



SentiCode: A new paradigm for one-time training and global prediction in multilingual sentiment analysis

Mohamed Raouf Kanfoud¹ · Abdelkrim Bouramoul¹

Received: 28 December 2021 / Revised: 27 April 2022 / Accepted: 28 April 2022 /

Published online: 25 May 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

The main objective of multilingual sentiment analysis is to analyze reviews regardless of the original language in which they are written. Switching from one language to another is very common on social media platforms. Analyzing these multilingual reviews is a challenge since each language is different in terms of syntax, grammar, etc. This paper presents a new language-independent representation approach for sentiment analysis, SentiCode. Unlike previous work in multilingual sentiment analysis, the proposed approach does not rely on machine translation to bridge the gap between different languages. Instead, it exploits common features of languages, such as part-of-speech tags used in Universal Dependencies. Equally important, SentiCode enables sentiment analysis in multi-language and multi-domain environments simultaneously. Several experiments were conducted using machine/deep learning techniques to evaluate the performance of SentiCode in multilingual (English, French, German, Arabic, and Russian) and multi-domain environments. In addition, the vocabulary proposed by SentiCode and the effect of each token were evaluated by the ablation method. The results highlight the 70% accuracy of SentiCode, with the best trade-off between efficiency and computing time (training and testing) in a total of about 0.67 seconds, which is very convenient for real-time applications.

Keywords Multilingual sentiment analysis · SentiCode · Machine learning · Natural language processing · Part-of-speech tags

1 Introduction

Sentiment analysis applies natural language processing, opinion mining, and text analytics to derive subjective information from textual data. The online community is becoming an important part of today's business world. The increase in Internet users from different cultures implies a high variability in the style and languages used to express opinions. A

✉ Mohamed Raouf Kanfoud
raouf.kanfoud@univ-constantine2.dz

Abdelkrim Bouramoul
abdelkrim.bouramoul@univ-constantine2.dz

¹ MISC Laboratory, Constantine 2 University Abdelhamid Mehri, Constantine 25000, Algeria

variety of general opinions that fall into the category of sentiment analysis can be generated by explicit or indirect measures. This involves determining whether or not the opinion is related to a topic or product of interest to gain insight into how people feel about the product, service, or brand online. Analyzing these opinions derived from social media platforms (Singh & Singh, 2021) such as Facebook, Twitter, and YouTube is challenging, especially when multiple languages are used (Londhe et al., 2021). The objective is to provide high-quality performance (accuracy and precision) and a high level of coverage (recall). Researchers have proposed and developed different approaches categorized into different categories machine learning (Sebastiani, 2002) knowledge-based (Cambria, 2016) lexicons (Taboada et al., 2011) rule/case-based (Berka, 2020) and hybrid (Appel et al., 2016) for specific languages (often for English), but they do not contain all relevant expressions or cover all languages. Although multilingual sentiment analysis has attracted much interest from researchers, most efforts are focused on English sentiment analysis. However, most efforts are focused on using standard machine translation, either for training or prediction or even cross-lingual in both phases. Certainly, the performance of these methods is influenced by the quality of the translation process (Balahur & Turchi, 2012), and it depends more on its existence. In this work, we propose a new method called SentiCode (SENTiment CODE), which allows the representation of various features in English and non-English languages without involving machine translation at any stage. We take advantage of existing tools in the different languages, such as part-of-speech taggers, which are by far the most efficient and capable of 99% accuracy. A series of extensive experiments were conducted to evaluate the performance of SentiCode in multilingual and multi-domain environments. We also established a trade-off between efficiency and computing time (i.e., training and prediction). Furthermore, we compared different features, such as unigrams and bigrams on vector space representations. An additional experiment on the training data size was also carried out. SentiCode was trained using different traditional machine learning and deep learning techniques. The rest of the paper is organized as follows: Section 2 underlines previous work in the field; Section 3 presents the SentiCode method; Section 4 describes the techniques and libraries used in the experiment; Section 5 discusses the results, while Section 6 concludes the paper and highlights future work.

2 Related work

Most existing methods for sentiment analysis are designed to address English language content or, at most, evaluate the English language and claim that the proposed approach applies to other languages. However, these methods regarding multilingual sentiment analysis rely on the machine translation approach to bridge the gap between different languages (Banea et al., 2008; Balahur & Turchi, 2012; Araujo et al., 2016). There are two approaches to sentiment analysis widely adopted in the literature. The first, is based on the so-called lexicon-based methods. Lexicons are constructed manually or automatically, starting with a seed word list then expanding this list from dictionaries or corpus. Some of these lexicons contain a list of words labeled with their polarity, such as the General Inquirer (Stone et al., 1966). Similarly, the MPQA Subjectivity lexicon (Wilson et al., 2005) includes over 8000 words annotated into four classes of feelings: positive, negative, both, and neutral. Another rich lexicon is SentiWordNet (SWN) (Baccianella et al., 2010). It is a lexical resource for sentiment analysis, and it was created based on the English lexical called WordNet; the latter groups different grammatical classes. SentiWordNet assigns

positive and negative scores to each synset (group of synonyms). In contrast, the objective score is obtained by: one minus the positive score plus the negative (objective = 1 - (positive + minus)). There are two versions of SentiWordNet: SentiWordNet 1.0 (Esuli & Sebastiani, 2006) annotates WordNet 2.0 and SentiWordNet3.0 (Baccianella et al., 2010) annotates WordNet 3.0. (Hutto & Gilbert, 2014) introduced the VADER model (Valence Aware Dictionary for sEntiment Reasoning) for sentiment analysis in social media texts. The core of VADER is based on a sentiment lexicon containing a list of words, western emoticons, acronyms, and slang commonly used in social texts. These features are associated with their sentiment intensity, ranging from -4 to 4. The annotation was performed by Turkers recruited from Amazon Mechanical Turk. In addition to the lexicon, VADER employs five manually generalized heuristics: punctuation, capitalization, degree modifiers, opposing expressions, and negation. The authors of VADER claimed that it is not a black box and can be extended or modified for other purposes. It should be noted that General Inquirer, MPQA Subjectivity Lexicon, SentiWordNet, and VADER are freely available and widely used in the literature. Unlike the general lexicon mentioned above, there are specialized lexicons for specific domains. Yekrangi & Abdolvand, (2021) constructed a lexicon for the financial market's domain. The authors evaluated its performance by calculating the correlation between dollar price trends and sentiment scores, which averages 60% due to the finance-specific words in the specialized lexicon. Then there are the so-called machine learning techniques, which are widely adopted. (Pang et al., 2002) used three machine learning techniques, including Naïve Bayes (NB), Maximum Entropy (ME), and Support Vector Machine (SVM) for sentiment classification. These three techniques were tested and compared on a set of features (Pang et al., 2002). The results indicate that unigrams and presence count are the most effective for movie reviews. Also, these machine learning techniques provide a good result compared to the baseline based on human-rated word lists. Besides, the authors pointed out that SVM is the most effective while NB is the least effective, but with small differences. (Tripathy et al., 2016) used four machine learning techniques: SVM, NB ME, SGD (Stochastic Gradient Descent) and fed them with n-gram and TF-IDF (term frequency-inverse document frequency) features extracted from IMDb (Internet Movie Database) movie reviews. The results show that unigram and bigram perform better than trigram or more. (Jagdale et al., 2019) selected two machine learning algorithms, namely NB and SVM, to analyze the Amazon website reviews in different categories such as cameras, tablets, and TVs. The most adopted machine learning techniques are SVM and NB from the previously mentioned work. Moreover, SVM gives very promising results. Recently (Zhang et al., 2019) proposed a new model called quantum-inspired sentiment representation, based on quantum theory to represent semantic information and sentiment. Combining the two approaches (lexicon-based and machine learning-based) could give better results than the individual approach, as the work by (Lu & Tsou, 2010) on the diversity of output results between lexicon and machine learning techniques. In conversational sentiment analysis, a bidirectional emotional recurrent unit (BiERU) is used to extract the sentiment of each message in a text-based conversation (Li et al., 2022). The emotional recurrent units are used with fewer parameters compatible with multiparty conversations without adjustments. Another aspect of sentiment analysis is detecting sentiment on different product entities. Aspect-based sentiment analysis (ABSA) refers to considering the characteristics of a product or entity instead of a document or sentence level (Do et al., 2019). In (Liang et al., 2022), an LSTM (Long Short-Term Memory) layer was used to capture context, and a GCN layer (graph convolutional network) was built to capture the potential sentiment in a particular context. ABSA is beyond the scope of this paper.

Multilingual Sentiment analysis (MSA) is more challenging and has gained interest over the years. Boiy & Moens, (2009) analyzed multilingual (English, Dutch, and French) and multi-domain sentiments while using various representations such as unigrams, negation, discourse features, compound words, and verbs for sentiment classification. The authors proposed a three-layer cascade approach starting with detecting whether the text is opinionated or not; if yes, the second level is called to determine its polarity, and if the output of this second level is not above a threshold, then a third level is needed. Banea et al., (2010) used multiple features from several languages and showed how these features would behave together for the source and target language experience in six languages: English, Arabic, French, German, Romanian and Spanish. While (Cui et al., 2011) Integrate characters and punctuation next to emoticons to evaluate multilingual tweets. Ghorbel, (2012) classified French posts from a forum. The proposed method is based on extracting word polarity from the following PoS (part of speech) tags: adjectives, adverbs, nouns, and verbs from SWN. As SWN is in English and the posts to be classified are in French, the author used/compared two ways of translation: using traditional machine translation or translation by EuroWordNet. Štajner et al., (2013) compared different combinations of features and showed that machine translation could generate a training corpus in another language. The authors also showed that general-purpose lexicons could achieve similar performance to domain-specific lexicons. Balahur & Turchi, (2014) conducted a series of experiments based on translated data from the SemEval 2013 corpus in four other languages to evaluate sentiment analysis systems in a multilingual environment. In the end, they concluded that poor translation has a bad effect on the final sentiment classification. Balahur et al., (2015) performed polarity classification on monolingual and multilingual texts (English, Spanish, and English-Spanish) comparing three techniques: Monolingual model, Monolingual pipeline model, and Multilingual model. In addition to training an English-Spanish tagger to handle code-switched texts. The authors found that the multilingual model performs better for code-switched text. Therefore, they recommend it as the best choice when the messages to be analyzed are in multiple languages. Zhou et al., (2016) conducted a cross-lingual analysis using English as the source language and French, German, and Japanese as the target language. The proposed model called BiDRL (Bilingual Document Representation Learning) is based on integrating bilingual representation words from the source and target languages using machine translation to translate between them. Nguyen & Le Nguyen, (2018) used convolutional neural network (CNN) and LSTM over n-gram word embedding to capture the sentiment of multilingual and multi-domain YouTube comments by applying experiments on the SenTube dataset. The authors claimed that the proposed model applies to any language without relying on linguistic features. Chen et al., (2019) proposed a novel ELSA (Emoji-powered representation learning for Cross-Lingual Sentiment Analysis) framework to solve the problem of cross-lingual sentiment analysis using emojis as an instrument. In terms of accuracy, ELSA outperforms the three compared methods, namely MT-BOW (Machine Translation - Bag of Words), CL-RL (Cross-Lingual Representation Learning), and BiDRL (Bilingual Document Representation Learning). In (Cruz Paulino et al., 2021), the authors used tweets about COVID-19 written in Filipino or English to conduct their experiment. The Filipino tweets were translated into English to be annotated. Then, building classifier models to evaluate these multilingual tweets by employing count vectorizers as features. SVC was found to be the best-performed model. (Chen et al., 2019) and (Cruz Paulino et al., 2021) used machine translation to bridge the gap between the languages used in their experiments. Therefore, multilingual

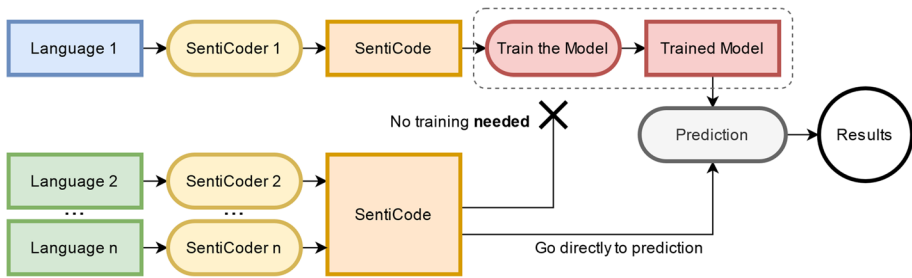


Fig. 1 The main idea and goal of the proposed approach SentiCode

sentiment analysis is an active research area. Many studies have been conducted on MSA (Lo et al., 2017; Abdullah & Rusli, 2021; Agüero-Torales et al., 2021), discussing, comparing, and reviewing concepts, results, and trends.

3 Proposed method

This section first highlights the challenge of multilingual sentiment analysis and explains the motivation of the proposed approach. Then, the SentiCode algorithm, its process, and its vocabulary are described.

3.1 Problem and motivation

There are about 7000 languages¹ in the world. People express their opinions in different languages, whether native or preferred, through different support and canals such as social media platforms. At the same time, with the emergence of online business entities (e.g., e-commerce), producers and buyers need to know people’s opinions and feelings about other people, products, events, or anything else for different reasons. To this end, sentiment analysis techniques are applied. One of these techniques is a supervised machine learning algorithm, which requires prior training; therefore, labeled data is required.

People use different languages, so it is mandatory to train models for different languages. Moreover, this process will also require labeled data for each language (which is difficult to obtain), not to mention the required training time. Thus, considering what was highlighted above and following the adage “train once, use many times”, this work proposes an approach that allows us to train a model and then exploit it several times to deal with multilingual sentiment analysis.

3.2 The core idea of the proposal

We propose an approach called the SentiCode approach (SentiCode for SENTiment CODE), where instead of training a new model for each language with all the laborious work that follows, we focus our efforts on a single abstraction, representation, pseudocode and train a single model capable of handling many other languages, saving time and effort

¹ <https://www.ethnologue.com/guides/how-many-languages>

Table 1 The SentiCode vocabulary

Tokens	Meaning	Role	Example
NOT	Negative term	Handle negation	no, not, never . . .
POS	Positive	Prior positive sentiment	/
NEG	Negative	Prior negative sentiment	/
ADJ	Adjective	Bear sentiment	good, bad
ADV	Adverb	Bear sentiment	badly
NOUN	Noun	Bear sentiment	surprises
VERB	Verb	Bear sentiment	like, hate

(data collection and labeling, retraining, etc.) The SentiCode approach uses SentiCoder (see Algorithm 1), which generates a code from the raw text of the studied language. The generated code is named SentiCode (like the name of the proposed approach). Figure 1 illustrates the main idea and goal of the SentiCode approach. The full process of SentiCode uses a symbolic approach (extract knowledge-based features such as adjectives, adverbs, negation, etc., to generate SentiCode) and a sub symbolic approach (statistical and machine learning algorithms to infer the expressed sentiment in text) (Cambria et al., 2020).

3.3 Proposed algorithm for SentiCode

The syntax or vocabulary of SentiCode should include common features between languages. Thus, in this paper, we detail the first implementation of SentiCode (other versions of SentiCode will be studied in future work). In the first version, we considered the grammatical units such as adjectives, adverbs, nouns, and verbs for the reason that these units are sentiment-bearing words (Taboada et al., 2011; Polanyi & Zaenen, 2006; Hatzivassiloglou & Wiebe, 2000). Hence, they are referred to as bearer tokens. Subsequently, to extract this feature, we use PoS taggers. The Universal Dependencies (Nivre et al., 2020) PoS tags² annotation was followed to apply a cross-lingual representation for all bearer tokens. Furthermore, when a positive word is combined with a negation, such as “not”, it takes on a negative polarity (Polanyi & Zaenen, 2006), and the opposite is also correct. Therefore, negation is handled in SentiCode. All negation terms are replaced by the token NOT. This rule is applied to all languages. Furthermore, to tackle the task of sentiment analysis, we enriched the SentiCode vocabulary with two tokens representing the prior polarity of bearer tokens, the POS token stands for the positive polarity, and the NEG token stands for negative polarity. POS and NEG are placed in front of bearer tokens. The vocabulary size is 7, including ADJ, ADV, NOUN, VERB, NOT, POS, and NEG. Table 1 summarizes the vocabulary of SentiCode.

² <https://universaldependencies.org/u/pos/>

Algorithm 1 The proposed algorithm of SentiCode.

Require: S: Sentence in natural language
Ensure: O: Equivalent sentence in SentiCode

- 1: tags \leftarrow Generate PoS tags of the sentence S
- 2: **for** word, tag IN tags **do**
- 3: **if** word in negation_list **then**
- 4: comment : Negation term
- 5: O.append('NOT')
- 6: **else if** tag IN ['ADJ', 'ADV', 'NOUN', 'VERB'] **then**
- 7: **if** word IS positive **then**
- 8: polarity \leftarrow 'POS'
- 9: **else if** word IS negative **then**
- 10: polarity \leftarrow 'NEG'
- 11: **end if**
- 12: o.append(tag)
- 13: **if** polarity == 'POS' OR polarity == 'NEG' **then**
- 14: comment : Add word polarity
- 15: O.append(polarity)
- 16: **end if**
- 17: **end if**
- 18: **end for**

4 Experimental setup

This section describes the data that was used in the experiment. In addition, we present the generated SentiCode and the machine learning techniques and tools used to evaluate the proposed approach.

4.1 Dataset

Since we addressed a multilingual sentiment analysis task, we needed a multilingual dataset. Therefore, our experiments were performed on the dataset (Prettenhofer & Stein, 2010a) presented in (Prettenhofer & Stein, 2010b). It consists of Amazon reviews in three categories (books, DVDs, and music) in four languages (English, French, German and Japanese). However, we only used English, French, and German to evaluate the proposed approach. There is a training document (file name: training.review) and a test document (file name: test.review) containing 2000 reviews each for each language-domain pair. The last document (file name: unlabeled.review) was discarded. The structure of the files is in XML format, where each review has attributes described in Table 2. According to (Blitzer et al., 2007), a review with a rating > 3 is labeled as positive, while a review with a rating < 3 is labeled as negative. Thus, we obtained a balanced dataset between positive and negative reviews, i.e., the number of positive reviews = the number of negative reviews (1000 reviews for each category). This labeling was applied for all language-domain pairs

Table 2 Review attributes

Attribute	Description
Summary	Summary review
Rating	A sentiment score varies from 0 to 5
Text	Text review
Category	Review’s domain (books, DVDs, music)

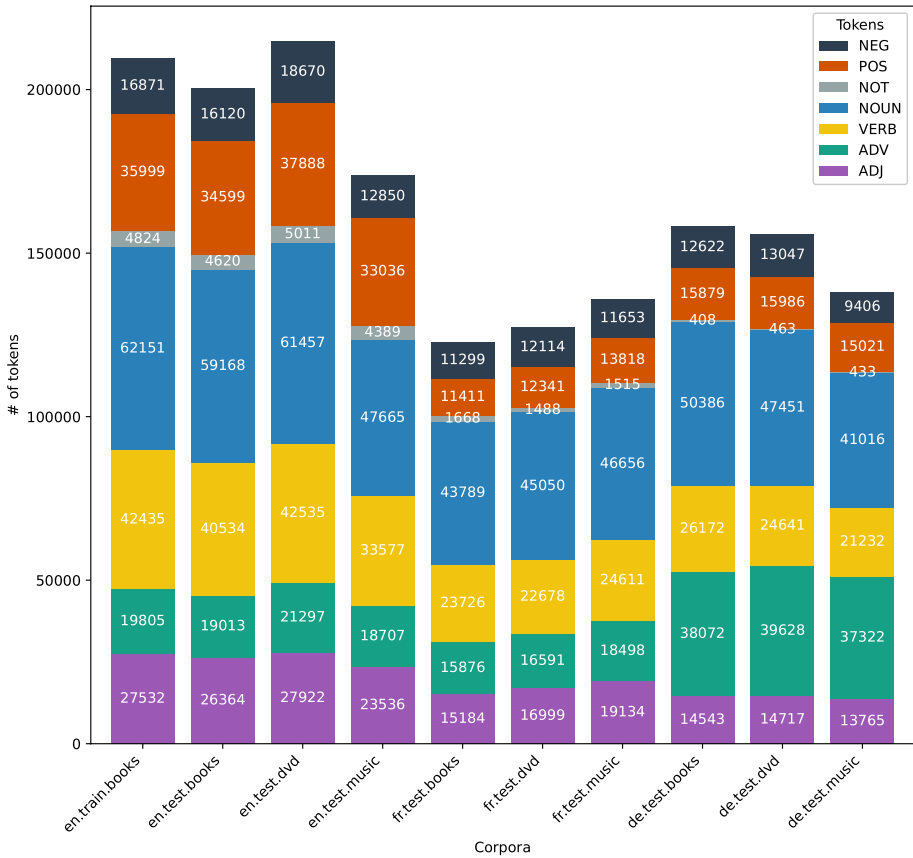


Fig. 2 Tokens distribution in SentiCode of the different testing corpora. (The stacked values in bars have the same order in legend upper right)

(English books, English DVDs, English music, French books, French DVDs, French music, German books, German DVDs, and German music).

4.2 Generated SentiCode for the experiments

This subsection presents the statistics of the generated SentiCode corpora (en.train.books, en.test.books, en.test.dvd, en.test.music, fr.test.books, fr.test.dvd, fr.test.music, de.test.books, de.test.dvd, and de.test.music) from the raw text (Prettenhofer & Stein, 2010b) of

Table 3 Example of the generated SentiCode from a raw text

Review	It's an educational book to read. My child and I like reading it together
PoS tags	('It', 'PRON'), ('s', 'VERB'), ('an', 'DET'), ('educational', 'ADJ'), ('book', 'NOUN'), ('to', 'PRT'), ('read', 'VERB'), ('.', '.'), ('My', 'PRON'), ('child', 'NOUN'), ('and', 'CONJ'), ('I', 'PRON'), ('like', 'VERB'), ('reading', 'VERB'), ('it', 'PRON'), ('together', 'ADV')]
SentiCode	VERB ADJ NOUN VERB POS NOUN POS VERB POS VERB ADV

Table 4 Characteristics of PoS taggers and negation terms used for English, French, and German

Languages	PoS taggers (spaCy)	Negation
English	Trained on	en_core_web_sm
	Accuracy	0.97
French	Trained on	fr_core_news_sm
	Accuracy	0.97
German	Trained on	de_core_news_sm
	Accuracy	0.98
		no, not, none, nobody, nothing, neither, nowhere, never n'est, n'a, ne, non keine, nein, no, ohne, neimals, gegen, weniger (Wiegand et al., 2018)

three languages in a stacked bar chart for comparison in Figure 2. In addition it gives an example of the generated SentiCode by Algorithm 1. This example in Table 3 shows an original review in English, its PoS tags and finally the equivalent SentiCode. It is important to note that some reviews are so short that their PoS tags cannot be generated. For example, the review "Hallo" from the German books corpus was ignored.

4.3 Machine learning techniques and tools

To evaluate the SentiCode approach, we first conducted an extensive experiment on three languages: English, French, and German. We evaluated SentiCode on machine learning by selecting four state-of-art machine learning algorithms (Pang et al., 2002; Tripathy et al., 2016; Ahuja et al., 2019) along with multilayer perceptron (MLP) to test its consistency in sentiment analysis against different classifiers. These machine learning algorithms were employed (trained/tested) with default parameters defined by the Scikit-learn library (Pedregosa et al., 2011), except for MLP is tuned with $\alpha=1e-5$ along with one hidden layer with 10 neurons and $\text{random_state}=1$, while the rest of parameters were kept as the default ones. Algorithm 1 was applied to generate the SentiCode. In addition, PoS tags were extracted using the spaCy library. Table 4 summarizes the characteristics of the PoS taggers used for the three languages. Also, Table 4 lists the negation terms that were replaced by the NOT token in the SentiCode.

All parts of the system were implemented in Python³ language. Additionally, the Scikit-Learn library⁴ (Pedregosa et al., 2011) was used for the machine learning algorithms and spaCy⁵ to parse the raw text and extract PoS tags. Furthermore, to obtain the prior polarity

³ Python version 3.7.0

⁴ Scikit-learn library version 0.24.2

⁵ spaCy library version 3.1.0

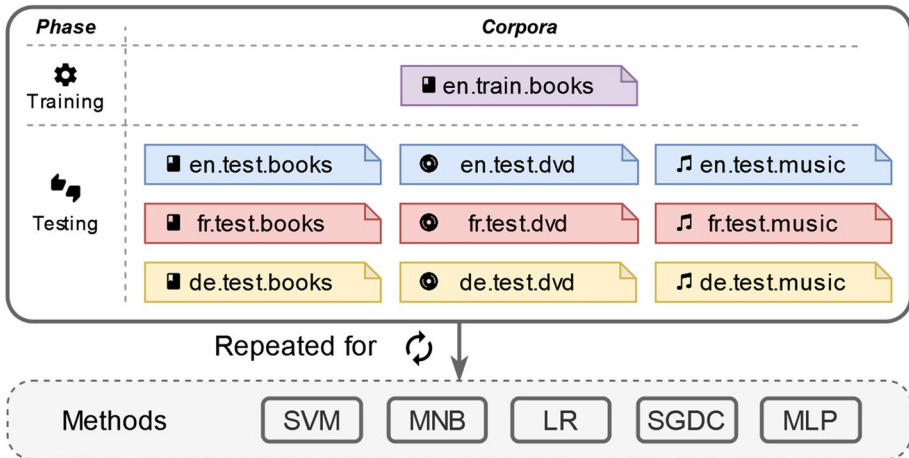


Fig. 3 Cross-language-domain training/testing process repeated over five different classifiers

of the bearer tokens, the GraLexi approach (Kanfoud & Bouramoul, 2022) was used for English; for French and German, we used another list (Chen & Skiena, 2014). Next, we used the vector space model, namely TF-IDF. TF-IDF is a numerical statistic that relates the importance of a term to a document in a collection or corpus. In simple terms, TF-IDF measures the importance of a term relative to a document. Furthermore, we compared the results obtained using uni-gram, bi-gram, and uni-gram + bi-gram. After that, we started the training process with the SentiCode provided in en.train.books (i.e., English with books domain). Then we evaluated it with nine pairs of language-domains, namely: English-books (en.test.books), English-DVDs (en.test.dvd), English-music (en.test.music), French-books (fr.test.books), French-DVDs (fr.test.dvd), French-music (fr.test.music), German-books (de.test.books), German-DVDs (de.test.dvd), German-music (de.test.music). In other words, the models were tested with the same/different languages and the same/different domains. Figure 3 represents the cross-language-domain training and testing process repeated on five different classifiers (i.e., SVM, MNB, LR, SGDC, MLP)

5 Results and discussion

This section discusses the results obtained. First, we demonstrated that SentiCode achieves consistent results using different machine learning algorithms. Next, we evaluated the SentiCode vocabulary by applying the ablation technique to show the impact of each token in the proposed vocabulary. Then, the cross-language and cross-domain validations were performed to demonstrate the effectiveness of the trained SentiCode on languages other than English and domains other than books. After that, we evaluated the SentiCode trade-off between performance and computing time (training and prediction). The error analysis of misclassified reviews is presented, along with some suggestions for correcting the final classification. In the last evaluation, we evaluated SentiCode on corpora with a combined language and domain (the corpus in training and the corpus in prediction gather SentiCode generated from English, French, and German of three domains: books, DVDs, and music). The last step assessed SentiCode with non-western languages (Arabic and Russian) and evaluated the ternary classification (positive, negative, and neutral).

5.1 Early results of SentiCode

This subsection presents our results in terms of accuracy (Acc.), macro f1-score (M-F1, as the data was balanced), and Matthews correlation coefficient (MCC) (Chicco & Jurman, 2020). Tables 5, 6, 7, 8 and 9 report all the obtained results using SVM, MNB, LR, SGDC, and MLP, respectively

5.2 Quality analysis of the SentiCode vocabulary

After reporting the results of SentiCode on the same and different languages and domains, the next step was to evaluate the vocabulary effect. Subsequently, we determined which tokens significantly impact the quality of discrimination between classes. In short, we removed token by token by decreasing the size of the SentiCode vocabulary. The experiment was conducted on English-books corpora (trained with the en.train.books corpus, then tested with the en.test.books corpus) with six tunings:

Table 5 SentiCode results using the SVM algorithm

Tested on	uni-gram			bi-gram			uni-gram + bi-gram		
	Acc.	M-F1	MCC	Acc.	M-F1	MCC	Acc.	M-F1	MCC
en.test.books	67.90	67.89	35.82	70.55	70.55	41.10	70.40	70.40	40.80
en.test.dvd	67.70	67.67	35.46	68.45	68.37	37.08	69.30	69.25	38.73
en.test.music	67.20	67.17	34.46	67.90	67.88	35.83	68.50	68.46	37.10
fr.test.books	65.65	65.65	31.31	62.25	62.06	24.75	64.00	63.88	28.18
fr.test.dvd	66.20	66.20	32.40	62.00	61.68	24.42	64.65	64.47	29.60
fr.test.music	63.81	63.77	27.69	61.11	60.53	22.88	62.56	62.09	25.74
de.test.books	60.68	59.71	22.46	60.03	59.16	20.96	59.93	57.12	23.09
de.test.dvd	62.65	62.39	25.66	59.80	59.15	20.25	60.00	57.75	22.55
de.test.music	59.70	58.72	20.39	59.60	58.55	20.25	58.00	54.78	18.92

Table 6 SentiCode results using the MNB algorithm

Tested on	uni-gram			bi-gram			uni-gram + bi-gram		
	Acc.	M-F1	MCC	Acc.	M-F1	MCC	Acc.	M-F1	MCC
en.test.books	67.50	67.44	35.14	69.00	68.95	38.13	68.65	68.57	37.49
en.test.dvd	67.25	67.23	34.54	68.20	68.16	36.49	67.95	67.93	35.95
en.test.music	67.90	67.87	35.86	68.25	68.21	36.59	68.00	67.94	36.14
fr.test.books	66.40	66.40	32.81	64.90	64.71	30.13	65.60	65.50	31.38
fr.test.dvd	67.25	67.25	34.51	64.60	64.17	29.92	65.45	65.20	31.35
fr.test.music	66.17	66.02	32.60	63.66	62.78	28.68	64.46	63.80	30.01
de.test.books	63.18	63.18	26.36	61.03	59.46	23.98	61.28	60.43	23.58
de.test.dvd	62.35	62.19	24.91	60.20	59.03	21.68	61.85	61.45	24.20
de.test.music	64.95	64.93	29.93	60.10	57.91	22.70	62.30	61.11	26.26

Table 7 SentiCode results using the LR algorithm

Tested on	uni-gram			bi-gram			uni-gram + bi-gram		
	Acc.	M-F1	MCC	Acc.	M-F1	MCC	Acc.	M-F1	MCC
en.test.books	67.05	67.02	34.16	70.50	70.50	41.01	69.75	69.74	39.53
en.test.dvd	68.10	68.04	36.33	69.65	69.61	39.40	69.05	69.01	38.20
en.test.music	68.05	68.01	36.19	68.50	68.43	37.16	68.50	68.45	37.13
fr.test.books	66.85	66.54	34.33	66.50	66.49	33.03	66.50	66.43	33.15
fr.test.dvd	67.50	67.31	35.42	66.00	65.99	32.03	67.05	67.05	34.11
fr.test.music	67.12	67.09	34.29	63.91	63.79	28.00	65.82	65.80	31.66
de.test.books	63.83	63.70	27.87	59.73	58.03	21.23	62.33	61.98	25.12
de.test.dvd	62.95	62.46	26.61	61.00	59.85	23.38	62.70	62.57	25.58
de.test.music	65.20	65.20	30.41	59.35	56.56	21.70	62.20	61.09	25.93

Table 8 SentiCode results using the SGDC algorithm

Tested on	uni-gram			bi-gram			uni-gram + bi-gram		
	Acc.	M-F1	MCC	Acc.	M-F1	MCC	Acc.	M-F1	MCC
en.test.books	68.00	68.00	36.01	70.00	69.86	40.39	70.85	70.85	41.70
en.test.dvd	68.20	68.13	36.56	69.50	69.49	39.03	69.90	69.80	40.07
en.test.music	67.75	67.73	35.55	68.70	68.42	38.08	68.70	68.67	37.48
fr.test.books	65.50	65.27	31.42	64.30	64.25	28.69	64.35	64.35	28.70
fr.test.dvd	66.35	66.23	32.93	64.90	64.71	30.12	65.00	64.97	30.05
fr.test.music	66.87	66.87	33.74	62.91	62.37	26.57	63.06	62.89	26.36
de.test.books	64.13	64.13	28.27	60.43	60.11	21.20	61.48	61.06	23.46
de.test.dvd	62.90	62.80	25.94	61.25	61.05	22.74	61.00	60.61	22.44
de.test.music	63.95	63.78	28.17	60.60	59.51	22.44	60.85	59.29	23.58

Table 9 SentiCode results using MLP

Tested on	uni-gram			bi-gram			uni-gram + bi-gram		
	Acc.	M-F1	MCC	Acc.	M-F1	MCC	Acc.	M-F1	MCC
en.test.books	67.53	67.53	35.07	70.69	70.69	41.37	70.64	70.63	41.27
en.test.dvd	68.48	68.41	37.16	68.73	68.70	37.56	69.28	69.25	38.65
en.test.music	67.98	67.98	35.98	68.88	68.88	37.78	68.58	68.56	37.23
fr.test.books	64.83	64.39	30.42	64.98	64.96	30.01	64.68	64.68	29.37
fr.test.dvd	66.33	66.05	33.23	63.98	63.97	27.98	64.33	64.29	28.73
fr.test.music	66.00	65.97	32.07	63.35	63.29	26.76	62.29	62.11	24.80
de.test.books	63.78	63.65	27.76	58.88	56.67	19.93	58.28	55.09	19.61
de.test.dvd	63.04	62.63	26.71	58.29	56.36	18.24	57.29	54.43	16.80
de.test.music	64.48	64.46	28.99	59.27	56.67	21.26	58.87	54.86	22.09

- Keep all tokens (the proposed holistic version of the SentiCode)
- Remove NOT token (to assess the impact of negation)
- Remove POS token (effect of prior positive polarity of bearer tokens)
- Remove NEG token (effect of prior negative polarity of bearer tokens)
- Remove POS and NEG tokens (effect of both prior positive and prior negative polarities)
- Remove POS, NEG, and NOT tokens (effect of prior polarities and handling the negation)

Figure 4 compares the effect of the proposed tokens (POS, NEG, and NOT) in SentiCode. The results show that processing negation and taking into account the prior polarity of the bearer tokens contribute to better accuracy. Similarly, negation processing (see the NOT token bar in Fig. 4) has a significant impact compared to POS and NEG tokens alone.

5.3 Cross-evaluation of SentiCode with other training corpora

Another question we raised during the experiments was: *What about training the models using other corpora than en.train.books?* To answer this question, a heat map in Fig. 5

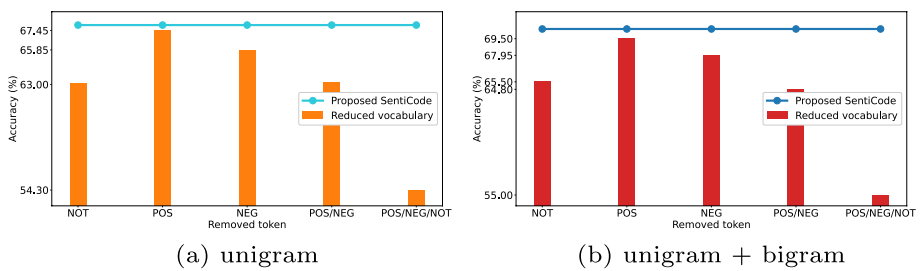


Fig. 4 The relevance of SentiCode vocabulary

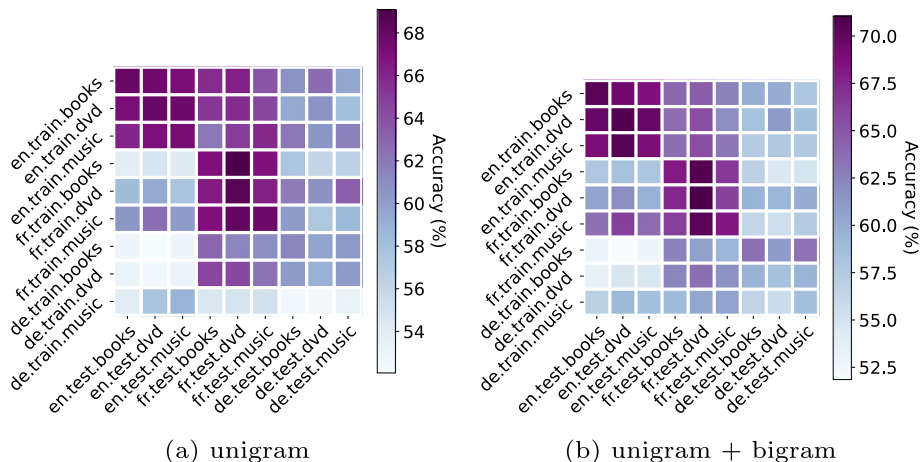


Fig. 5 The heatmap of cross-evaluation

shows the accuracy of the cross-language-domain consisting of 81 language-domain pairs. The results are different and depend on the training corpus. Thus, the results are better when using a corpus rich in terms of various patterns and handling negation. A rich corpus allows to train the model well and determine the discriminating features to differentiate the classes, as in this study with the en.train.books corpus. Figure 6 summarizes the token distribution for the additional corpora used in the cross-evaluation.

5.4 Trade-off between effectiveness and computing time

This subsection presents the assessment of SentiCode with three other methods in terms of performance, using the accuracy metric and time in seconds. Furthermore, we determine the winner method by classifying the trade-off between the accuracy and the computing time using the Balanced Integration Score (BIS) metric (Liesefeld & Janczyk, 2019). The method with the higher BIS performs better.

All evaluated methods use the SVM algorithm by applying the TF-IDF vector space with uni-gram as features on book domain corpora. The first method (trainFR__testFR) was tested using a classical approach by training and testing on the same language. In the

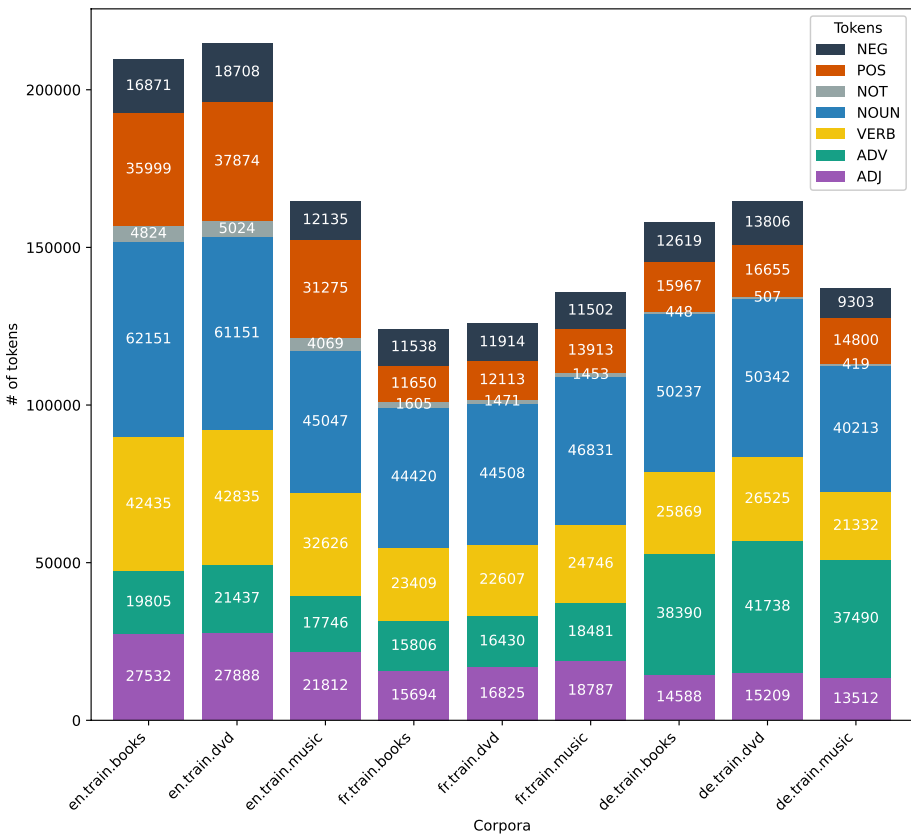


Fig. 6 Tokens distribution in SentiCode of the different training corpora. (The stacked values in bars have the same order in legend upper right)

second method (trainEN__testFR), the model was trained with English text and then tested directly (without translation) in the French language. While in the third method (trainEN__testTranslatedFRtoEN), we evaluated a model trained in English and tested with French text translated into English. The accuracy and total time (training time + prediction time) are highlighted in Table 10. Although SentiCode (SentiCode_En__FR) does not have a maximum performance (Acc. =65.65), it has by far the best total computing time (time =0.36s). Moreover, SentiCode has a good trade-off between performance and total time among other tested methods with BIS ≈ 1.44. It is important to note that the experiments were performed on a personal laptop: *Intel Core i7 4th generation CPU 2.10 GHz 2.70 GHz and 8Go of RAM*. The operation mentioned above was repeated for the German language, where German replaced French. Moreover, based on Table 10, we observe similar results for the German language. SentiCode (SentiCode_EN__DE) is by far faster than the other methods on German language and has the best trade-off with BIS ≈ 1.24. This good trade-off of SentiCode makes it suitable for real applications where speed plays a good role in such commercial applications.

5.5 Error analysis

This subsection, highlights some misclassified reviews found in the results. We showed the original text of the review with its equivalent SentiCode in addition to the actual and predicted class. Furthermore, we explained and proposed an adjustment at the level of SentiCode to reclassify the review correctly.

Table 10 Trade-off between accuracy and time

Langs	Methods	Time (s)	Acc.	BIS
French	trainFR__testFR	261.18	82.80%	0.577979316
	trainEN__testFR	264.01	50.25%	-2.011215883
	trainEN__testTranslatedFRtoEN	290.14	78.55%	-0.00224069
	SentiCode_EN__FR	0.36	65.65%	1.435477257
German	trainDE__testDE	376.15	82.40%	0.042101875
	trainEN__testDE	266.38	49.85%	-1.661086421
	trainEN__testTranslatedDEtoEN	298.04	79.50%	0.374142354
	SentiCode_EN__DE	0.67	63.30%	1.244842193

Table 11 Example 1 of misclassified English review

En.dvd.148	Sentence	Class
Review	it's a great day for da hood. i loved all three of em but unfortunately only own friday	positive
SentiCode	ADJ POS NOUN NOUN VERB ADV NEG ADV POS ADJ	negative
Explication	The verb loved has not had its prior polarity, thus decreasing the possibility to classify this review as positive	/
Adjustment	ADJ POS NOUN NOUN VERB POS ADV NEG ADV POS ADJ	Positive (correct)

Tables 11, 12, and 13 highlight three examples of misclassified reviews, one from each language. The added tokens are in **bold** in the adjustment row, while removed tokens are ~~crossed~~.

5.6 Application of SentiCode on a corpus with a combined language and domain

SentiCode is suitable for multilingual and multi-domain sentiment analysis and requires only one training session. After that, the trained model is used several times for other texts

Table 12 Example 2 of misclassified French review

Fr.books.33	Sentence	Class
Review	Très beau livre qui donne envie de tester toutes les recettes. Sommaire très pratique avec une photo de chaque recette.	positive
In English	Very nice book that makes you want to test all the recipes. Very practical summary with a photo of each recipe.	/
SentiCode	ADV NEG NOUN POS ADJ VERB NOUN VERB ADJ NOUN NOUN NEG ADV NEG NOUN POS NOUN NOUN	negative
Explication	The adverb <i>Très</i> (meaning <i>very</i> in English) was found in negative lexicon, where it is not convenient. Removing or ignoring it (since it has no effect on either positivity or negativity) helps to reclassify this review correctly.	/
Adjustment	ADV NEG NOUN POS ADJ VERB NOUN VERB ADJ NOUN NOUN NEG ADV NEG NOUN POS NOUN NOUN	Positive (correct)

Table 13 Example 3 of misclassified German review

De.books.49	Sentence	Class
Review	Ich finde dieses Buch nicht gerade Weltklasse. Wenn man anfängt zu lesen, ist man nicht gerade erpicht darauf weiter zu lesen. Jedoch ist es im allgemeinen ganz gut und spannend.	positive
In English	I don't think this book is exactly world class. When you start reading, you aren't exactly eager to read any further. However, it is generally quite good and exciting.	/
SentiCode	VERB NOUN ADV POS NOUN VERB VERB ADV POS ADV ADV ADV POS VERB ADV ADJ NEG ADV POS ADV POS ADV NEG	negative
Explication	Weak positive	/
Adjustment	None	/

written in different languages or domains. Equally important, SentiCode requires no translation before making a prediction; this saves time and effort.

This subsection presents the flexibility of SentiCode in multi-language-domain training/testing by combining the nine test corpora, en.test.books, en.test.dvd, en.test.music, fr.test.books, fr.test.dvd, fr.test.music, de.test.books, de.test.dvd, de.test.music into one called multi.test.all. Similarly, the nine train corpora, en.train.books, en.train.dvd, en.train.music, fr.train.books, fr.train.dvd, fr.train.music, de.train.books, de.train.dvd, de.train.music (see Fig. 2 for their statistics), were combined to create another corpus called multi.train.all. The multi.train.all corpus was used for training and the multi.test.all corpus was used for testing. The experiment was conducted on five classifiers: SVM, MNB, LR, SGDC, and MLP. Then, multi-training was evaluated in terms of n-grams to present the role of context. Figure 7a shows that using unigram only leads to low performance for all models used in the experiment, while giving more context, e.g., bigram, will improve accuracy. Figure 7b shows that 50% of the training corpus is sufficient to achieve stable performance. Another observation based on Fig. 7a and b is the performance of SVM; it is higher than the others. The same observation is made with SVM in (Zhang et al., 2019). Regarding the time consuming, SVM is the slowest, mainly with a large number of features, e.g., using an n-gram range of (2, 3). SVM took 90.69 seconds to train the model and 187.20 seconds to end prediction. Meanwhile, MNB, LR, SGDC, MLP took 2.31, 3.63, 2.67, 39.25 seconds in the training phase and 4.70, 6.35, 5.06, 39.32 seconds to end prediction.

5.7 SentiCode with non-western languages

After assessing SentiCode in western languages (English, French, and German), we assessed how SentiCode performs in non-western languages, taking Russian and Arabic as additional ones. For Arabic, we used the ASTD dataset (Nabil et al., 2015) to generate

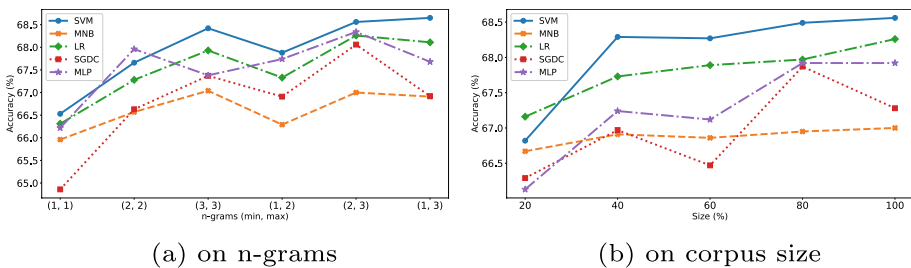


Fig. 7 The varied performance of SentiCode

Table 14 The different resources used in the generation of SentiCode

Languages	Datasets	PoS Tagger	Lexicons	Negation terms list
Russian	Araujo et al., (2016)	spaCy	Loukachevitch & Levchik, (2016)	нет, не, никто ничто, нигде, никогда
Arabic	Nabil et al., (2015)	NLTK ¹	Chen & Skiena, (2014)	لا، لم، لن

¹ NLTK library version 2.6.3

Table 15 cross-evaluation on western and non-western languages using SVM

Test on	Arabic			Russian			multi.test.all		
	Acc.	M-F1	MCC	Acc.	M-F1	MCC	Acc.	M-F1	MCC
Arabic	68.05	67.93	36.22	56.43	55.29	14.58	59.38	59.12	18.99
Russian	63.08	60.84	29.39	60.52	60.07	22.85	62.61	61.86	26.28
multi.train.all	59.82	55.88	23.94	62.25	62.16	25.00	66.54	66.51	33.16

Table 16 cross-evaluation on western and non-western languages using MLP

Test on	Arabic			Russian			multi.test.all		
	Acc.	M-F1	MCC	Acc.	M-F1	MCC	Acc.	M-F1	MCC
Arabic	70.29	70.24	40.60	59.73	59.69	19.76	60.17	58.22	22.53
Russian	64.04	62.33	30.69	62.38	61.83	26.35	64.68	64.45	29.73
multi.train.all	59.37	55.06	23.29	61.34	61.25	22.57	66.22	66.20	32.49

SentiCode, referred to as *ar.senticode*, with some pre-processing (Remove diacritization, lemmatization⁶). On the other side, we employed the dataset kindly provided by (Araujo et al., 2016) for the Russian language (Tweets in Russian, 1145 positive, 1188 negative, and 1635 neutral). The generated SentiCode from Russian is referred to *ru.senticode*. Besides, the resources used to generate SentiCode (following the Algorithm 1 presented before) from these two languages are presented in Table 14.

After, the cross-evaluations between Arabic, Russian and western languages (English, French, and German, i.e., multi.train.all for training and multi.test.all for testing) were conducted on SVM and MLP (with the same configuration as the previous experiment). The results are detailed in Tables 15 and 16 for each method, respectively. In these cross-evaluations, it is important to note that we removed the third class for Arabic and Russian (neutral reviews) to keep the consistency between data for the western language, which is labeled only on binary classes (positive and negative). We used the 80-20 rule for training and testing in Arabic to split the ar.senticode dataset (80% for training and 20% for testing). The same was carried out for the Russian language. We used the entire data for all pairs.

The results show good interoperability between western and non-western languages and are justly balanced. We recorded an accuracy of 64.04% for the experiment where the training was in Russian and the test in Arabic. Also, we recorded 64.68% of accuracy in the experiment with Russian as training data and western languages as testing data. Furthermore, we reported 64.04%, 62.33%, 30.69% as accuracy, M-F1, MCC for the Russian-Arabic pair, which is higher than the Russian-Russian pair.

5.8 SentiCode with non-binary classifications

Finally, we assessed the SentiCode on a dataset with ternary classification. In addition to ar.senticode and ru.senticode, we supported our dataset with a third one, generating

⁶ Use PyArabic 0.6.14 and qalsadi 0.4.5 tools

Table 17 SentiCode results on ternary classification

Models	Acc.	M-F1	MCC
MLP	54.89 ± 0.003	54.95 ± 0.003	32.56 ± 0.004
SVM	54.65 ± 0.004	54.62 ± 0.005	32.41 ± 0.006
LR	54.45 ± 0.007	54.29 ± 0.007	32.05 ± 0.010
SGDC	53.10 ± 0.005	52.65 ± 0.008	30.21 ± 0.008
MNB	52.11 ± 0.007	51.91 ± 0.007	28.33 ± 0.011

its SentiCode from (Potts et al., 2021) using the same resources used for English in Section 4.3, taking into consideration three classes: positive, negative, and neutral. This new SentiCode generated from the English language is called en.senticode. The final dataset used in the experiment for ternary classification is gathered from ar.senticode, ru.senticode and en.senticode to obtain a mixed dataset, namely mixed.senticode. The experiment is conducted on SVM, MNB, LR, SGDC, and MLP, with the same previous parameters and tuning. The evaluation used 10-fold cross-validation on global balanced data (4000 positives, 4000 negatives, and 4000 neutrals), reviewing the accuracy mean, macro f1 score, and MCC, and standard deviation. Table 17 lists the results obtained.

Table 17 shows that the results are highly comparable. Where MLP is more efficient and MNB is less efficient. The important thing also to notice is the consistency of results. We observe a very small standard deviation (1e-3) value resulting from 10-fold cross-validation.

6 Conclusion

This paper introduced SentiCode as a new sentiment language for analysis and domain adaptation of multilingual sentiment. This work is the first implementation of its vocabulary. The results are very consistent for multilingual sentiment analysis and domain adaptation; they reached 70% - Regarding the obtained results, SentiCode can be improved and needs further analysis. SentiCode is a newcomer in the field with strong competition and interest among researchers and could attract more attention for more experiments on other languages, domains, or further on its basic structure. Furthermore, one of our objectives in presenting SentiCode is to propose it as a baseline for comparison with other future approaches in the field of multilingual sentiment analysis, as we reported results in various metrics such as accuracy, f1-score, and MCC coupled with different tuning.

For future work, we propose two axes: first, the enhancement of the algorithm by applying more advanced techniques to handle negation and developing prior polarity detection. The second objective concerns the holistic aspect by applying SentiCode to other languages not tested in this paper. Finally, we will train and test it on multiclass classification. To conclude, SentiCode will help focus research on designing and training SentiCode rather than dispersing efforts on many languages. Therefore, we can use it directly for prediction without training for each language beforehand. Consequently, we will save time and effort. **Data availability** The datasets used to generate SentiCode during the current study are available in the Zenodo repository, <https://zenodo.org/record/3251672>.

The datasets of SentiCode generated and analysed during the current study are available in the figshare repository, <https://doi.org/10.6084/m9.figshare.17695559>.

Declarations

Conflict of Interests The authors declare that they have no conflict of interest.

References

- Abdullah, N. A. S., & Rusli, N. I. A. (2021). Multilingual sentiment analysis: A systematic literature review. *Pertanika. Journal of Science & Technology*, 29(1).
- Agüero-Torales, M. M., Salas, J. I. A., & López-Herrera, A. G. (2021). Deep learning and multilingual sentiment analysis on social media data: An overview. *Applied Soft Computing*, 107373.
- Ahuja, R., Chug, A., Kohli, S., Gupta, S., & Ahuja, P. (2019). The impact of features extraction on the sentiment analysis. *Procedia Computer Science*, 152, 341–348. International Conference on Pervasive Computing Advances and Applications- perCAA, 2019. <https://doi.org/10.1016/j.procs.2019.05.008>.
- Appel, O., Chiclana, F., Carter, J., & Fujita, H. (2016). A hybrid approach to the sentiment analysis problem at the sentence level. *Knowledge-Based Systems*, 108, 110–124. <https://doi.org/10.1016/j.knsys.2016.05.040>. New Avenues in Knowledge Bases for Natural Language Processing.
- Araujo, M., Reis, J., Pereira, A., & Benevenuto, F. (2016). An evaluation of machine translation for multilingual sentence-level sentiment analysis. In *Proceedings of the 31st annual ACM symposium on applied computing*, (pp. 1140–1145).
- Baccianella, S., Esuli, A., & Sebastiani, F. (2010). Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *Lrec*, (Vol. 10 pp. 2200–2204).
- Balahur, A., & Turchi, M. (2012). Multilingual sentiment analysis using machine translation?. In *Proceedings of the 3rd workshop in computational approaches to subjectivity and sentiment analysis*, (pp. 52–60). Association for Computational Linguistics. <https://aclanthology.org/W12-3709>.
- Balahur, A., & Turchi, M. (2014). Comparative experiments using supervised learning and machine translation for multilingual sentiment analysis. *Computer Speech & Language*, 28(1), 56–75. <https://doi.org/10.1016/j.csl.2013.03.004>.
- Banea, C., Mihalcea, R., & Wiebe, J. (2010). Multilingual subjectivity: Are more languages better?. In *Proceedings of the 23rd international conference on computational linguistics (Coling 2010)*, (pp. 28–36). Coling 2010 Organizing Committee. <https://aclanthology.org/C10-1004>.
- Banea, C., Mihalcea, R., Wiebe, J., & Hassan, S. (2008). Multilingual subjectivity analysis using machine translation. In *Proceedings of the 2008 conference on empirical methods in natural language processing*, (pp. 127–135). Association for Computational Linguistics. <https://aclanthology.org/D08-1014>.
- Berka, P. (2020). Sentiment analysis using rule-based and case-based reasoning. *Journal of Intelligent Information Systems*, 55(1), 51–66. <https://doi.org/10.1007/s10844-019-00591-8>.
- Blitzer, J., Dredze, M., & Pereira, F. (2007). Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, (pp. 440–447). <https://www.aclweb.org/anthology/P07-1056.pdf>.
- Boiy, E., & Moens, M.-F. (2009). A machine learning approach to sentiment analysis in multilingual web texts. *Information Retrieval*, 12(5), 526–558.
- Cambria, E. (2016). Affective computing and sentiment analysis. *IEEE Intelligent Systems*, 31(2), 102–107. <https://doi.org/10.1109/MIS.2016.31>.
- Cambria, E., Li, Y., Xing, F. Z., Poria, S., & Kwok, K. (2020). Senticnet 6: Ensemble application of symbolic and subsymbolic ai for sentiment analysis. In *Proceedings of the 29th ACM international conference on artificial intelligence and knowledge management*, (pp. 105–114).
- Chen, Z., Shen, S., Hu, Z., Lu, X., Mei, Q., & Liu, X. (2019). Emoji-powered representation learning for cross-lingual sentiment classification. In *The world wide web conference. WWW '19*, (pp. 251–262). Association for Computing Machinery. <https://doi.org/10.1145/3308558.3313600>.
- Chen, Y., & Skiena, S. (2014). Building sentiment lexicons for all major languages. In *Proceedings of the 52nd annual meeting of the association for computational linguistics (Volume 2: Short Papers)*, (pp. 383–389). Association for Computational Linguistics. <https://doi.org/10.3115/v1/p14-2063>.
- Chicco, D., & Jurman, G. (2020). The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21(1), 1–13.
- Cruz Paulino, J. L., Antoja Almirol, L. C., Cruz Favila, J. M., Loria Aquino, K. A. G., Hernandez De La Cruz, A., & Roxas, R. E. (2021). *Multilingual sentiment analysis on short text document using*

- semi-supervised machine learning*, (pp. 164–170). Association for Computing Machinery. <https://doi.org/10.1145/3485768.3485775>.
- Cui, A., Zhang, M., Liu, Y., & Ma, S. (2011). Emotion tokens: Bridging the gap among multilingual twitter sentiment analysis. In *Asia Information Retrieval Symposium*, (pp. 238–249). Springer.
- Do, H. H., Prasad, P., Maag, A., & Alsadoon, A. (2019). Deep learning for aspect-based sentiment analysis: a comparative review. *Expert Systems with Applications*, 118, 272–299. <https://doi.org/10.1016/j.eswa.2018.10.003>.
- Esuli, A., & Sebastiani, F. (2006). Sentiwordnet: A publicly available lexical resource for opinion mining. In *LREC*, (Vol. 6 pp. 417–422). Citeseer.
- Ghorbel, H. (2012). Experiments in cross-lingual sentiment analysis in discussion forums. In K. Aberer, A. Flache, W. Jager, L. Liu, J. Tang, & C Gu  ret (Eds.) *Social informatics* (pp. 138–151). Springer.
- Hatzivassiloglou, V., & Wiebe, J.M. (2000). Effects of adjective orientation and gradability on sentence subjectivity. In *COLING '00*, (pp. 299–305). Association for Computational Linguistics. <https://doi.org/10.3115/990820.990864>.
- Hutto, C., & Gilbert, E. (2014). Vader: a parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the international AAAI conference on web and social media*, (Vol. 8)
- Jagdale, R. S., Shirsat, V. S., & Deshmukh, S. N. (2019). Sentiment analysis on product reviews using machine learning techniques. In *Cognitive informatics and soft computing* (pp. 639–647). Springer.
- Kanfoud, M. R., & Bouramoul, A. (2022). Linking the linguistic resources using graph structure for multilingual sentiment analysis. In X. -S. Yang, S. Sherratt, N. Dey, & A Joshi (Eds.) *Proceedings of sixth international congress on information and communication technology* (pp. 973–982). Springer.
- Li, W., Shao, W., Ji, S., & Cambria, E. (2022). Bieru: Bidirectional emotional recurrent unit for conversational sentiment analysis. *Neurocomputing*, 467, 73–82.
- Liang, B., Su, H., Gui, L., Cambria, E., & Xu, R. (2022). Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. *Knowledge-Based Systems*, 107643, 235.
- Liesefeld, H.R., & Janczyk, M. (2019). Combining speed and accuracy to control for speed-accuracy trade-offs(?). *Behavior Research Methods*, 51(1), 40–60. <https://doi.org/10.3758/s13428-018-1076-x>.
- Lo, S. L., Cambria, E., Chiong, R., & Cornforth, D. (2017). Multilingual sentiment analysis: from formal to informal and scarce resource languages. *Artificial Intelligence Review*, 48(4), 499–527.
- Londhe, D. D., Kumari, A., & Emmanuel, M. (2021). Challenges in multilingual and mixed script sentiment analysis. In *2021 6Th international conference for convergence in technology (i2CT)*. (pp. 1–6). <https://doi.org/10.1109/I2CT51068.2021.9418087>.
- Loukachevitch, N., & Levchik, A. (2016). Creating a general Russian sentiment lexicon. In *Proceedings of the tenth international conference on language resources and evaluation (LREC'16)*, (pp. 1171–1176). European Language Resources Association (ELRA). <https://aclanthology.org/L16-1186>.
- Lu, B., & Tsou, B.K. (2010). Combining a large sentiment lexicon and machine learning for subjectivity classification. In *2010 International conference on machine learning and cybernetics*, vol. 6, (pp. 3311–3316). IEEE.
- Nabil, M., Aly, M., & Atiya, A. (2015). ASTD: Arabic Sentiment tweets dataset. In *Proceedings of the 2015 conference on empirical methods in natural language processing. Association for Computational Linguistics*. <https://doi.org/10.18653/v1/d15-1299>.
- Nguyen, H. T., & Le Nguyen, M. (2018). Multilingual opinion mining on youtube—a convolutional n-gram bilstm word embedding. *Information Processing & Management*, 54(3), 451–462.
- Nivre, J., de Marneffe, M. -C., Ginter, F., Haji  , J., Manning, C. D., Pyysalo, S., Schuster, S., Tyers, F., & Zeman, D. (2020). Universal dependencies v2: An evergrowing multilingual treebank collection. arXiv:2004.10643
- Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on empirical methods in natural language processing*, (Vol. 10 pp. 79–86). Association for Computational Linguistics.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., & et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Polanyi, L., & Zaenen, A. (2006). Contextual valence shifters. In J.G. Shanahan, Y. Qu, & J. Wiebe (Eds.) *Computing attitude and affect in text: theory and applications*. https://doi.org/10.1007/1-4020-4102-0_1 (pp. 1–10). Dordrecht: Springer.
- Potts, C., Wu, Z., Geiger, A., & Kiela, D. (2021). Dynasent: A dynamic benchmark for sentiment analysis. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing*, (Volume 1: Long Papers), (pp. 2388–2404). Association for Computational Linguistics, Online. <https://doi.org/10.18653/v1/2021.acl-long.186>.

- Prettenhofer, P., & Stein, B. (2010). Webis Cross-Lingual Sentiment Dataset 2010 (Webis-CLS-10). Zenodo. <https://doi.org/10.5281/zenodo.3251672>.
- Prettenhofer, P., & Stein, B. (2010). Cross-language text classification using structural correspondence learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, (pp. 1118–1127).
- Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys (CSUR)*, 34(1), 1–47.
- Singh, L. G., & Singh, S. R. (2021). Empirical study of sentiment analysis tools and techniques on societal topics. *Journal of Intelligent Information Systems*, 56(2), 379–407.
- Stone, P. J., Dunphy, D. C., & Smith, M. S. (1966). *The general inquirer: a computer approach to content analysis*. MIT Press.
- Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2), 267–307.
- Tripathy, A., Agrawal, A., & Rath, S. K. (2016). Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*, 57, 117–126. <https://doi.org/10.1016/j.eswa.2016.03.028>.
- Vilares, D., Alonso, M.A., & Gómez-Rodríguez, C. (2015). Sentiment analysis on monolingual, multilingual and code-switching Twitter corpora. In *Proceedings of the 6th workshop on computational approaches to subjectivity, sentiment and social media analysis*, (pp. 2–8). Association for Computational Linguistics. <https://doi.org/10.18653/v1/W15-2902>. <https://aclanthology.org/W15-2902>.
- Wiegand, M., Wolf, M., & Ruppenhofer, J. In G. Rehm T Declerck (Eds.) (2018). *Negation modeling for german polarity classification*. Cham: Springer.
- Wilson, T., Wiebe, J., & Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, (pp. 347–354).
- Yekrang, M., & Abdolvand, N. (2021). Financial markets sentiment analysis: Developing a specialized lexicon. *Journal of Intelligent Information Systems*, 57(1), 127–146.
- Zhang, Y., Song, D., Zhang, P., Li, X., & Wang, P. (2019). A quantum-inspired sentiment representation model for twitter sentiment analysis. *Applied Intelligence*, 49(8), 3093–3108.
- Zhou, X., Wan, X., & Xiao, J. (2016). Cross-lingual sentiment classification with bilingual document representation learning. In *Proceedings of the 54th annual meeting of the association for computational linguistics*, (Volume 1: Long Papers), (pp. 1403–1412). Association for Computational Linguistics. <https://doi.org/10.18653/v1/P16-1133>. <https://aclanthology.org/P16-1133>.
- Štajner, T., Novalija, I., & Mladenčić, D. (2013). Informal multilingual multi-domain sentiment analysis. *Informatica*, 37(4).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.