Hate Speech and Stereotypes with Artificial Neural Networks

Giulio Biondi[1], Valentina Franzoni[1,2(✉)], Alessio Mancinelli[1], Alfredo Milani[1], and Rajdeep Niyogi[3]

[1] Department of Mathematics and Computer Science, University of Perugia, Perugia, Italy
{giulio.biondi,alfredo.milani}@unipg.it, valentina.franzoni@dmi.unipg.it, alessio.mancinelli@studenti.unipg.it
[2] Department of Computer Science, Hong Kong Baptist University, Kowloon, Hong Kong, China
[3] IIT Indian Institute of Technology at Roorkee, Roorkee, India
rajdeep.niyogi@cs.iitr.ac.in

Abstract. Information technology is ubiquitously integrated into all areas of human and social life. It becomes progressively critical to build trust in systems while exposing their limitations along with utility and values. The harmonic integration of applications into society will promote the ability of the individuals to positively adapt to change (resilience and response) only if, instead of imposing an AI-centric world on humans, the central goal is to rearrange AI to environments around all aspects of human life. Current AI architectures and applications are increasingly designed taking into account ethical issues, to support the educative role of advanced research tools in improving the interaction between individuals and ultimately in the betterment of society. This work aims at detecting hate speech and stereotypes in textual communication using Artificial Neural Networks and Natural Language Processing. Starting from data from the Evalita 2020 competition, we analyse 6851 tweets which include stereotypes and hate speech, including text and emoticons. Training our neural network with the Adam optimizer, we obtain very promising results, of accuracy for the hate speech task and the stereotype prediction task.

Keywords: NLP · Affective computing · Ethical computing

1 Introduction

The debate over free speech, stereotypes and hate speech, and legislation with international protocols to be applied to digital social networks is a topic of broad interest and still without consensus on a solution [11]. In social media and in all communication channels that involve relatively short texts, including the headlines of newspaper articles, the diffusion of extremist ideas is favoured by the

This work has been supported by MIUR PRIN PHRAME Project, and EmoRe.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022
O. Gervasi et al. (Eds.): ICCSA 2022 Workshops, LNCS 13379, pp. 15–32, 2022.
https://doi.org/10.1007/978-3-031-10545-6_2

rapid mode of communication that does not promote depth and argumentation [9,17,18]. Extremist ideas are often exploited by politicians and journalists with click-bait marketing models. Hate speech is usually referred to lack of politically correct attitude or animated conduct where an individual or a community, often characterized by wide cultural differences, is denigrated on the basis of one of its characteristics, such as the not universally accepted concept of the human race, or skin colour, national or geographical origin, sex or gender identity, disability or health status, religious faith or beliefs, ideological orientations. While countries and companies decide what to consider a shared undesirable outcome, AI research is developing techniques to enhance the detection of hate speech and stereotypes.

Techniques based on Natural Language Processing (NLP) and Artificial Neural Networks (ANN) are among the most promising methods for stereotypes and hate speech detection [1,3,5,15]. This work considers and tests ANN and NLP on the data of the Evalita 2020 competition [6,12] for supervised classification including 6851 tweets labelled for stereotypes and hate speech. The added value of the Evalita data sets is to provide text in Italian [14], while most of the data sets for research on hate speech and stereotypes are in English only [4,8,10]. Non-English languages are being taken into consideration recently for the same classification tasks [7,13,16] because of the relevance and criticality of the issue. After a preprocessing phase, we proceed to split data into training and test sets and the proposed novel ANN prototype is tested obtaining very promising results for hate speech and good results for stereotype detection in the Italian language.

2 Architecture of the Proposed Solution

The solution we propose and discuss has been developed in different stages, see workflow in Fig. 1, where rectangles represent data, and oval shapes represent activities:

- Data Collection, data acquisition and preliminary transformation (e.g., emoticons to text)
- Data Preprocessing: cleaning, filtering, features, word embedding computation
- Data set creation & split: training set and test set split
- Training: ANNs model setup, training and generation of trained model
- Testing: application of trained model to test set
- Results analysis

2.1 Data Set Source and Preprocessing

The original data set is from the Evalita 2020 competition [6] proposing different NLP and speech challenging tasks in the Italian language. The Evalita 2020 data set consists of Twitter messages targeting specific social groups prone to stereotypes, e.g., foreigners, people of minority religions in Italy, and nomads, collected using specific keywords. Such minorities are often at the heart of the

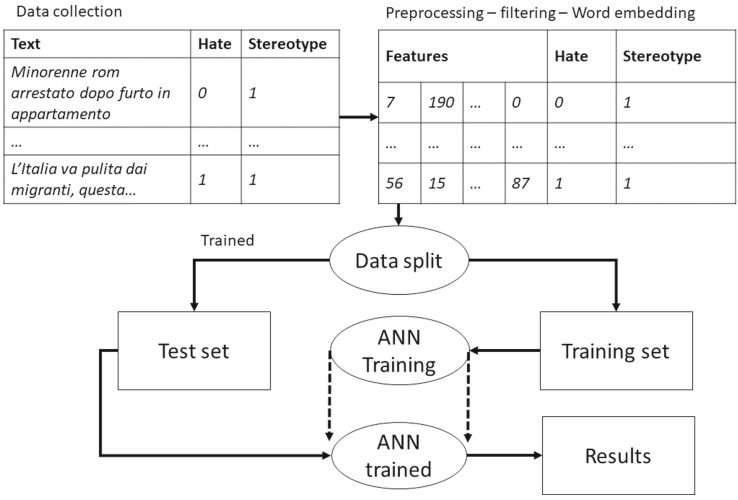


Fig. 1. Architectural flow of the solution

public social and political debate, and target of hate speech. The data set is available in a tab-separated format and contains 6851 tweets, with the following hate speech/stereotypes distribution:

- 4074 with no feelings of hate
- 2777 with feelings of hate
- 3804 with no stereotypes
- 3047 with stereotypes.

We decided to avoid balancing classes because in our case the loss due to eliminating terms is greater than the advantage of balancing them. In the data set, we consider the tweet text and the tags about *hate speech (hs)* and *stereotype*. Before transforming text into a format suitable for training the model, we exploit a preprocessing phase filtering put noisy non-meaningful words. We, therefore, eliminate some specific strings within the text, e.g. URLs or *@user* tags, and stop-words in the Italian language (i.e. prepositions, articles, possessive, demonstrative and interrogative adjectives, conjunctions). Usually, in NLP tasks, proper names and location names are also not taken into consideration. However, in our case, they can prove meaningful, e.g. Rome as the Italian seat of the government, or names/surnames of politicians involved in the public debate; therefore, we decided not to filter them out. Moreover, Particular attention has been paid to emoticons (i.e., the symbolic representation of emojis) [2], which could bring useful information that the model can learn to recognise. For their transformation, we used the emoji library that maps the analysed emoji with its respective text-symbolic representation. For the removal of stop-words, we used the nltk library that filters out articles, lexical particles and words without meaning in Italian by using a predefined list of strings. Another common technique in Natural Language Processing tasks is *stemming*, i.e., reducing inflected

or derived forms of a word to the stem or root of the word to identify the same meaning in different contexts. Although Italian is a very inflected language, when compared for instance to English preliminary experiments in this work did not evidentiate any improvement using stemming, and therefore no preprocessing was done in this sense. In Table 1 sample tweets before and after the cleaning/filtering phase are shown.

Table 1. Examples of text before and after preprocessing

Original:
@user Rom ? Non illudiamoci. Loro non si cambieranno mai, non vogliono lavorare e sono disposti a tutto per rubare soldi. Il loro dio e' il denaro e non ascoltano altrove. Mi dispiace per il clochard.
Preprocessed:
rom illudiamoci cambieranno mai vogliono lavorare disposti rubare soldi dio denaro ascoltano altrove dispiace clochard
Original:
Leggendo questa notizia, mi vengono in mente la Fornero e Monti! Hanno varato una Legge per mandare in pensione gli ITALIANI a due passi dalla miglior vita, per dare i NOSTRI soldi agli IMMIGRATI? Ma chi li appoggia ancora, non si VERGOGNA? URL
Preprocessed:
leggendo notizia vengono mente fornero monti varato legge mandare pensione italiani due passi miglior vita dare soldi immigrati appoggia ancora vergogna
Original:
Zuccaro il magistrato leghista massone escluso dai suoi colleghi dalle indagini mafiose e costretto per farsi notare ad attaccare le ONG e i poveri immigrati
Preprocessed:
zuccaro magistrato leghista massone escluso colleghi indagini mafiose costretto farsi notare attaccare ong poveri immigrati thinking_face thinking_face
Original:
Penso che chi non vuole vedere che gli attentati terroristici sono motivati da volontá di sottomettersi tutti (islam) sia in malafede
Preprocessed:
penso vuole vedere attentati terroristici motivati volontá sottomettersi islam malafede
Original:
@user @user Va deindustrializzata la tratta degli africani e resa compatibile con lo stato di diritto.
Preprocessed:
va deindustrializzata tratta africani resa compatibile stato diritto

2.2 Word Frequency Features and Hate/Stereotype Classes

In this section we discuss some elements characterizing word frequency which can be used to characterize hate speech and/or stereotype. In the following we provide and analyze one Italian sentence for each of the three different classes (i.e., hate speech and stereotypes, hate speech only, stereotypes only); to make the criteria more understandable a translation in English of the sentences and the single words are shown, moreover a general statistic view of the word usage in the data set is provided:

Sentence labelled both hate speech and stereotypes: *#londonattack ask the good ol' boy what the hell they want to do with the Islamic because if I ask, all I get is insults on my mouth.*

Sentence labelled neither stereotypes nor hate speech: *@user @user in facts since they have gained with the #rom fields everything was ok with #Alemanno #Hypocrites*

Sentence labelled no hate speech but stereotypes: *Of course in #Tunisia it must not be easy to investigate extremism in #Tunisia Islamic. They are all extremists.*

Sentence labelled hate speech but no stereotypes: *#SicilyChannel has broken my balls all over the world: politicians, smugglers and immigrants. Close borders and stop.*

A the statistics of the 20 most used words is shown in Fig. 2 the corresponding English words are in order: *not, is, Roms, migrants, are, immigrants, Italy, everybody, Rome, has, Italians, immigration, foreigners, have, more, only/alone, home, camp, terrorists, away.* On the horizontal axis frequency of use is shown, while the term is shown on the vertical axis. Figure 3–Fig. 4 show a class-based analysis:

- Figure 3a shows the 20 Italian words most-used by hate speakers, in order: *not, is, are, everybody, immigrants, migrants, Italy, Roms, Italians, has, have, home, foreigners, terrorists, Islamic, Muslims, more, illegal immigrants, only/alone, Islam*
- Figure 3b shows the 20 words most-used by non-hate speakers: *Roms, non, is, migrants, are, Rome, Italy, immigrants, everybody, has, immigration, camp, nomad, foreigners, have, Italians, more, away, only/alone, Salvini*
- Figure 4a shows 20 words most-used by writers using stereotypes: *non, is, Roms, are, immigrants, everybody, migrants, Italy, has, Italians, have, Rome, foreigners, home, more, illegal immigrants, only/alone, Islamic, Muslim, terrorists*
- Figure 4b shows 20 words most-used by writers not using stereotypes: *non, Roms, is, migrants, are, Italy, immigrants, everybody, Rome, immigration, has, camp, foreigners, Italians, more, away, Salvini (italian politician), have, only/alone, nomads*

It is interesting that while the set of words is almost the same, except for some meaningful words, the frequency of use distribution is the most relevant feature

to distinguish the user groups. The most used word throughout the data set is *Roms* (nomads from Romania), with 1479 occurrences. Users writing texts containing hate speech (i.e., hate speakers) use more frequently words such as *tutti* (*All*) (i.e., conveying absolute and ultimate concepts), *immigranti* (*immigrants*) and *migranti* (*migrants*), compared to non-hate speakers, using more words like *Roma* (*Rome*) and *Italia* (*Italy*), showing more conceptual focus on facts than on characteristics. The most used words by people using stereotypes are also *tutti* (*All*), *Roms* and *immigranti* (*immigrants*), while in the rest of the classes we always have the words Roma (*Rome*) and *migranti* (*migrants*). As we can see, the word *Roma* (*Rome*) is very often used by both of them since is the seat of the Italian government then is a central place often mentioned on decisions about how to manage immigration issues. The word *only* in this paper correspond to the English for the Italian word *solo*. In Italian "solo" can mean "only" or "alone", and the meaning may change in context of different tweets, but it seems apparent that the meaning *only* is the best translation, being a term used to separate classes. Figure 5 and Fig. 6 show the intersection of words present in the text both including and not including hate speech, and the intersection of words present both in the text including and not including stereotypes, which are somehow shared between different groups of users.

Regarding the intersection between the twenty most frequent words used by those who write texts containing hate speech, more than half of the words are used by both classes (i.e., from left to right correspond in English to *Roms, not, is, migrants, are, Italy, immigrates, all, has, foreigners, have, Italians, more, alone/only*) but with different frequency. The same observation applies to the intersection between the most frequent words of those who write texts containing stereotypes and not (i.e., from left to right they correspond in English to, *not, Roms, is, migrants, are, Italians, immigrants, all, Rome, has, foreigners, Italians, more, have, only/alone*).

Analysing the emoji frequency within the texts, we have circa only 282 over 7000 tweets containing emojis. The emoji angry is the most used both by those who write texts containing feelings of hate and stereotypes, while emoji face with tears of joy (i.e., laugh) is used very frequently from all classes (see Fig. 7a–Fig. 8b). Despite the trivially lower frequency of emoticons than words, the significance of emojis for the meaning understatement is apparent. It has to be noted that the problem of hate speech in the social network is particularly diffused because text messages, especially short ones and written without a cognitive filtering step taking time to re-read and correct the message, are particularly guided by emotions, and do not convey the intention of the writer. Therefore, such immediate text replays to posts can be easily misunderstood, while emoticons, despite being still supported for the emotional side and not explaining intentions, can clarify the meaning of the text.

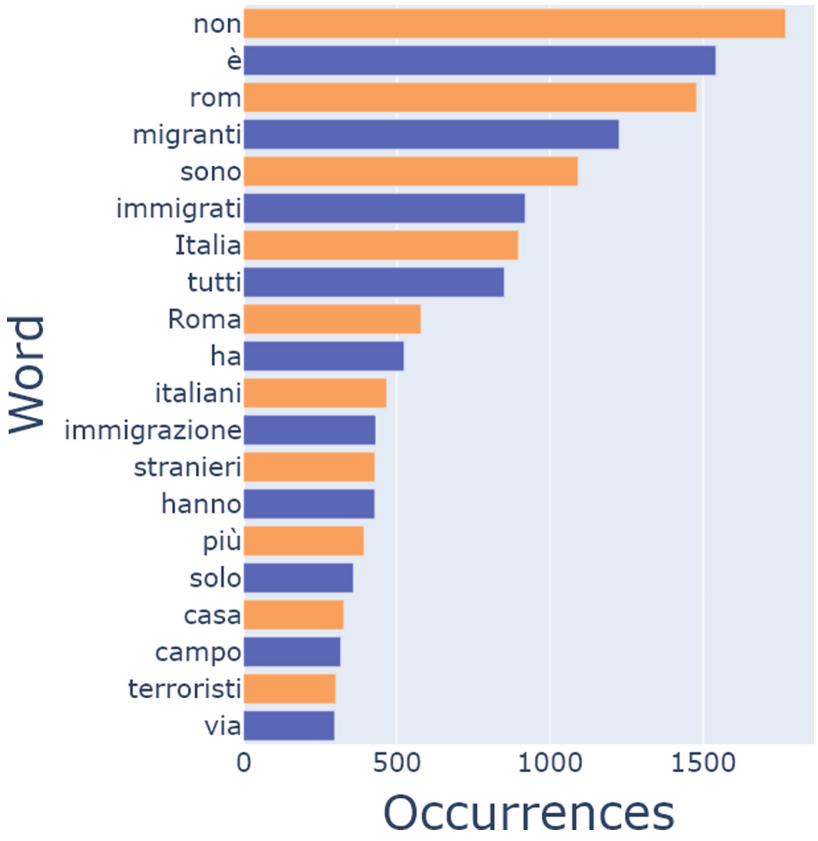
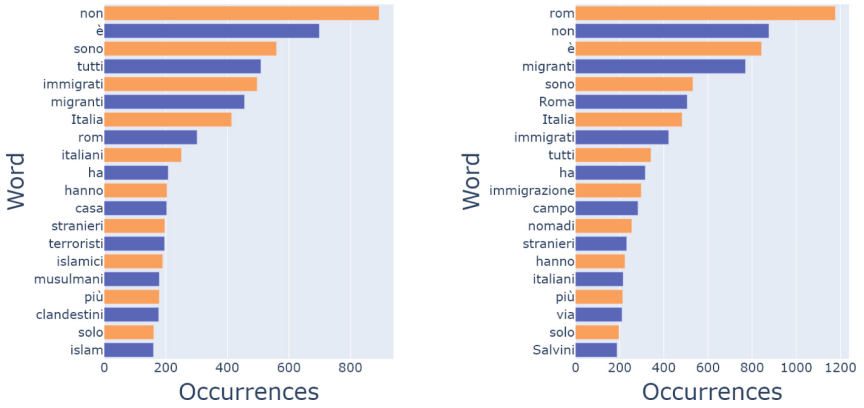


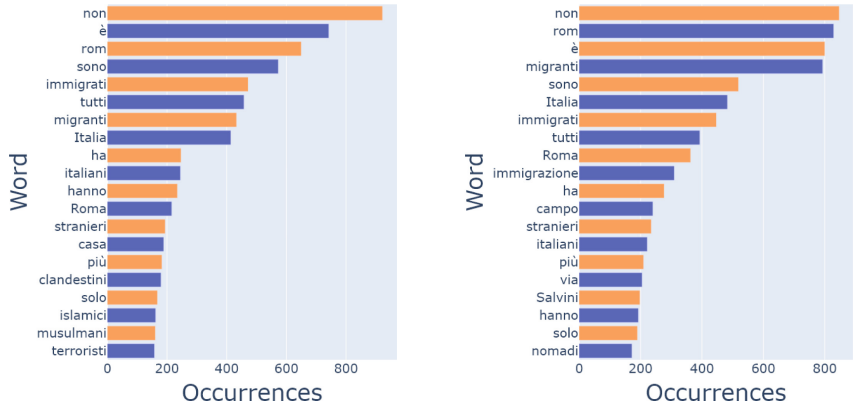
Fig. 2. Twenty most-used words in the whole data set



(a) Twenty most-used words by hate speakers

(b) Twenty most-used words by non-hate speakers

Fig. 3. Twenty most-used words by hate and non-hate speakers



(a) Twenty most-used words by people using stereotypes

(b) Twenty most-used words by people not using stereotypes

Fig. 4. Twenty most-used words by people using and not using stereotypes

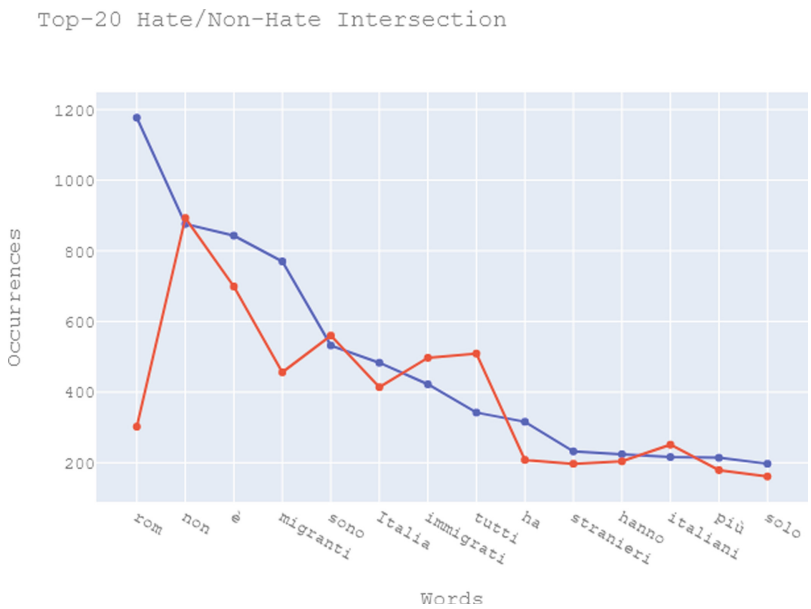


Fig. 5. Intersection of words shared between hate speakers and non-hate speakers

2.3 Feature Engineering

For the feature engineering phase, a preliminary analysis on the length of tweets was carried out, to investigate the possible correlations between text length and Hate/Stereotypes speech. The findings are reported in Table 2; on average content flagged as containing Hate Speech or Stereotypical is longer in terms of

Table 2. Average sentence length, # of characters

Class	Average # characters
Non-hate	136.40
Hate	165.73
Non-Stereotype	137.95
Stereotype	161.19

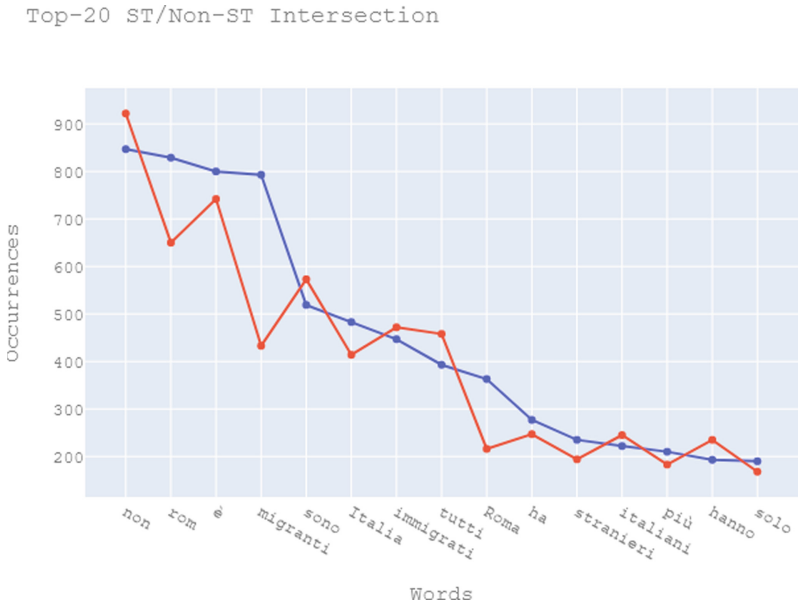
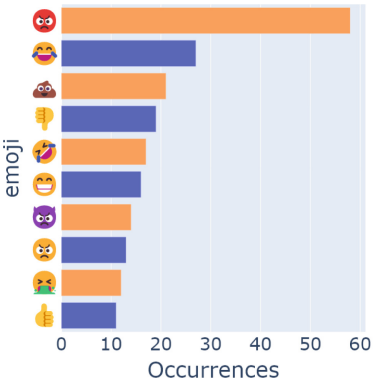


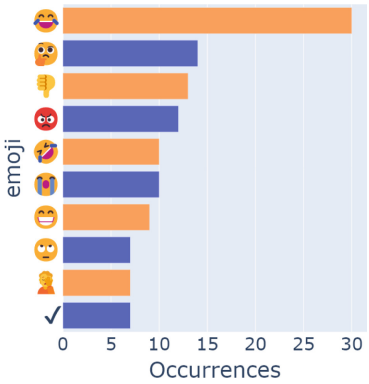
Fig. 6. Intersection of words shared between users using and not using stereotypes

characters. The previous statement is supported also by the figures in Table 3, showing the average length of tweets in terms of words; again, Hate-Speech or Stereotype tweets present a higher number of words. The following features have been computed for each tweet, on the basis of the previous figures and the data in Subsect. 2.2:

- # of words (including stop-words)
- # of characters
- Average Words Length in the sentence (including stop-words)
- Number of stop-words
- Number of words among the top 20 most used words in the whole data set
- Number of words among the top 10 most used words in hate speech tweets
- Number of words among the top 10 most used words in stereotype tweets
- Number of words among the top 10 most used words in non-hate speech tweets

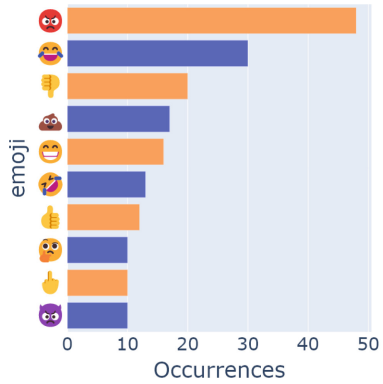


(a) Ten most-used emojis by hate speakers

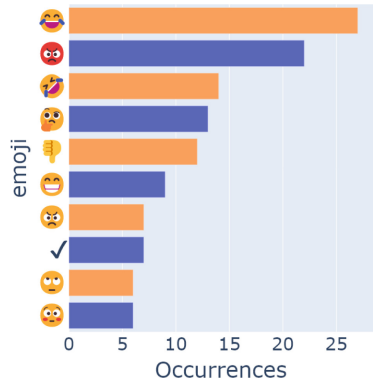


(b) Ten most-used emojis by non-hate speakers

Fig. 7. Top-10 emojis in hate and non-hate speech



(a) Ten most-used emojis by people using stereotypes



(b) Ten most-used emojis by people not using stereotypes

Fig. 8. Top-10 emojis in stereotype and non-stereotype speech

- Number of words among the top 10 most used words in non-stereotype tweets
- # Punctuation signs
- # Exclamation marks
- # Question Marks
- # Fully capitalized words
- # Words starting with a capital letter
- # mentions (@user)
- # of Hashtag signs (#)
- # emojis in the tweet
- # of emojis among the top 5 most used words in hate speech tweets
- # of emojis among the top 5 most used words in non-hate speech tweets

Table 3. Average sentence length, # of words

Class	Average # words
Non-hate	26.34
Hate	31.44
Non-Stereotype	26.59
Stereotype	30.67

– # of emojis among the top 5 most used words in stereotype tweets
– # of emojis among the top 5 most used words in non-stereotype tweets

In the experiments, word embeddings were also calculated and employed as features. Word embedding layers create a higher-dimensional vectorial dense representation of words in a corpus. Words which frequently appear close to each other are mapped to closer vectors; embeddings are frequently employed in NLP machine learning tasks.

3 Artificial Neural Networks for Stereotypes and Hate Speech Detection

The purpose of the network described in the following is to solve two binary classification tasks. The first task classifies tweets as Hateful/Non-Hateful, and the second as containing/not containing stereotypes.

Fully connected neural networks have been employed, with several network preliminary designs tested to choose the best performing ones for the final experiments, for training the network and finally conducting the experimental test. In experiments, the data set training and test sets have been split with an 80%–20% rate.

For the HS detection task, the network has been designed as follows:

– An Embedding layer was prepended to learn the Word Embeddings, with a vocabulary size of 17986, i.e. the number of unique words in the data set after preprocessing, and output size and input length of 42, i.e. the maximum length of a sentence in the data set.
– A Flatten layer, to transform the 2D, [42*42] output of the Embedding layer, to a 1D vector of length 1764.
– A Concatenate layer, to concatenate the word embeddings to the 21 numeric features previously described in Sect. 2.3, with a final vector of 1785 features.
– Three Dense (i.e. Fully Connected) layers, respectively having 64, 32 and 2 neurons; the activation function was Softmax for the last layer, and ReLU for the others.

The structure of the network is shown in Fig. 9. The network was trained using the Adam Optimizer for 4 epochs, with a batch size of 128; the Loss function was Categorical Crossentropy. Additional experiments were performed, feeding

the network either with text features only or word embeddings only. All the settings, including the network structure, were kept fixed, except for the number of training epochs, raised to 20 when only the text features were considered and lowered to 3 for the training on embeddings only. For the Stereotypes task, experiments were performed using the same network structure and combinations of features tested for the HS task. The network was trained for 4, 3 and 20 epochs for, respectively, the Text Features+Embeddings input, the Embeddings only input, and the Text Features only input. Other settings are the same as in the HSD experiment.

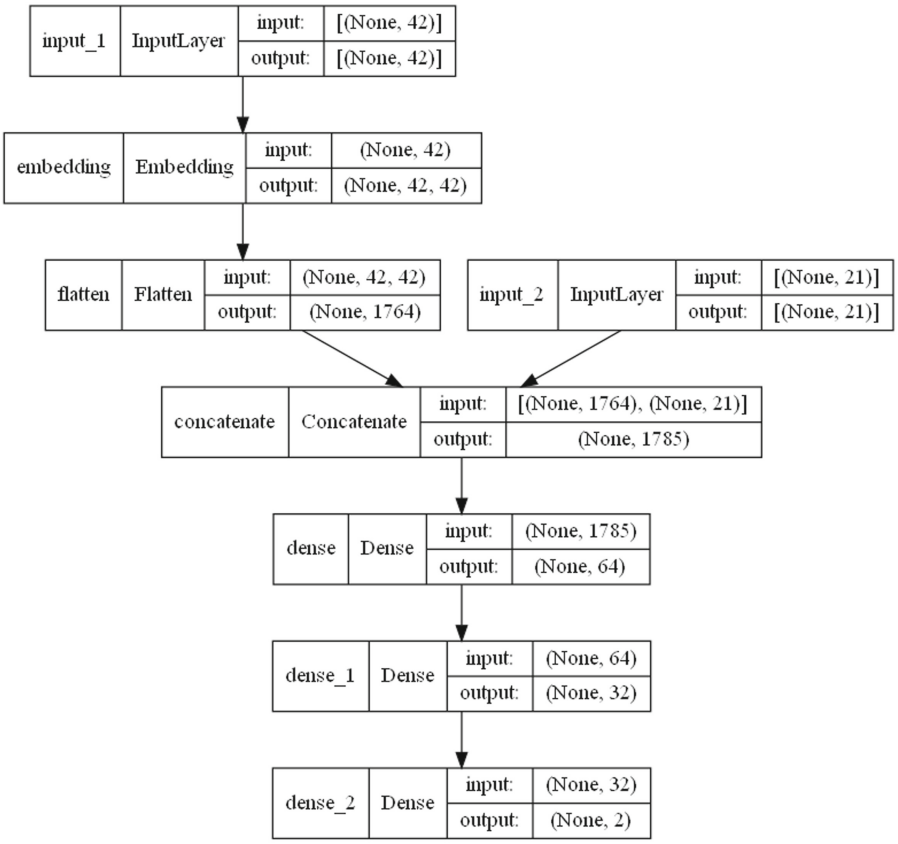


Fig. 9. Neural network structure, HS task

4 Results and Discussion

In this section the results of the experiments for the HSD and ST tasks are reported and discussed.

4.1 Evaluation Metrics

The evaluation metrics include global Accuracy and Macro-Average F1-Score and per-class Precision, Recall, and F1-Score. Let define:

- TP_{Hate} = True Positives i.e., the number of correctly classified items belonging to class Hate
- FN_{Hate} = False Negatives i.e., the number of misclassified items belonging to class Hate
- FP_{Hate} = False Positives i.e., the number of Non-Hate items erroneously classified as Hate
- TN_{Hate} = True Negative i.e., the number of correctly classified Non-Hate items

The same definitions apply to the (resp. Stereotype, Non-Hate or Non-Stereotype) All the per-class metrics are defined in terms of TP, TN, FP, FN. Precision, measuring the rate of relevant instances among the retrieved instanced:

$$precision_i = \frac{TP_i}{TP_i + FP_i}$$

Recall, measuring the rate of relevant instances that have been recalled:

$$recall_i = \frac{TP_i}{TP_i + FN_i}$$

F1, evenly weighting precision and recall, for each class i:

$$F1_i = \frac{2 * precision_i * recall_i}{precision_i + recall_i}$$

Macro-Average F1, calculated as the average of the per-class F1-Scores:

$$Macro - AverageF1 = \sum_{i=1}^{k} \frac{F1_i}{|i|}$$

Accuracy, in binary classification tasks, is defined as the ratio between the number of correctly classified items and the total number of items in the data set:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

The final results are evaluated with Accuracy and Macro-F1.

4.2 Hate Speech Detection Task

In Table 4(a) the per-class results are reported for the model exploiting both text features and word embeddings; the corresponding confusion matrix is reported in Table 5(a). The accuracy and Macro Average F1-Score values are, respectively, 75.47% and 73.5%, making this model the best performing for the HSD task.

Table 4. Per-class metrics, HSD task

(a) Word embeddings + Text features					
Class	Precision	Recall	F-Score	Accuracy	Support
0 (Non-hate)	78%	83%	80%	75.47%	833
1 (Hate)	71%	64%	67%		537
(b) Word embeddings only					
Class	Precision	Recall	F-Score	Accuracy	Support
0 (Non-hate)	78%	77%	78%	72.84%	833
1 (Hate)	65%	66%	66%		537
(c) Text features only					
Class	Precision	Recall	F-Score	Accuracy	Support
0 (Non-hate)	67%	85%	75%	65.98%	833
1 (Hate)	61%	36%	45%		537

Table 5. Confusion matrices, HSD task

(a) Word embeddings + Text features		
Class	Predicted non-hate	Predicted Hate
0 (Non-hate)	689	144
1 (Hate)	192	345
(b) Word embeddings only		
Class	Predicted non-hate	Predicted Hate
0 (Non-hate)	643	190
1 (Hate)	182	355
(c) Text features only		
Class	Predicted non-hate	Predicted Hate
0 (Non-hate)	710	123
1 (Hate)	343	194

In Fig. 10 the Accuracy and Loss values during the training process are shown. A higher number of epochs led to overfitting; therefore, the training process was stopped after four epochs. The additional experiments, exploiting either word embeddings only or text features only, delivered lower results. Complete figures are reported in Table 4(b), Table 5(b), Table 4(c) and Table 5(c).

The accuracy values for the Text Features only and Word Embeddings only experiments are, respectively, 65.98% and 72.84%, and the Macro-Average F1-Score 60% and 72%. It is apparent, from the results, that the most significant contribution to the task is given by the Word Embeddings, while the text features increase the performance only marginally.

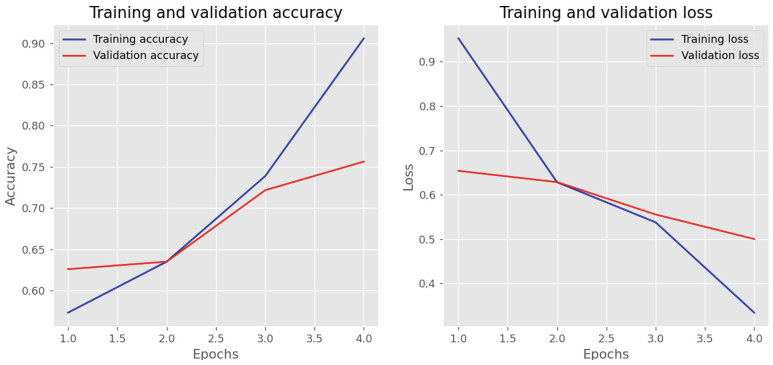


Fig. 10. Accuracy and Loss during training

4.3 Stereotypes Detection Task

Stereotypes detection proved to be a more difficult task for an automated system. For all the experimental settings, the figures are lower than the corresponding experiments for HSD. The best results, reported in detail in Table 6(a) and Table 7(a) were again obtained after training the network with the Word Embeddings and Text Features, which delivered an Accuracy of 69.19% and a Macro-Average F1-Score of 68%.

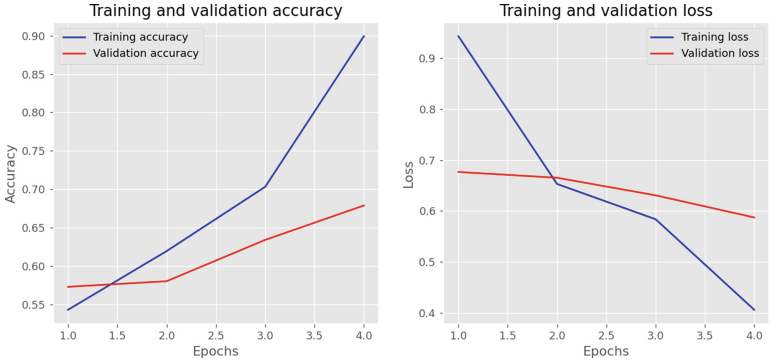


Fig. 11. Accuracy and Loss during training

In Fig. 11 the Accuracy and Loss values during the training process are shown. A higher number of epochs led to overfitting; therefore, the training process was stopped after four epochs. Feeding the network with the Word Embeddings only led to a slight loss in performance, lowering the Accuracy and Macro-Average F1-score to, respectively, 68.75% and 68.5%, showing a prominent role for the Embeddings in the ST task; complete figures are available in

Table 6. Per class metrics, ST task

(a) Word embeddings + Text features					
Class	Precision	Recall	F-Score	Accuracy	Support
0 (Non-stereotype)	70%	78%	74%	69.19%	773
1 (Stereotype)	67%	57%	62%		597
(b) Word embeddings only					
Class	Precision	Recall	F-Score	Accuracy	Support
0 (Non-stereotype)	75%	68%	71%	68.75%	773
1 (Stereotype)	63%	70%	66%		597
(c) Text features only					
Class	Precision	Recall	F-Score	Accuracy	Support
0 (Non-stereotype)	60%	84%	70%	59.92%	773
1 (Stereotype)	58%	29%	39%		597

Table 7. Confusion matrices, ST task

(a) Word embeddings + Text features		
Class	Predicted Non-stereotype	Predicted stereotype
0 (Non-stereotype)	605	168
1 (Stereotype)	254	343
(b) Word embeddings only		
Class	Predicted Non-stereotype	Predicted stereotype
0 (Non-stereotype)	522	251
1 (Stereotype)	177	420
(c) Text features only		
Class	Predicted Non-stereotype	Predicted stereotype
0 (Non-stereotype)	648	125
1 (Stereotype)	424	173

Table 6(b) and Table 7(b). Experiments with the Text Features only did not deliver any appreciable result; the system is only marginally capable of recognizing stereotypes, classifying most of the tweets in the data set as stereotypes-free. The accuracy score, in this case, is 59.92%, and the Macro-Average F1-Score is 54.5%, determined by the low F1-Score for the Stereotype class; results are reported in Table 6(c) and Table 7(c).

5 Conclusions

In this work we present a system for detecting hate speech and stereotypes usage in text communication in Italian language using Artificial Neural Networks and

Natural Language Processing; such systems prove useful in online communication contexts to promote non-aggressive and respectful dialogue between users of social networks, taking into account language-specific aspects. The system has been trained and tested on a data set used for the Evalita 2020 competition; it achieves very promising results on Hate Speech Detection, with an accuracy of 75.47% and F1-Score of 73.5%, and good results on the Stereotype Detection task, with an accuracy of 69.19% and F1-Score of 68.0%. Future experiments aim at improving the performance by experimenting additional textual features and verifying the proposed architecture for hate speech/stereotypes detection on longer texts.

References

1. Pedregosa, F., et al.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**, 2825–2830 (2011). http://dl.acm.org/citation.cfm?id=1953048.2078195
2. Atif, M., Franzoni, V.: Tell me more: automating emojis classification for better accessibility and emotional context recognition. Fut. Internet **14**(5) (2022). https://doi.org/10.3390/fi14050142, https://www.mdpi.com/1999-5903/14/5/142
3. Badjatiya, P., Gupta, S., Gupta, M., Varma, V.: Deep learning for hate speech detection in tweets. In: Proceedings of the 26th International Conference on World Wide Web Companion - WWW 2017 Companion, pp. 759–760. ACM Press, New York (2017). https://doi.org/10.1145/3041021.3054223, http://dl.acm.org/citation.cfm?doid=3041021.3054223
4. Benítez-Andrade, J.e.a.: José alberto. PeerJ Comput. Sci., e906 (2022). https://doi.org/10.7717/peerj-cs.906
5. Bird, S., Klein, E., Loper, E.: Natural Language Processing with Python, 1st edn. O'Reilly Media Inc, Newton (2009)
6. Bosco, C., Dell'Orletta, F., Poletto, F., Sanguinetti, M., Tesconi, M.: Overview of the Evalita 2018 Hate Speech Detection Task. In: Caselli, T., Novielli, N., Patti, V., Rosso, P. (eds.) Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2018). CEUR.org, Turin (2018)
7. Daury Cesar Fabriz, G.H.D.M.: O papel das plataformas de redes sociais diante do dever de combater o discurso de Ódio no brasil. Revista da Facultade de Direito (2022)
8. Davidson, T., Warmsley, D., Macy, M., Weber, I.: Automated hate speech detection and the problem of offensive language (2017). http://arxiv.org/abs/1703.04009
9. Del Vigna, F., Cimino, A., Dell'Orletta, F., Petrocchi, M., Tesconi, M.: Hate me, hate me not: hate speech detection on Facebook. In: CEUR Workshop Proceedings (2017)
10. Gitari, N.D., Zuping, Z., Damien, H., Long, J.: A lexicon-based approach for hate speech detection. Int. J. Multimedia Ubiq. Eng. (2015). https://doi.org/10.14257/ijmue.2015.10.4.21
11. Howard, J.W.: Free speech and hate speech. Ann. Rev. Polit. Sci. **22**(1), 93–109 (2019)
12. Manuela, S., et al.: Haspeede 2@evalita2020: overview of the evalita 2020 hate speech detection task, vol. 2765 (2020)

13. Marreddy, M., Oota, S.R., Vakada, L.S., Chinni, V.C., Mamidi, R.: Am i a resource-poor language? data sets, embeddings, models and analysis for four different nlp tasks in telugu language. ACM Trans. Asian Low-Resour. Lang. Inf. Process. (2022). https://doi.org/10.1145/3531535
14. Santucci, V., Spina, S., Milani, A., Biondi, G., Bari, G.D.: Detecting hate speech for italian language in social media. In: Caselli, T., Novielli, N., Patti, V., Rosso, P. (eds.) Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018) Co-located with the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Turin, Italy, 12–13 December 2018, CEUR Workshop Proceedings, vol. 2263. CEUR-WS.org (2018). http://ceur-ws.org/Vol-2263/paper041.pdf
15. Schmidt, A., Wiegand, M.: A survey on hate speech detection using natural language processing. In: Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media (2017). https://doi.org/10.18653/v1/W17-1101
16. Susumu Annaka, G.K.: Can a constitutional monarch influence democratic preferences? Japanese emperor and the regulation of public expression. Social Sci. Q. (2022). https://doi.org/10.1111/ssqu.13152
17. Waseem, Z., Hovy, D.: Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In: Proceedings of the NAACL Student Research Workshop (2016). https://doi.org/10.18653/v1/N16-2013
18. Zhang, Z., Luo, L.: Hate speech detection: a solved problem? the challenging case of long tail on twitter (2018). http://arxiv.org/abs/1803.03662