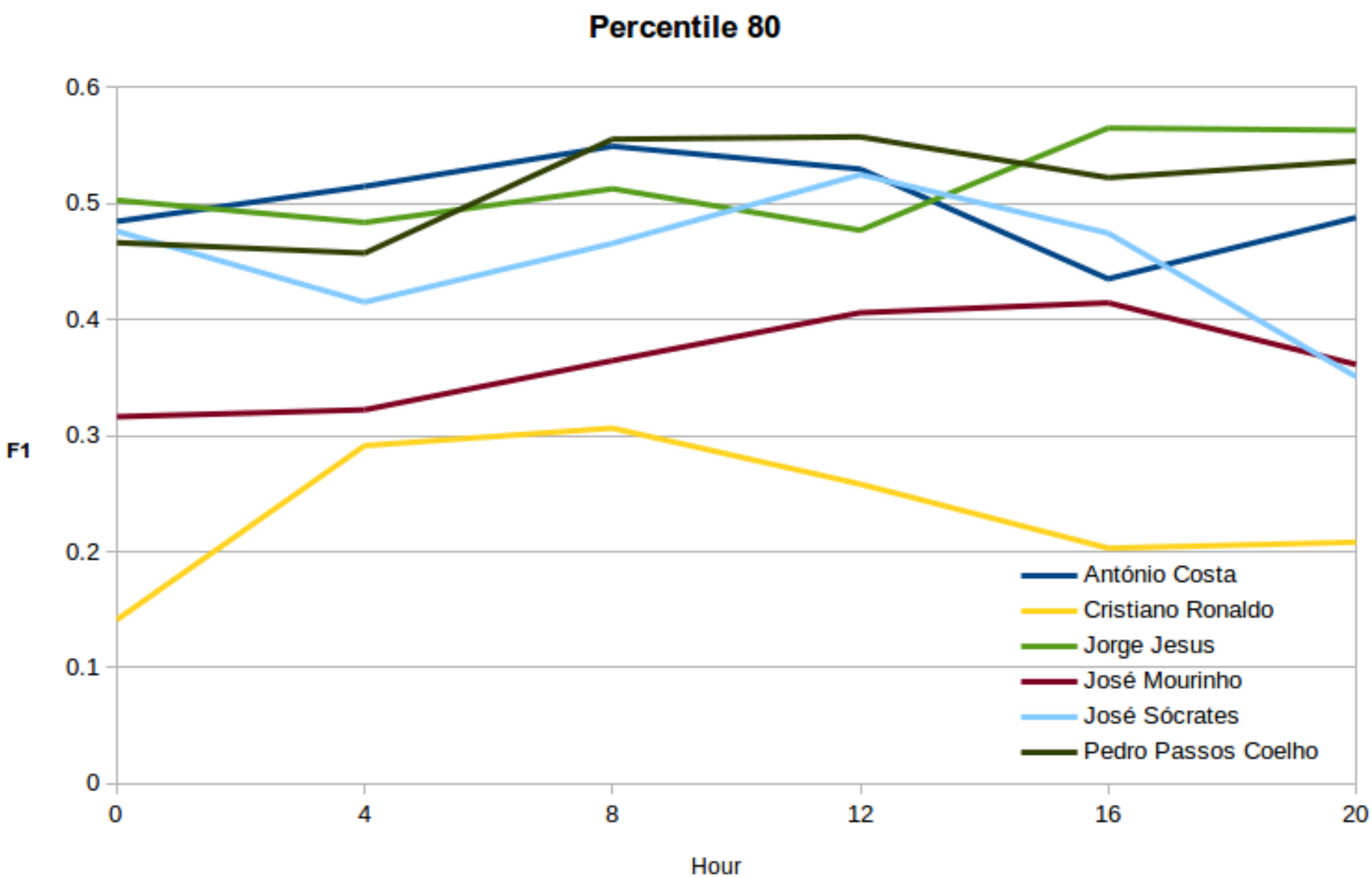

Percentile 50
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
F1
0
4
8
12
16
20
Hour
António Costa
Cristiano Ronaldo
Jorge Jesus
José Mourinho
José Sócrates
Pedro Passos Coelho

Results



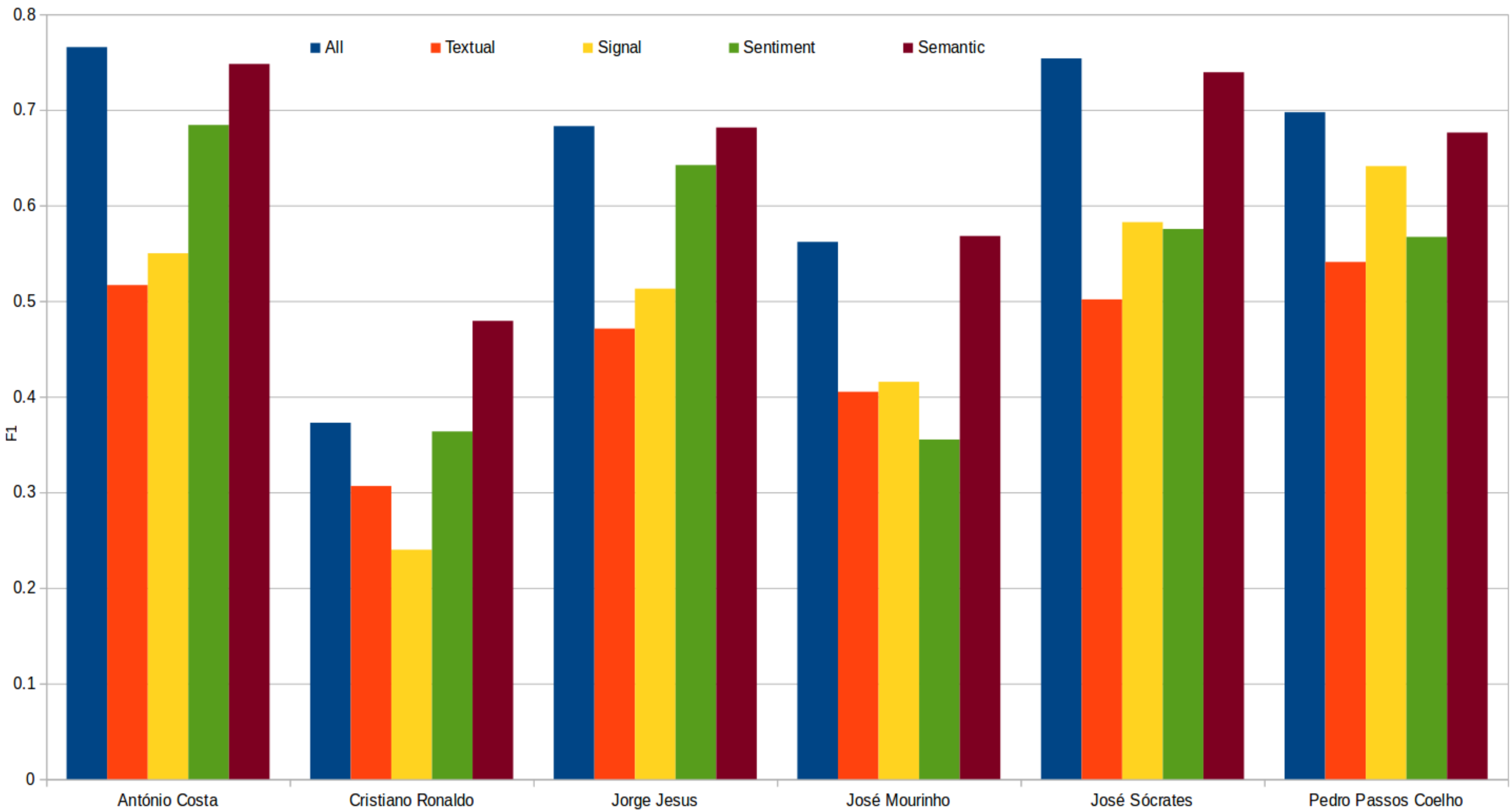
Percentile 65
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
F1
0
4
8
12
16
20
Hour
António Costa
Cristiano Ronaldo
Jorge Jesus
José Mourinho
José Sócrates
Pedro Passos Coelho

Results



Percentile 80
0.6
0.5
0.4
0.3
0.2
0.1
0
F1
0
4
8
12
16
20
Hour
António Costa
Cristiano Ronaldo
Jorge Jesus
José Mourinho
José Sócrates
Pedro Passos Coelho

Results k=0.5, tp=12:00



Results

F1 scores above 0.7 for politicians and balanced dataset (k=0.5)

Moment of prediction influences performance

Semantic and Sentiment are the most informative type of features

Not so good results for live events, e.g. TV debates

Questions?

pssc[at]fe.up.pt