A supervised machine learning method to classify Dutch-language news items

Susan Vermeer

Python 3.5.2

Contents

Based on a supervised machine learning method, we developed a classifier in *Python* (version 3.5.2) that returns the news topic of Dutch-language news items (as a *string*). To train the classifier, we collected more than 1 million news items from approximately 150 different Dutch-language news websites, as well as search engines and social media, collected over 8 months in 2017/18.

Installation

```
pip install numpy #version 1.12.1
pip install scikit-learn #version 0.19.2
pip install pandas #version 0.19.2
```

Usage

There are three *pickle* modules; based on three different pre-processing steps:

1. The **'all words' classifier** maps the original text of a news item into a news category:

```
from sklearn.externals import joblib
clf=joblib.load('PassiveAggressive_text_Dutch_news.pkl')
topic=clf.predict([text])
```

2. The **'stop word' classifier** maps the original text without stop words of a news item into a news category (see section Machine learning):

```
from sklearn.externals import joblib
clf=joblib.load('PassiveAggressive_stopwords_Dutch_news.pkl')
topic=clf.predict([text])
```

3. The **'lead' classifier** maps the first 75 words of the original text without stop words of a news item into a news category (see section Machine learning):

```
from sklearn.externals import joblib
clf=joblib.load('PassiveAggressive_lead_Dutch_news.pkl')
topic=clf.predict([text])
```

Content analysis

We used a coding scheme (developed by Shoemaker & Cohen, 2005) to guide the annotating process (see Appendix). The **unit of analysis** is a single news item. Every news item must contain at least two sentences, and can be presented in different news formats (e.g., articles, columns, etc.). The coding scheme merely consists of one variable, which covers the topic of the news item. This variable distinguishes *four* different topics (i.e., (1) *Politics*, (2) *Business*, (3) *Entertainment* and (4) *Other*), illustrated by various subtopics. The list of subtopics that we developed is rather detailed, so that we can identify the most relevant topic to each news item. It is, however, possible that an item would suitably be annotated as being relevant to more than one topic. In this case, we asked the annotators to indicate the most dominant topic present when merely reading the *first five sentences* of a news item. Finally, if the annotator is indecisive, the topic is not included in the list, or it concerns a cookie consent message, there is a fifth option: *N/A*.

Intercoder reliability

Two human annotators independently annotated approximately 500 news items. The assignment of news items to these four categories reached a Cohen's kappa score of .88, which can be interpreted as almost perfect. On this basis, one annotator analyzed an additional 3,200 news items in a step-wise approach.

Machine learning

Next, we used the Python *scikit-learn machine learning* library (see Pedregosa et al., 2011) to train and test the Passive-Aggressive (PA) algorithm (Crammer, Dekel, Keshet, Shalev-Shwartz, & Singer, 2006), which is known to perform well in various text classification tasks, including Dutch-language news items (see e.g., Burscher, Vliegenthart, & De Vreese, 2015). Before training the classifier, we converted the text to a bag-of-words model and used this as the input for the model. Different pre-processing steps have been used resulting in three different text categories:

1. The **'all words' category** comprises the original text of the news item.

```
text=df['text']
```

2. The **'stop words' category** comprises the original text of the news item without stopwords. We retrieved the list of stop words from the Python NLTK package (see Bird & Loper, 2016):

```
import nltk #version 3.2.4
from nltk.corpus import stopwords
nltk.download('stopwords')
```

   And, we removed Dutch stop words such as articles (e.g., *the*, *a* and *an*), personal pronouns (e.g., *I*, *me* and *he*), coordinating conjunctions (e.g., *for*, *but* and *so*), and prepositions (e.g., *in*, *towards* and *before*).

```
df['text_stop']=df['text'].str.lower()
stop=set(stopwords.words('dutch'))
df['text_stop']=df['text_stop'].str.split()
df['text_stop']=df['text_stop'].apply(lambda x:[item for item in x if item not in stop])
df['text_stop']=df['text_stop'].apply(lambda x:' '.join(x))
text_stop=df['text_stop']
```

3. And, the **'lead' category** comprises the lead (i.e., first 75 words) of a news item after removing stop words, as facts are generally presented in descending order of importance (Pöttker, 2003).

```
df['text_lead']=df['text_stop']
def proc(s):
    l=s.split()
    return ' '.join(l[:75])
df['text_lead']=[proc(s) for s in df['text_lead'].values.tolist()]
text_lead=df['text_lead']
```

This results in three different texts categories:

```
textcolumns={'text':text,'text_stop':text_stop,'text_lead':text_lead}
```

Hyperparameters

We applied a random sampling procedure to split the dataset into a training set (80 percent; N=2,963), on which we trained the three classifiers, and a test set (20 percent; N=738), on which we evaluated the classifiers (Burscher et al., 2015).

```
import sklearn
from sklearn.pipeline import Pipeline
from sklearn.model_selection import train_test_split
from sklearn.model_selection import GridSearchCV
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.linear_model import PassiveAggressiveClassifier
results=pd.DataFrame()
gridsearchresults=pd.DataFrame()
DV='topic'
y=df[DV].as_matrix()
X=df[textcolumns]
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=42)
X_train.shape
```

For every classifier, we measured its ability to accurately classify unseen labelled examples based on the *precision*, *recall* and *accuracy*; based on the following classification report:

```
def classification_report_df(report, report_name):
   report_data=[]
   lines=report.split('\n')
   for line in lines[2:]:
      row={}
      row_data=line.split(' ')
      row_data=[item for item in row_data if len(item)>1]
      if len(row_data)>2:
         row['classifier']=report_name
         row['class']=row_data[0]
         row['precision']=float(row_data[1])
         row['recall']=float(row_data[2])
         row['f1_score']=float(row_data[3])
         row['support']=float(row_data[4])
         report_data.append(row)
   dataframe=pd.DataFrame.from_dict(report_data)
   return dataframe
```

Additionally, we tested various combinations of hyperparameters to find the ultimate combination to tune the classifiers, for example how to convert a collection of text documents to a matrix of token counts (CountVectorizer), whether to transform a count matrix to a normalized tf or tf-idf representation (TfidfTransformer), and the maximum number of passes over the training data (i.e., epochs; see Table 1).

```
def findbestparams(X_train,X_name,y_train,clf_pipeline,parameters,classifiername):
    print('Started with',X_name)
    gs_clf=GridSearchCV(clf_pipeline,parameters,n_jobs=-1)
    gs_clf=gs_clf.fit(X_train[X_name],y_train)
    res=gs_clf.best_params_
    res['textcolumn']=X_name
    res['classifier']=classifiername
    return res

clf_pipeline=Pipeline([('vect',CountVectorizer()),('tfidf',TfidfTransformer()),('clf',
     PassiveAggressiveClassifier(class_weight='balanced'))])
parameters={'vect__ngram_range':[(1,1),(1,2),(1,3)],'tfidf__use_idf':(True,False),
            'clf__loss':('squared_hinge','hinge'),'clf__n_iter':(5,10,15)}

for textcol in textcolumns:
    res=findbestparams(X_train,textcol,y_train,clf_pipeline,parameters,'PassiveAggressive')
    gridsearchresults=gridsearchresults.append(pd.DataFrame([res]))
```

We applied the ultimate combination of hyperparameters to tune the classifiers:

```
def applybestparams_PassiveAggressive(classifier,X_train,X_name,y_train,X_test,y_test,
     gridsearchresults):
    bestparams=gridsearchresults[(gridsearchresults.classifier==classifier)
        &(gridsearchresults.textcolumn==X_name)]
    bp=bestparams.iloc[0].to_dict()
    if bp['tfidf__use_idf']==True:
        text_clf=Pipeline([('vect',CountVectorizer(bp['vect__ngram_range'])),('tfidf',
            TfidfTransformer()),('clf',classifiers[classifier](class_weight='balanced',
            n_iter=bp['clf__n_iter'],loss=bp['clf__loss']))]
    if bp['tfidf__use_idf']==False:
        text_clf=Pipeline([('vect',CountVectorizer(bp['vect__ngram_range'])),('tfidf',
            TfidfTransformer()),('clf',classifiers[classifier](class_weight='balanced',
            n_iter=bp['clf__n_iter'],loss=bp['clf__loss']))])
```

```
    text_clf=text_clf.fit(X_train[X_name],y_train)
    predicted=text_clf.predict(X_test[X_name])
    pred_PA=metrics.classification_report(y_test,predicted)
    res_apply=classification_report_df(pred_PA,'PassiveAggressive_'+X_name)
    joblib.dump(text_clf,'PassiveAggressive_'+X_name+'string.pkl')
    return res_apply
```

And, we measured—for each classifier—its ability to accurately classify unseen labelled examples based on the *precision*, *recall* and *accuracy* (see Table 1).

```
for textcol in textcolumns:
    res_apply=applybestparams_PassiveAggressive('PassiveAggressive',X_train,textcol,y_train
        ,X_test,y_test,gridsearchresults)
    results=results.append(res_apply)
```

Table 1
Performance measures and hyperparameters for the Passive Aggressive algorithm

	All words	Stopwords	Lead
Performance measures			
Accuracy	.82	.82	.81
Precision	.82	.82	.81
Recall	.83	.83	.82
Hyperparameters			
CountVectorizer()	1,1	1,1	1,1
TfidfTransformer()	True	True	True
Optimization Iteration	5.0	10.0	15.0
Hinge-Loss function	L1-loss	L1-loss	L2-loss

Performance measures

Table 2 presents the *precision*, *recall* and *accuracy* for every classifier per topic, and reflects predictions for items outside the training set.

Table 2
Performance measures for the Passive Aggressive algorithm per topic

	Accuracy	Precision	Recall
Politics			
All words	.86	.83	.88
Stopword removal	.85	.83	.86
Lead	.82	.80	.84
Business			
All words	.66	.70	.62
Stopword removal	.68	.77	.60
Lead	.68	.76	.61
Entertainment			
All words	.89	.88	.90
Stopword removal	.89	.85	.93
Lead	.88	.85	.92
Other			
All words	.63	.67	.59
Stopword removal	.61	.68	.55
Lead	.60	.69	.52
N/A			
All words	.87	.84	.80
Stopword removal	.92	.93	.90
Lead	.90	.90	.90

References

Bird, S., & Loper, E. (2016). The natural language toolkit NLTK: The Natural Language Toolkit. In *Proceedings of the acl-02 workshop on effective tools and methodologies for teaching natural language processing and computational linguistics* (pp. 63–70). doi: 10.3115/1225403.1225421

Burscher, B., Vliegenthart, R., & De Vreese, C. H. (2015). Using supervised machine learning to code policy issues: Can classifiers generalize across contexts? *Annals of the American Academy of Political and Social Science*, *659*(1), 122–131. doi: 10.1177/0002716215569441

Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., & Singer, Y. (2006). Online Passive-Aggressive algorithms. *Journal of Machine Learning Research*, *7*, 551–585. doi: 10.1.1.9.3429

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830. doi: 10.1007/s13398-014-0173-7.2

Pöttker, H. (2003). News and its communicative quality: The inverted pyramid—when and why did it appear? *Journalism Studies*, *4*(4), 501–511. doi: 10.1080/1461670032000136596

Shoemaker, P. J., & Cohen, A. A. (2005). *News around the World: Content, Practitioners, and the Public*. New York, NY: Routledge. doi: 10.4324/9780203959091

Appendix

Codebook

Q: What topic is most dominantly present in the news item?

Topic	*Subtopic*	Description
1. Politics	*Internal politics*	News items covering legislative activities (e.g., discussion of a new law), executive activities (e.g., announcement by the president), judicial decisions, constitutional issues, elections, political fundraisers and donations, political appointments, statements and activities of individual politicians, inter-party or relations, internal party relations, activities of interest groups, referendum, public opinion, abuse of political power or corruption, abortion, commission of inquiry, resignation of politician, and fall of government (vote of no confidence);
	International politics	News items about international relations, including activities of international political organizations, individual politicians or political parties, diplomatic visits, negotiations or agreements, promises of aid or cooperation, policy statements, wars between countries, international tensions and disagreements, international terrorism, and embargo. **Not included:** personal stories, e.g., community workers helping tsunami victims **(3) Entertainment**;

	Military and defense	News items about military activities, appointments and firings in the military, government defense policy and action, and protest at government defense policy.
2. Business	*Economy*	News items about the state of economy, economic indexes (e.g., domestic production numbers), job market, appointments, fiscal measures, budget issues, natural resources, monopolies, tariffs, economic legal issues, donations, and stock market situation;
	Labor and industrial relations	News items covering union activities (e.g., lobbying), disputes, strikes, legal measures and policy, relations between employer associations and workers, and condition of workers;
	Business, commerce, industry	News items about business activities, legal measures and policy, international business, globalization, stock market, mergers and acquisitions, e-commerce, technology, tourism, agriculture, trade with foreign countries, and appointments and firings;
	Transportation	News items about transportation systems, public transportation issues, automobiles, driving behaviour, parking issues, aviation, trains, subway, and transportation-related constructions;

Health, welfare, social services	News items covering health policies and legal measures, health insurance issues, health epidemic, new medications, new health technology or medical practice, social services, non-profit organizations, benefit events for a good cause, health malpractice suits, poverty level, poverty conditions, health advice, success in rehabilitation, drug problems, prostitution, and women trafficking;
Education	News items about the general educational policy, funding of education, educational reform, preschool education, secondary education, higher education (colleges and universities), teacher training, teacher wages, students, parental issues, level of teaching and teaching standards, school curriculum, examination, relations between teachers and parents, relations between teachers and students, registration for school, opening and closing of schools, and sectorial education (e.g., religious vs. secular). **Not included:** debates about the political decision-making process related to education **(1) Politics**;
Energy	News items about energy supply, energy costs, and technology developments.

3. Entertainment	*Internal order*	News items about civil war, peaceful demonstrations, violent demonstrations, crime levels, small crimes, police management, espionage, fire brigade, prison conditions, corruption (**Not included:** political corruption **(1) Politics**), police behaviour, white collar crime, judicial decisions, child abuse, pedophilia, violence, political assassinations, murder, robbery, crime investigation, assault, rape, criminal association (e.g., Mafia), fraud, and libel suit;
	Housing	News items related to housing supply, living conditions, construction, mortgages, building permits, city planning, and housing demolition;
	Social relations	News items covering gender relations, sexual orientation issues, ethnic relations, class relations, age differences, family relations, and minority-majority relations;
	Accidents and disasters	News items about natural disasters, fire, and other accidents (e.g., car, plane, train, work, military-related, home, crowd);

Sports	News items covering sports results, training, records, individual athletes, coaches, teams, leagues, drug use in sports, fan behaviour, legal measures, appointments and firings, events, Olympic training, and championships. **Not included:** National economic benefits due to organizing a sports event **(2) Business**;
Culture	Music, theatre, opera, dance, film, photography, literature and poetry, painting and sculpturing, television shows, radio shows, museums, general exhibits, festivals and competitions, and prizes and awards. **Not included:** News items about culture-related policies **(1) Politics** or subsidies **(2) Business**;
Fashion	Fashion shows, beauty contests, models, fashion products, and fashion trends (e.g., trend colors, body piercing);
Ceremonies	Official government ceremonies, national holiday ceremonies, ethnic ceremonies, and anniversaries of events;
Human interest	Celebrities, non-celebrities, animal stories, travel stories, record attempts, supernatural or mystical stories, trends, games, gadgets, mystery, food, advice (e.g., on love, insurance, stock), and lottery results.

4. Other	*Population*	News items about general population statistics, communities, values, immigration, emigration, and visa issues;
	Science and technology	News items covering standards, inventions, individual scientists, scientific organizations, computer issues, multimedia issues, space exploration, and problems related to science or technology. **Not included:** news items about the political consequences **(1) Politics**;
	Communication	News items covering industry-wide issues and statistics, journalism and media in general, newspapers, network television, cable television, radio, magazines, Internet, (mobile) phones, media regulation, and technical aspects of communication;
	Environment	News items covering threats to environment (e.g., pollution), natural resources, activities of environmental organizations, garbage collection, and conservation (e.g., energy saving or parks). **Not included:** Economical considerations from a social-economic perspective **(2) Business**;
	Weather	News items comprising weather maps and statistics, forecasting, weather warnings, weather phenomena, and general weather stories (e.g., coldest winter);

	Religion	Religious holidays or ceremonies, religious proclamations by senior religious leaders, conflict between religious groups, religious tourism, and holy places. **Not included:** Political debates about religion or integration issues of religious minorities **(1) Politics**.
5. N/A		”I don’t know”, ”Other, namely _____”, a cookie consent message, or a message asking to disable AdBlock.