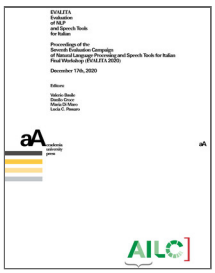

Valerio Basile, Danilo Croce, Maria Maro and Lucia C. Passaro (dir.)

EVALITA Evaluation of NLP and Speech Tools for Italian - December 17th, 2020
Proceedings of the Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian Final Workshop

Accademia University Press

DeepReading @ SardiStance: Combining Textual, Social and Emotional Features

María S. Espinosa, Rodrigo Agerri, Alvaro Rodrigo and Roberto Centeno

DOI: 10.4000/books.aaccademia.7129
Publisher: Accademia University Press
Place of publication: Torino
Year of publication: 2020
Published on OpenEdition Books: 11 May 2021
Serie: Collana dell'Associazione Italiana di Linguistica Computazionale
Electronic ISBN: 9791280136329

DeepReading @ SardiStance: Combining Textual, Social and Emotional Features

María S. Espinosa
NLP & IR Group
UNED, Spain
mespinosa@lsi.uned.es

Rodrigo Agerri
HiTZ Center - Ixa
University of the Basque Country UPV/EHU
rodrigo.agerri@ehu.eus

Alvaro Rodrigo
NLP & IR Group
UNED, Spain
alvarory@lsi.uned.es

Roberto Centeno
NLP & IR Group
UNED, Spain
rcenteno@lsi.uned.es

Abstract

In this paper we describe our participation to the SardiStance shared task held at EVALITA 2020. We developed a set of classifiers that combined text features, such as the best performing systems based on large pre-trained language models, together with user profile features, such as psychological traits and social media user interactions. The classification algorithms chosen for our models were various monolingual and multilingual Transformer models for text only classification, and XGBoost for the non-textual features. The combination of the textual and contextual models was performed by a weighted voting ensemble learning system. Our approach obtained the best score for Task B, on Contextual Stance Detection.

1 Introduction

One of the most important research topics in the field of Natural Language Processing (NLP) is automatic information extraction from textual data. The recent rise of social media has completely changed the way in which people communicate their ideas and has thus led to the emergence of new research problems regarding the automatic analysis of online contents, such as sentiment analysis, emotion recognition, or fake news detection. Stance detection (usually considered as a subproblem of sentiment analysis) is part of the aforementioned family of research problems (Küçük and Can, 2020). While there are various formulation of the stance detection task, for SardiStance 2020 the aim is to detect the stance (AGAINST, FAVOR or NEUTRAL) conveyed by a given tweet with respect to a specific, previously given topic (Mohammad et al., 2016), namely, about the Sardines movement in Italy.

Thus, we address the problem of automatic stance detection in tweets written in Italian language for the SardiStance 2020 shared task (Cignarella et al., 2020), organized within EVALITA 2020 (Basile et al., 2020). In this paper we include the participation of three teams within the framework of the DeepReading project [1]: (1) Ixa Group, (2) UNED group, and (3) DeepReading Group. While Ixa focused on developing text classifiers based on textual information only (Task A), UNED was more interested in exploring how to use contextual information available (Task B). Likewise, DeepReading is the product of combining both Ixa and UNED systems into one.

In this sense, the main idea behind our model is to exploit textual information, based on fine-tuning large pre-trained language models for text classification, together with contextual information using several feature categories, such as psychological traits of the user, social media data, and network based features. As a result of our joint effort, we submitted 4 and 5 runs, respectively, to tasks A and B. The official results show that our systems obtained the 3rd position among the constrained runs submitted to Task A, which considered only textual information for prediction, and 1st position from 13 participants for Task B, which considered textual and contextual information.

[1] http://ixa2.si.ehu.es/deepreading/

2 Systems Description

In this section we first describe the text classification systems developed for Task A and then the contextual features used to train XGBoost classifiers for Task B. We also include a description of the strategies used to combine the classifiers from both tasks, which resulted in the winner system for Task B.

2.1 Task A: Textual Stance Detection

The main objective of our participation in Task A was to benchmark the performance, on the stance detection task for Italian, of large pre-trained language models based on the transformer architecture (Vaswani et al., 2017). This would help us to identify the best performing models which will be leveraged to generate features for Task B (Contextual Stance Detection).

As for many other Natural Language Processing (NLP) tasks, current best performing systems for text classification are based on large pre-trained language models which allow to build rich representations of text based on contextual word embeddings. Deep learning methods in NLP represent words as continuous vectors on a low dimensional space, called word embeddings. The first approaches generated static word embeddings (Mikolov et al., 2013; Bojanowski et al., 2017), namely, they provided a unique vector-based representation for a given word independently of the context in which the word occurs. This means that polysemy cannot be represented.

In order to address this problem, contextual word embeddings were proposed. The idea is to be able to generate word representations according to the context in which the word occurs. Currently there are many approaches to generate such contextual word representations, but we will focus on publicly available multilingual and monolingual pre-trained models for Italian.

There are several multilingual versions of these models. Thus, the multilingual version of BERT (Devlin et al., 2019) was trained for the top 100 languages with the largest Wikipedias. More recently, XLM-RoBERTa (Conneau et al., 2019) distributes a multilingual model which contains 104 languages trained on 2.5 TB of Common Crawl data. Italian is included in both multilingual models.

These multilingual models perform very well in tasks involving high-resourced languages such as English or Spanish, but their performance drops when applied to languages not so well represented in the language model (Agerri et al., 2020). Although this is still an open issue, a number of reasons can be found in the literature. First, each language has to share the quota of substrings and parameters with the rest of the languages represented in the pre-trained multilingual model. As the quota of substrings partially depends on corpus size, this means that larger languages such as English or Spanish are better represented than other languages such as Italian. Moreover, multilingual models also seem to behave better for structurally similar languages (Karthikeyan et al., 2020).

We have benchmarked four monolingual pre-trained language models for Italian: AlBERTo, GilBERTo, UmBERTo and Italian BERT XXL with the aim of comparing them with respect to the multilingual pre-trained models previosly mentioned, namely, mBERT and XLM-RoBERTa.

AlBERTo is a BERT *base* pre-trained lowercased model containing a vocabulary of 128k terms from 200M of Italian tweets (Polignano et al., 2019).

The Italian BERT XXL models [2] are also based on the BERT *base* architecture. The training data contains the Italian Wikipedia, various parts of the OPUS corpus and the OSCAR corpus for Italian (Ortiz Suárez et al., 2019), for a total of 81GB of Italian text.

GilBERTo[3] is based on the RoBERTa *base* (Liu et al., 2019) architecture, an improved, optimized version of BERT which discards the next sentence prediction task. The model was trained using the Italian Oscar (Ortiz Suárez et al., 2019), which contains 71GB of text. The vocabulary used consisted of 32k BPE subwords tokenized by the SentencePiece tokenizer[4].

UmBERTo[5] also leverages the RoBERTa *base* architecture, the OSCAR corpus for Italian and the SentencePiece tokenizer, but it adds Whole Word Masking to the training process. The idea is to mask an entire word, instead of subwords, if at least one of all (sub-)tokens generated by SentencePiece was originally selected as mask.

[2] https://github.com/dbmdz/berts
[3] https://github.com/idb-ita/GilBERTo
[4] https://github.com/google/sentencepiece
[5] https://github.com/musixmatchresearch/umberto

2.2 Task B: Contextual Stance Detection

In this task, we use several sets of features with the purpose of trying to model user's behaviour when writing a tweet. We obtain such features from both the text and the social network. Our hypothesis is that the stance of a user regarding a particular tweet is highly correlated with the way of writing of the own user extracted in terms of psychological and emotional features. On the other hand, we focus on exploring how the concept of "homophily", namely, the tendency of individuals to associate and bond with similar individuals, previously studied in DellaPosta et al. (2015). In order to test this hypothesis, we have tested different models that are explained below.

In this task, we use several sets of features with the purpose of trying to model user's behaviour when writing a tweet. We obtain such features from text and the network.

The complete set of features extracted from the data is depicted in Table 1. The set of features used in the model can be divided into five main types: psychological, emotional, Twitter-based, network-based, and language model features.

Category	Feature name	Description
Psychological features	pers_pred	personality prediction
	self_pred	self-revealing prediction
	info_pred	information-seeking prediction
	action_pred	action-seeking prediction
	fact_pred	fact-oriented prediction
Emotion freatures	arousal	mean arousal value
	valence	mean valence value
	russell	emotion value on Russell's model
Twitter features	statuses_count	number of tweets posted by user
	friends_count	number of following users
	followers_count	number of follower users
	created_at	account creation date
Network features	d_favor	mean distance to users in favor
	d_against	mean distance to users against
	d_none	mean distance to neutral users
Language model features	p_favor	prob. of tweet being in favor
	p_against	prob. of tweet being against
	p_none	prob. of tweet being neutral

Table 1: Complete set of features extracted from the data.

Psychological features. These features were extracted using a third-party API developed by Symanto[6]. Each tweet was sent to the API in order to retrieve the personality traits and communication styles obtained from the analysis of the tweet contents.

The personality traits value would be either "emotional" or "rational" depending on the analysis of the user's text. The value returned by the API when the communication styles are requested is a collection of traits, such as *self-revealing*, which means sharing one's own experience and opinion; *fact-oriented*, which implies focusing on factual information, objective observations or statements; *information-seeking*, that is, posing questions; and *action-seeking* or aiming to trigger someone's action by giving recommendation, requests or advice.

Emotional features. In order to retrieve the emotion values from the tweets, we used Russell's circumplex model of affect (Russell, 1980). Russell argues that emotions can be conceptualized in a two-dimensional continuous space where the axes correspond to the degree of arousal and valence (or pleasure). These two dimensions form a Cartesian space that can be configured in a circular order in which the different combinations of valence and arousal correspond to one of four discrete emotion regions: tired, tense, excited, and pleased.

The values for the degree of arousal and valence of the tweets were obtained using an adaptation to Italian language of the Affective Norms for English Words (ANEW) (Bradley and Lang, 1999). This database was developed from translations of the 1,034 English words present in the ANEW dictionary and from words taken from Italian semantic norms (Montefinese et al., 2014).

Twitter features. Exploring how the users behave in the social network could offer some insights on the stance tendency of the users. The collection of Twitter data of each user contained four features: the number of statuses published by the user, the number of users followed by the user, the number of users following the user, and the creation date of the Twitter account of the user.

Network features. Using the `FRIEND.csv` data provided, we built a network consisting of 669817 nodes (or users) and 2847197 edges (or relationships) in order to represent the *following* network of the users. From that network, we extracted a sub-graph containing the users of known stance from the training data and the users involved in testing in order to calculate the mean distances of each user to the rest of known stance users using the following formula:

$$d_T(n) = \frac{\sum_{i=1}^{|T|} \frac{1}{d^2_{n \to i}}}{|T|}$$

where $|T|$ is the total number of users of a determined stance (AGAINST, FAVOR, NONE) and

[6]https://symanto-research.github.io/symanto-docs/

Team	Model	Rank	$\mathbf{F1}_{avg}$	$\mathbf{F1}_{Against}$	$\mathbf{F1}_{Favour}$	$\mathbf{F1}_{None}$
DeepReading	Italian BERT XXL	3	66.21	75.80	56.63	42.13
Ixa	UmBERTo	4	64.73	76.16	53.30	38.88
Ixa	GilBERTo	6	61.71	75.43	48.00	36.75
DeepReading	XML-RoBERTa	8	60.04	69.66	50.42	39.16
-	baseline	12-13	57.84	71.58	44.09	27.64

Table 2: Official Results for Task A.

$d^2_{n\to i}$ corresponds to the square distance in users from node n to node i. From this calculation we obtained 3 values per user: mean distance to users against ($d_{against}$), mean distance to users in favor (d_{favor}), and mean distance to neutral users (d_{none})

Language model features. In order to incorporate the language model results into the rest of the features of the system we choose the best performing, at the development phase, of the models described in Section 2.1, which was UmBERTo. Since this kind of language models use a great amount of features for learning and training, the strategy used in order to incorporate the language model without having a great imbalance in the number of features representing each category, consisted in extracting the probabilities assigned by the model to each class for each tweet. In this way, the language model would be present in 3 of the 18 features of the model, and it would therefore have a balanced size with regards to the rest of features of the model.

3 Results

3.1 Task A

As we use the base version of every transformer model we can fine-tune them in a basic GPU of 12GB RAM. Hyperparameter tuning (batch size, maximum sequence length, learning rate and number of epochs) was performed on the development set. For mBERT, AlBERTo, Italian BERT XXL and UmBERTo the best configuration was: maximum sequence length 256, batch 32, learning rate 5e-5, and 5 epochs. For GilBERTo we used the same values except the number of epochs, which was increased to 10. Finally, the best performing hyperparameters for XLM-RoBERTa was the following: maximum sequence length 256, batch 16, learning rate 2e-5, and 10 epochs.

While the monolingual models clearly outperformed both mBERT and XLM-RoBERTa on the development data, we decided to submit the three best monolingual runs and the best multilingual one. Table 2 reports the official results obtained by each of the models and their position with respect to the ranking of *constrained* runs for Task A released by the task organizers. Our submission based on Italian BERT XXL was clearly the best of our four runs, although its performance was around 1.5 scores in F1 lower than the winner system for Task A. Furthermore, the ranking obtained in the test does not correspond with the results obtained during the development phase, where UmBERTo outperformed the other monolingual models by more than 3 points in F1 score.

3.2 Task B

We presented a total of five models to Task B, which consisted of different combinations of the features listed in Table 1.

Models 1, 2, and 3. During the training and development phases of the models, several configurations were tested on models 1, 2, and 3, including training with different classifiers, such as Random Forest Classifier, Decision Tree Classifier and XGBoost Classifier. The best performing classifier was XGBoost configured for multi-class classification and taking into account class weights in order to deal with the imbalance present in the data. XGBoost is an efficient and scalable implementation of gradient boosting framework by (Friedman, 2001). With regards to the set of features, the first approach to the task considered only psychological, emotion, and Twitter features. For the second model, network features were added to the feature set. Finally, model 3 considered the probabilities of each class (AGAINST, FAVOR, NONE) predicted by the UmBERTo language model as three additional features for training.

Models 4 and 5. These two models were constructed using voting based ensemble learning. The voting system for model 4 considered predictions of models 1, 2, and 3 as well as predictions by the best performing language models on

Team	Model	Rank	$\mathbf{F1}_{avg}$	$\mathbf{F1}_{Against}$	$\mathbf{F1}_{Favour}$	$\mathbf{F1}_{None}$
Ixa	Model 5	1	74.45	85.62	63.29	42.14
DeepReading	Model 3	3	72.30	83.68	60.93	33.64
DeepReading	Model 4	4	72.22	83.00	61.43	42.51
UNED	Model 2	7	68.88	81.75	56.00	24.55
-	baseline	10-11	62.84	76.72	48.95	30.09
UNED	Model 1	13	53.13	73.99	32.26	20.00

Table 3: Ranking results of model 1 to 5 in task B of the competition.

the development data: UmBERTo, GilBERTo, and Italian BERT XXL, described in Section 2.1. The most common predicted value among the 6 systems was chosen as the final prediction of model 4. In case of having two or more values with the same counts, the final value is randomly selected. On the other hand, model 5 used a weighted voting ensemble learning in which each of the systems considered had as weight the F1 value obtained on the development data. Therefore, the model considered the weighted predictions of each system in order to choose the final prediction.

Table 3 shows the official results obtained by each model and their position with respect to the ranking for Task B on Contextual Stance Detection. As it can be noted, model 5 ranked first in this task, obtaining an average F1 of 0.7445. Models 3 and 4 also had promising results in the official test set, ranking third and fourth, respectively, and just 0.0079 below the system which obtained the second best result. Model 2 had a slightly worse performance, ranking seventh from a total of 13, but still 0.0604 above the baseline. Finally, model 1 had the lowest performance, ranking last for the task.

4 Discussion

Figure 1 shows the confusion matrices obtained from the released gold test data for each of the five runs submitted to task B. As it can be noticed, the performance of each model is increasingly better from the first to the fifth, as new features are added to them. The biggest increase, especially with respect to false positives in the AGAINST class, takes place from model 1 to model 2, that is, with the inclusion of network features into the model. This indicates that considering contextual information for stance detection tasks, such as the stance of those who are part of the friendship network of the user, can help determine their stance more accurately.

Furthermore, we can see that predictions from model 3 also experimented a great increase in true positives of each of the classes. This increase is related to the inclusion of the language model into the features of model 2, which demonstrates the importance of textual data in stance detection tasks.

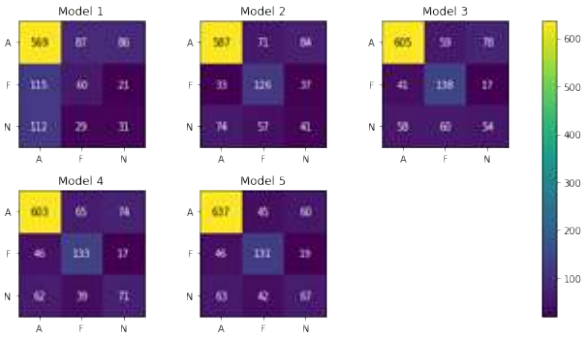

Figure 1: Confusion matrices for models 1 to 5 on test data.

Finally, models 4 and 5 shows the adequacy of combining several complementary systems in order to improve results. Since each single model can detect the stance for different instances, a proper combination of them could outperform single models.

5 Conclusions and Future Work

In this paper we have shown the benefits of exploiting information from different and heterogeneous sources. For our participation to the SardiStance 2020 shared task we have experimented with classifiers trained with the textual content of the tweets as well as with features based on social networks. This combination of features has allowed us to obtain the best overall results in the task.

As future work, we plan to further explore the contribution of network information. Besides, we want to develop new divergent models and study how to combine them.

Acknowledgments

This work has been partially funded by the Spanish Ministry of Science, Innovation and Universities (DeepReading RTI2018-096846-B-C21, MCIU/AEI/FEDER, UE), and DeepText (KK-2020/00088), funded by the Basque Government. Rodrigo Agerri is additionally funded by the RYC-2017-23647 fellowship and acknowledges the donation of a Titan V GPU by the NVIDIA Corporation. Maria S. Espinosa is also funded by the European Social Fund through the Youth Employment Initiative (YEI 2019).

References

[Agerri et al.2020] Rodrigo Agerri, Iñaki San Vicente, Jon Ander Campos, Ander Barrena, Xabier Saralegi, Aitor Soroa, and Eneko Agirre. 2020. Give your text representation models some love: the case for basque. In *LREC 2020*, pages 4781–4788.

[Basile et al.2020] Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro. 2020. EVALITA 2020: Overview of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. In Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro, editors, *EVALITA 2020*. CEUR-WS.org.

[Bojanowski et al.2017] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *TACL*, 5:135–146.

[Bradley and Lang1999] Margaret M Bradley and Peter J Lang. 1999. Affective norms for english words (anew): Instruction manual and affective ratings. Technical Report 1, Technical report C-1, the center for research in psychophysiology, University of Florida.

[Cignarella et al.2020] Alessandra Teresa Cignarella, Mirko Lai, Cristina Bosco, Viviana Patti, and Paolo Rosso. 2020. SardiStance@EVALITA2020: Overview of the Task on Stance Detection in Italian Tweets. In Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro, editors, *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org.

[Conneau et al.2019] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv:1911.02116*.

[Devlin et al.2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT 2019*, pages 4171–4186.

[Friedman2001] Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

[Karthikeyan et al.2020] K Karthikeyan, Zihan Wang, Stephen Mayhew, and Dan Roth. 2020. Cross-lingual ability of multilingual bert: An empirical study. In *International Conference on Learning Representations (ICLR)*.

[Küçük and Can2020] Dilek Küçük and Fazli Can. 2020. Stance detection: A survey. *ACM Computing Surveys (CSUR)*, 53(1):1–37.

[Liu et al.2019] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

[Mikolov et al.2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

[Mohammad et al.2016] Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. SemEval-2016 task 6: Detecting stance in tweets. In *SemEval-2016)*, pages 31–41.

[Montefinese et al.2014] Maria Montefinese, Ettore Ambrosini, Beth Fairfield, and Nicola Mammarella. 2014. The adaptation of the affective norms for english words (anew) for italian. *Behavior research methods*, 46(3):887–903.

[Ortiz Suárez et al.2019] Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pages 9–16.

[Polignano et al.2019] Marco Polignano, Pierpaolo Basile, Marco de Gemmis, Giovanni Semeraro, and Valerio Basile. 2019. AlBERTo: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets. In *Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2019)*, volume 2481. CEUR.

[Russell1980] James A Russell. 1980. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161.

[Vaswani et al.2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.