applied sciences

MDPI

Article

Time of Your Hate: The Challenge of Time in Hate Speech Detection on Social Media

Komal Florio [1,*], Valerio Basile [1], Marco Polignano [2], Pierpaolo Basile [2] and Viviana Patti [1]

[1] Department of Computer Science, University of Turin, 10149 Turin, Italy; valerio.basile@unito.it (V.B.); viviana.patti@unito.it (V.P.)

[2] Department of Computer Science, University of Bari "Aldo Moro", 70126 Bari, Italy; marco.polignano@uniba.it (M.P.); pierpaolo.basile@uniba.it (P.B.)

* Correspondence: komal.florio@unito.it

Received: 21 April 2020; Accepted: 9 June 2020; Published: 18 June 2020

Abstract: The availability of large annotated corpora from social media and the development of powerful classification approaches have contributed in an unprecedented way to tackle the challenge of monitoring users' opinions and sentiments in online social platforms across time. Such linguistic data are strongly affected by events and topic discourse, and this aspect is crucial when detecting phenomena such as hate speech, especially from a diachronic perspective. We address this challenge by focusing on a real case study: the "Contro l'odio" platform for monitoring hate speech against immigrants in the Italian Twittersphere. We explored the temporal robustness of a BERT model for Italian (AlBERTo), the current benchmark on non-diachronic detection settings. We tested different training strategies to evaluate how the classification performance is affected by adding more data temporally distant from the test set and hence potentially different in terms of topic and language use. Our analysis points out the limits that a supervised classification model encounters on data that are heavily influenced by events. Our results show how AlBERTo is highly sensitive to the temporal distance of the fine-tuning set. However, with an adequate time window, the performance increases, while requiring less annotated data than a traditional classifier.

Keywords: hate speech monitoring; diachronic analysis; microblogging data; supervised machine learning

1. Introduction

The increasing availability of textual data from social media platforms is essential to the development of training datasets for Natural Language Processing (NLP) prediction tasks. In particular, hate speech detection is the NLP task that aims at classifying segments of text based on their hateful content. The abundance of data allows the research community to tackle more in-depth long-standing questions such as understanding, measuring, and monitoring users' sentiment towards specific topics or events. However, it has also contextually brought new challenges such as the need to evaluate the performance of prediction models over different time spans and the necessity of accounting for relatively quick topic shifts in online discourse. This is particularly relevant in the context of hate speech on social media, where users very often react to breaking news from media and relevant real-life events. This dynamic is mirrored by the fact that the language of social media is produced spontaneously, therefore it is often characterized by significant variations over time in terms of topics and linguistic patterns. These considerations suggest that there is a deep need to precisely measure the robustness of hate speech detection systems over time, as usually they are trained on data that are widely separated on the temporal scale from the data used for testing the performance.

The motivation and the urgency for a diachronic study arose at first from observing the needs and the difficulties that were encountered in the development of hate speech monitoring platforms such as “Contro l’odio” [1], a web service launched in 2018 to predict and monitor hate speech messages against immigrants posted on the Italian Twitter. Monitoring and countering hate speech is a shared goal of several recent projects, which focused on different targets of hate, monitoring different countries and territories, and differ in the granularity of the detection, of the temporal spans considered, and regarding the visualization techniques provided to inspect the monitoring results. Let us mention the CREEP project on monitoring cyberbullying online [2], with an impact also on the Italian territory, HateMeter (http://hatemeter.eu/), with a special focus on Anti-Muslim hatred online in different European countries (Italy, France, and England), and the MANDOLA project [3] providing an infrastructure enabling the reporting of illegal hate-related speech.

The platform “Contro l’odio”, providing the monitoring setting which motivated this research work, combines computational linguistics analysis with map-based visualization techniques. It offers a *daily monitoring* of hate speech against immigrants in Italy and its evolution over time and space to provide users with an interactive interface for exploring the dynamics of the discourse of hate against immigrants in Italian social media (The platform is online and can be accessed at https://mappa.controlodio.it/). Three typical targets of discrimination related to this topical focus are taken into account: Migrants, Muslims, and Roma, since they exemplify discrimination based on nationality, religious beliefs, and ethnicity.

Since November 2018, the platform analyses daily Twitter posts and exploits temporal and geo-spatial information related to messages to ease the summarization of the hate detection outcome.

The automatic labeling of the tweets of the “Contro l’odio” platform is performed by a Support Vector Machine (SVM) classifier. It was trained on data from 2017 and then tested on messages streamed from October 2018 up to today. It is clear that for a service like this to be dependable and consistent over time, there is a need to explore in-depth the interplay of language and topic shifts in time and the robustness of the prediction system. We hence propose a novel approach to tackle the issue of diachronicity in hate speech prediction, by means of a transformer-based neural network classifier, AlBERTo, which is trained on Italian social media language data. AlBERTo provides a pre-trained language model of Italian, and it is fine-tuned on monthly samples from the “Contro l’odio” dataset to be able to classify instances of hate speech. In this paper, we introduce an evaluation of strategies to alleviate the diachronicity issue. In general, this work tackles the following questions:

RQ1 How can we evaluate the temporal robustness of different hate speech prediction systems, with respect to language and topic change over time?

RQ2 What is the impact of the size and temporal coverage of the training set on the temporal robustness of the prediction?

This paper is organized as follows: Section 2 contains an overview of related works, Section 3 contains a high-level description of our experiments setting, while the data we used for our work are described in Section 4. The details of our experiments and results are presented in Section 5, while a qualitative lexical analysis is included in Section 6.

2. Related Works

Hate speech detection is a relatively new topic of investigation in which artificial intelligence technology is applied to monitor extreme, potentially dangerous manifestation of hostility and toxic discourse online. The motivation to study hate speech from a computational perspective is manifold. On the one hand, computational linguistic techniques enable the scholar to gain insights and empirical evidence on intrinsic characteristics at the semantic and pragmatic level of a spreading phenomenon. On the other hand, there are several subjects, like institutions and ICT companies that need to comply with governments demands for counteracting hate speech online (see, for instance, the recently issued EU commission Code of Conduct on countering illegal hate speech online [4]). This generates an increasing necessity for automatic support to content moderation [5] or to monitor and map the diffusion of hate speech and its dynamics over a geographic territory [1], which is only possible on a large scale by employing computational methods. Moreover, having reliable methods to compute an index of online hate speech in relation to specific geo-times coordinates automatically opens the way to the possibility to investigate the interplay between the volume of hate speech messages and the socio-economic and demographic traditional indexes for a given area and period (see [6] for a preliminary proposal on the Italian case), or to study the impact of offline violent effects on hateful online messages [7].

The field has been recently surveyed in [8,9]. The vast majority of the papers analyzed in [8] describes approaches to hate speech detection based on supervised learning, where the task is treated as a sentence—or message—level binary text classification task. The different models and features presented in the literature are difficult to compare effectively because the results are evaluated on individual datasets that are often not public, hence the survey advocates for broader availability of publicly available data. This evaluation gap is being bridged recently by evaluation campaigns for English, Spanish (SemEval [10]), German [11], and Italian (EVALITA [12]), whose shared tasks released annotated datasets for hate speech detection. The availability of benchmarks for system evaluation and datasets for hate speech detection in different languages made the challenge of investigating architectures, which are also stable and well-performing across different languages, an exciting issue to research [13,14].

In this work, we focus on another novel research challenge related to the temporal robustness of hate speech detection models. We compared diachronic performance of two different prediction systems, namely a SVM model and BERT, a Bidirectional Encoder Representations from Transformers [15] open-sourced by Google, widely regarded as one of the most interesting breakthroughs in machine learning applied to NLP tasks. The main aim of BERT is to tackle the limited availability of annotated training data for NLP tasks by means of pre-train a general-purpose language representation models directly on the unannotated text, because datasets of this kind tend to be much larger and more easily available. BERT relies on the latest development in pre-training contextual representations, such as Semi-supervised Sequence Learning [16], ELMo [17], ULMFit [18], OpenAI Transformer [19] and Transformers [20] while implementing a deeply bidirectional architecture. This system represents a meaningful improvement from previous techniques because it combines two crucial features: context awareness and bidirectionality. Context awareness means that the model creates a representation for each word in the dictionary based on the other words in the sentence, while bidirectionality indicates that the model predicts a word based on both what precedes and follows the term subject of prediction. Being computationally very expensive, researchers only recently succeeded in training BERT deep neural networks. The aforementioned survey [9] includes the more recent BERT model and introduces a modified and more transparent version of an SVM classifier that does not, however, outperform BERT.

We focus on the Italian language on Twitter, building on AlBERTo [21], a BERT model pre-trained on the large-scale corpus of Italian Twitter data TWITA [22]. The authors of [21] show that BERT is a truly powerful tool when applied to training and test sets drawn from the same distribution, in particular for sentiment analysis and irony detection. This result is confirmed in [23], where AlBERTo is applied to hate speech detection on Italian social media. We trained AlBERTo on data that also encompassed the train and reference set from Haspeede [12], the first shared task on hate speech on Italian organized within EVALITA2018 evaluation campaign (http://www.di.unito.it/~tutreeb/haspeede-evalita18/index.html).

As stated in the introduction, this work aims to explore and compare the temporal robustness of hate speech detection models. This research goal emerged when reflecting on the setup of the project “Contro l’odio” (described in [1]), whose primary output consists of a web service dedicated to monitor, measure and visualize the rate of hate speech against immigrants in Italy, compared to the overall volume of messages of the Italian Twittersphere. In this context, it became evident that since the monitoring and classification of tweets are ongoing and the dataset on which the system is trained is fixed, the temporal robustness of the classification models is a non-trivial issue.

Computational approaches to the diachronic analysis of language [24] have been gaining momentum over the last decade. An interesting analysis of the dynamics of language changes has been provided by [25]. The authors describe what happens from the language analysis point of view on words that change their meaning during the time. Most of them show a *social contagion* where the meaning is changed by their common/wrong use on social media platforms. Clyne et al. in [26] discuss the changing of words meaning by the influence of immigrant languages, Lieberman et al. [27], instead, try to quantify these changes in the common language. These studies support our idea about the possible difficulties of an automatic machine learning approach to classify new sentences that have been collected in a time distant enough from the one of training data. We suppose that the language of hate speech is very volatile and influenced by events, and it changes words meaning faster than usual: all these considerations have encouraged us to investigate the robustness of some machine learning models over time.

The recent availability of long-term and large-scale digital corpora and the effectiveness of methods for representing words over time played a crucial role in the recent advances in this field. However, only a few attempts focused on social media [28,29], and their goal is to analyze linguistic aspects rather than understanding how lexical semantic change can affect performance in sentiment analysis or hate speech detection. From this perspective, our work represents a novelty: for the first time, we propose to tackle the issue of diachronic degradation of hate speech detection by exploring the temporal robustness of prediction models. The closest works found in recent literature are [30], where the authors explore the diachronic aspect in the context of user profiling, and [31], who provides a broader view on diachronicity in word embeddings and corpora. Nevertheless, this is the first work investigating the diachronic aspect in the specific context of hate speech detection, which is a crucial issue, especially in application settings devoted to monitoring the spread of the hate speech phenomenon over time.

3. Method and Models

We designed a series of experiments to evaluate several strategies for hate speech detection in a diachronic setting. Individually, all the experiments follow the same structure, where a classifier is trained or fine-tuned on a training set and tested on a smaller test set. To test the robustness of prediction models against changes in language and topic over time, we trained our models in two different scenarios and then compared the performance. In the first case, we used training data from *one single month*, while in the second case, we *progressively increased the size of the training* set by injecting information about the history of the corpus and hence the evolution of language and topics over time. We compared the performance of different models in terms of precision, recall, and F1-score. We focused on these metrics relative to the *positive class* (the presence of hate speech), because the task at hand is a *detection* task, as opposed to a *classification* task. To smooth out any possible statistical anomaly, we ran every prediction for five times, each with a different seed for the random number generator, and then we averaged the metrics over all the runs. We employed a series of test sets drawn from the "Contro l'odio" dataset [1]: each one is a sample covering one month of Twitter messages.

We focused on the use of two very different classification models: SVM [32] and AlBERTo, the Italian BERT language model [21].

The core contribution of our work relies on the exploration and evaluation of how the distance in time between a training and a test data impacts the performance of two models who display profound differences in how they were built and how they work when performing classification tasks.

SVMs belongs to the family of supervised machine learning algorithms and is commonly used to classify data into two independent classes, which very often consists of text classification [33,34]. In particular, the text, adequately encoded into its vectoral representation (e.g., TF-IDF [35], word-embedding [36]) is provided as training to the model in order to generalize the weight of the equation of a hyperplane which is able to divide the examples into the given classes. During the evaluation phase, when the text labels are unknown, the model applies the learned discrimination model for labeling the test examples. The SVM algorithm family is divided into two main classes: linear models, which represent the division of data into classes by means of a simple straight "line", and polynomial algorithms, which implement more sophisticated equation to perform the same task in more complex scenarios. The strategy used for the construction of hyperplane is commonly known as the kernel function. A commonly used kernel is RBF (Radial Basis Function) [37] which in general shows good performance for many NLP tasks [38,39].

BERT is a novel task-independent language model [15] based on the idea of creating a deep learning architecture. More specifically it encompasses encoder and a decoder, so that the encoding level can be used in more than one NLP task while the decoding level contains weights which are then optimized for a specific task (fine-tuning). For this reason, a general-purpose encoder should be able to provide an efficient representation of the terms, their position in the sentence, context, the grammatical structure of the sentence, and semantics of the terms. The idea behind such models is that if a model can predict the next word that follows in a sentence, then it can generalize the syntactic and semantic rules of the language. BERT [15] was developed to work with a strategy very similar to GPT [40], hence the basic version is trained on a Transformer network with 12 encoding levels, 768 dimensional states, and 12 heads of attention for a total of 110M of parameters trained on BooksCorpus [41] and Wikipedia English for 1M steps. The main difference with GPT lies in the learning phase, which is performed by scanning the span of text in both directions, from left to right and from right to left. This strategy is however not entirely a novelty as it was previously implemented in BiLSTMs [42]. Moreover, BERT uses a "masked language model": during the training, random terms are masked to be predicted by the net. Jointly, the network is also designed to potentially learn the next span of text from the one given in input. These variations on the GPT model enable BERT to be the current state-of-the-art language-understanding model. Larger versions of BERT (BERT large) have been released and are scoring better results than the normal scale models, but they require far more computational power. Considering the international focus on language models generated through deep neural networks and their lack for the Italian language, AlBERTo has been proposed as a valid resource to fill this gap, as it was developed starting from the BERT base model. AlBERTo has been trained on *TWITA* [22] a collection of domain-generic tweets in Italian extracted through API streams and free to use for research purposes. More details about AlBERTo are available in [21,23,43].

The SVM model has been implemented using the LibSVM java library (https://www.csie.ntu.edu.tw/~cjlin/libsvm/) [44]. We used the simplest version available: a linear version of the kernel and a value of the parameter C equal to its standard value of 1. We did not perform any approach of tuning of parameters because it is out of the purposes of the work. As already mentioned, the main goal of the work is to observe the influence of the temporal distance among training and test data in the performance of supervised machine learning models. Consequently, we were not interested in obtaining state-of-the-art results in the accuracy of classification.

The model based on AlBERTo has been implemented using mainly Tensorflow [45] and Keras (https://keras.io/), the famous deep learning library. The performance was evaluated with the metrics provided by scikit-learn (https://scikit-learn.org/). The fine-tuning of the AlBERTo model for the specific classification task was performed predominantly on Google Colab using a TPU. The evaluation phase has required only a GPU on the same platform. Google Colab has been chosen as the running environment because, at the moment, it represented the most efficient and powerful cloud computing platform available for training deep learning models for free. During the fine-tuning phase of AlBERTo, we estimated the number of learning epochs as a result of an empirical evaluation carried out on a validation subset made of 200 sentences extracted from the same data distribution used in training and testing. Starting with 2 epochs, we increased the number by two at a time up to 10. From the results of this setting, we observed that the best performance equals to 0.518, considering the F1-score on the positive class, was obtained by setting the number of epochs to 8. This value was used as a fixed parameter for all the fine-tuning processes. The learning rate has been kept at its default value of 2 × 10^{-5}, while the training and prediction batch size was set to 512 to improve the predictive accuracy of the model as much as possible. Since we mainly worked with short texts, we decided to leave the pre-defined maximum input size of 128 tokens. The fine-tuned version of AlBERTo has been used as part of a standard Keras classification model. In particular, as shown in Figure 1, we collected the embedding representing the input from the NSP-Dense layer of AlBERTo, i.e., the first dense layer after the CLS token embedding. Then we stacked on it a final dense layer with a SoftMax activation function in order to predict the probability that the sentence may be a hate speech (class equal to 1) or not (class equal to 0).

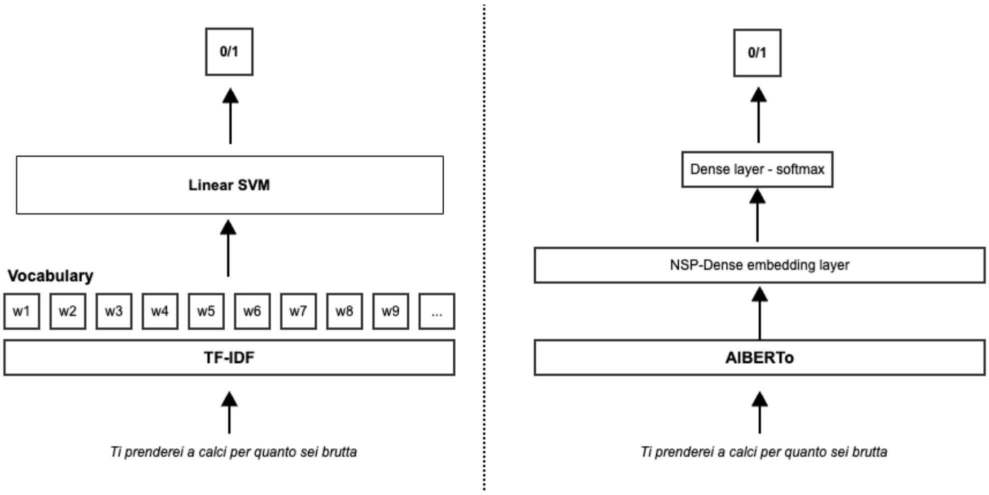

0/1
Linear SVM
Vocabulary
w1
w2
w3
w4
w5
w6
w7
w8
w9
...
TF-IDF
Ti prenderei a calci per quanto sei brutta
0/1
Dense layer - softmax
NSP-Dense embedding layer
AlBERTo
Ti prenderei a calci per quanto sei brutta

Figure 1. On the left it is showed the SVM model, on the right the one based on AlBERTo.

Following our assumption that the most frequently discussed topics in our data are strictly related to breaking news diffused by traditional and online media, and hence tend to shift in time relatively quickly, we performed a lexical analysis in order to capture and visualize this intuition. The results of the experiments are presented and discussed in Section 5, while the lexical analysis is described in depth in Section 6.

4. Data

We present in the following the training and test data used in our experiments. In particular, both the test set and part of the training are drawn from the same source: TWITA [22], a large-scale collection of Italian tweets.

Our training data originates from two different sets of annotated data. The first set consists of both the training and reference dataset of the Haspeede (Hate Speech Detection) shared task, organized within Evalita 2018 [12]: a total of 4000 tweets (3000 tweets in the training test and 1000 in the reference set) collected from 1 October 2016 to 25 April 2017. These messages were annotated with a mixed procedure: a subset was manually annotated by five independent experts while the rest of the data was crowdsourced on Figure Eight (The annotation guidelines are available here: https://github.com/msang/hate-speech-corpus/blob/master/GUIDELINES.pdf). The full annotation process (including information on the inter-annotator agreement) is presented in [46,47]. The second part of our training set is formed by data filtered from TWITA [22], a large-scale collection of tweets started in 2012 and currently ongoing. It relies on the Twitter Streaming API to download a sample of messages in Italian posted each day. Although the collection counts over half a billion tweets between 2012–2017, the subset targeted for our purposes is much smaller, resulting from a topic-based selection. We draw a selection of 3809 tweets from 2015 and 3200 from 2017 using a list of topic-based keywords and imposing the constrain of the tweet having a geotag in Italy, as we are interested in linguistics phenomena in Italian. The handcrafted list of keywords extends the work described in [46,47], where a set of terms was compiled by assessing which minority groups are most likely to be targeted for HS in the online discourse about immigration. This choice was based on the results of the 2015 Eurobarometer Survey on discrimination in the EU (http://ec.europa.eu/justice/fundamental-rights/files/factsheet_eurobarometer_fundamental_rights_2015.pdf). We further expanded this collection of terms using the Open Multilingual Wordnet (http://compling.hss.ntu.edu.sg/omw/) (OMW), not only to collect a larger number of messages, but also to capture a wider variety of expressions on this topic. For each of these keywords, we collected from OMW all the related hypernyms and co-hypernyms. We then manually cleaned this new set of keywords to retain only the most relevant ones. The final list of our keywords is in Table 1.

Table 1. List of keywords in Italian (and their English translation) used to compile the dataset.

Keywords in Italian	English Translation
clandestin*	*illegal immigrant(s)*
corano	*Quran*
emigrant*	*emigrant(s)*
emigrat*	*emigrant(s)*
esul*	*exile(s)*
fondamentalist*	*fundamentalist(s)*
imam	*imam*
iman	*imam*
immigrant*	*immigrant(s)*
immigrat*	*immigrant(s)*
Islam	*Islam*
islamismo	*islamism*
islamit*	*Islamist(s)*
maomettan*	*Mohammedan(s)*
migrant*	*migrant(s)*
migrazione	*migration*
mussulman*	*muslim(s)*
mussulmanesimo	*Islam*
musulman*	*muslim(s)*
nomad*	*nomad(s)*
profug*	*refugee(s)*
sfollat*	*displaced*
stranier*	*foreigner(s)*

(Asterisks * stand for the different combination of word endings in Italian, e.g., clandestin* represents clandestina, clandestino, clandestine and clandestini).

When filtering out text data, it is crucial to take into account the possibility of substring matching, e.g., “Rom” would match “Roma” (the capital city of Italy). We addressed this issue by implementing regular expressions in our filtering algorithm to match only entire keywords preceded and followed by white-spaces, punctuation, or beginning/end of a sentence. Our approach was purely string-based; therefore, we could collect tweets containing keywords occurring with a different meaning from expected, such as named entities (e.g., “Nomadi” is the Italian form for “nomads”, but also the name of a popular music band). However, upon manual inspection, we noticed that these occurrences are extremely rare in our collection. The tweets were then annotated by three independent contributors on Figure Eight, now Appen (https://appen.com/), chosen to be Italian speakers, geolocated in Italy and using the same guidelines as in [12]. More in detail, each tweet had a so-called confidence score that captures the level of agreement between multiple contributors and indicates the “confidence” in the validity of the labeling. This index, based on Krippendorff’s α metric, also incorporates a weighted average over the annotators’ trust scores that tracks the dependability and consistency of each annotator’s labeling history over time on the platform.

In the following sections, we will refer collectively to all the data used for training the models as Haspeede+. Our aim was to investigate the temporal robustness of BERT in predicting hate speech on Twitter messages related to immigration phenomena in Italy. In this context, we needed a set of data on this topic with a temporal structure that allowed us to capture variations in language and topics over time and then measure the hate speech detection systems performance using standard metrics, such as precision, recall, and F1-score. For this purpose, the data filtered as part of the "Contro l'odio" project, described in Section 1, were the perfect solution, both in terms of topic and temporal distribution. We used as test data random monthly samples of roughly 2000 tweets per set, from September 2018 to February 2019. This dataset was also entirely annotated on Appen with the same strategy illustrated before. In Table 2, we list the detailed size and class balance of all our datasets. We notice that the percentage of HS tweets decreases with time, while tweets are being annotated by the same set of annotators. A possible explanation of this is that the data from 2019 may be significantly different from the examples given to the annotators as guidelines (which belonged to previous years), and this yields to an inconsistently in the quality of the annotation results and ultimately to this class imbalance. The average length of the tweets is 24 words in the training sets and 38 words in the test sets. However, the training sets were collected using the standard Twitter API truncating messages longer than 140 characters, while the test sets were collected with the updated API returning the full messages. In terms of language variability, the type-token ratios are expectedly low, ranging from 10% to 18% across data sets.

Table 2. Dataset size and class balance.

Dataset	Size	% non-HS	% HS
Haspeede Set	4000	67.6	32.4
Figure Eight Train Set 1 (data from 2015)	3809	85.5	14.5
Figure Eight Train Set 2 (data from 2017)	3200	82.7	17.3
test 2018_09	1991	67.5	32.5
test 2018_10	2000	82.9	17.1
test 2018_11	2000	84.2	15.8
test 2018_12	2000	84.1	15.9
test 2019_01	2000	90.2	9.8
test 2019_02	2000	91.4	8.6

5. Experimental Evaluation

We devised a set of experiments that allowed us to track precisely how different models performed when trained and fine-tuned on different combinations of datasets, covering different temporal ranges.

5.1. Experimental Design

For what regards the prediction systems, we decided to compare AlBERTo against a traditional SVM, as it is the one in use in the "Contro l'odio" project. We then trained each system in two different scenarios: a *sliding window* model and an *incremental* model. In the first case we trained the system on month t_i and then tested it on the following month t_{i+1}.

In the second case instead we progressively incremented the size of the training set: we tested the models on month t_i but trained them on data from all the previous months, from t_0 to t_{i-1}. The rationale behind this choice was to evaluate how the system performance vary while injecting information on language and discourse about the recent past. To explore the interplay between the size of the training set and the temporal gap with the test data, we performed a second set of experiments with a fixed test set but adding Haspeede+ to each of the two training schemes. The reason for this was to evaluate how the systems performed when trained on a larger but older dataset, injected with information on language and topics far away in the past. For comparison, we also tested both systems after having trained them only on Haspeede+, as a baseline for comparison with the other settings.

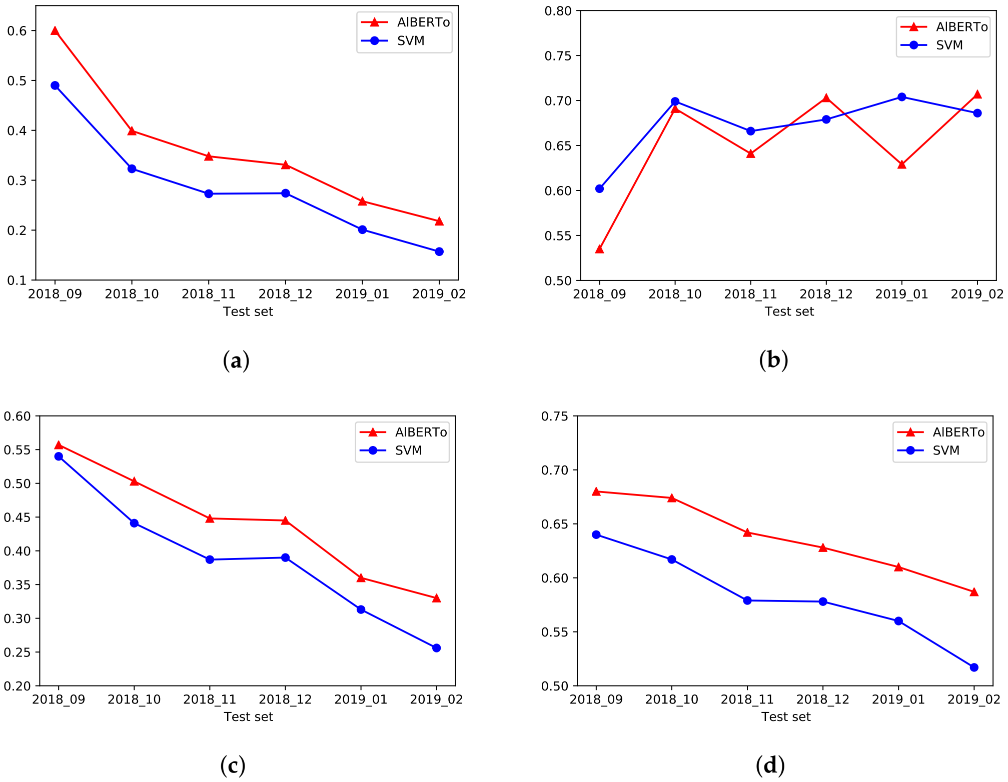
To smooth out any possible random effect, we repeated every single experiment for all the possible setups five times, each of which had a different random seed. We then computed the arithmetical average of the standard metrics.

Moving from the consideration that we are performing a hate speech detection task and not a more general classification task, we decided to focus on precision, recall, and F1 for the positive class (the presence of hate speech) and macro F1-score on both classes. The reason behind this choice is that we believe that the key point in our experiment was to measure the effectiveness of the algorithm when correctly detecting hate speech messages, rather than correctly labeling non-hate speech messages. As an example, we believed that for us, the model needed to be able to correctly classify the hate speech sentence "You are ugly, kill yourself" more than classifying the sentence "Today is a good day" as not hate speech. We also computed the macro F1-score that averages on both classes because, as seen in Table 2, the distribution of the label in our training datasets is unbalanced, and this last metrics provides a better insight on the system performance in this specific case.

5.2. Results

We trained the model on Haspeede+, a fixed set of data from 2015 and 2017, which are a few years older than our test set. This experiment represents a sort of baseline to evaluate the performance in the other setups. All the metrics from this setting are presented in Table 3.

We notice in Figure 2 that, as we expected, both the precision and the F1-score display a generically decreasing trend over time in both cases, and AlBERTo does not outperform the SVM significantly in this case. The two models display in general a similar trend over time for what regards the recall. These considerations are supported by the last chart presented in Figure 2, as the macro F1-score is built as an average of the F1 over the two classes. Specifically, in six months, the model based on AlBERTo has lost 0.227 points of F1 while SVM 0.284. However, this value is influenced by the recall that instead tends to increase with the passage of time as a consequence of its poorer ability of the model to return results accurately.



(**a**)

(**b**)

(**c**)

(**d**)

Figure 2. Evaluation of the model trained on Haspeede+ (fixed training set). (**a**) Precision on the positive class. (**b**) Recall on the positive class. (**c**) F1-score on the positive class. (**d**) Macro-averaged F1-score.

This trend becomes clear if we observe the chart of the precision of the positive class: this metric is crucial as our goal is the minimization of false positives. In such graph, the two models have an equivalent downward trend for each month, showing that the diversification of the language used in sentences strongly influences the classification performance.

In Figure 3 we present the compared results of the experiments with the two different training set scenarios: the sliding window (on the left side) and the incremental (on the right side). In each graph, we plotted the results with and without the injection of the Haspeede+ set. For the sake of clarity and completeness we present all the metrics for the experiments without the Haspeede+ dataset in Tables 4 and 5. The results of the experiments with the Haspeede+ dataset are instead listed in Tables 6 and 7.

Table 3. Numerical results of the evaluation of the model trained on Haspeede+ (fixed training set).

Test Set	SVM				AlBERTo			
	Prec.	**Rec.**	**F1**	**F1 macro**	**Prec.**	**Rec.**	**F1**	**F1 macro**
2018_9	0.490	0.602	0.540	0.640	0.600	0.535	0.557	0.680
2018_10	0.323	0.699	0.441	0.617	0.399	0.691	0.503	0.674
2018_11	0.273	0.666	0.387	0.579	0.348	0.641	0.448	0.642
2018_12	0.274	0.679	0.390	0.578	0.331	0.703	0.445	0.628
2019_01	0.201	0.704	0.313	0.560	0.258	0.629	0.360	0.610
2019_02	0.157	0.686	0.256	0.517	0.218	0.707	0.330	0.587

Table 4. Numerical results of the evaluation of the model trained on Sliding Window (no Haspeede+) dataset.

Test Set	SVM				AlBERTo			
	Prec.	**Rec.**	**F1**	**F1 macro**	**Prec.**	**Rec.**	**F1**	**F1 macro**
2018_10	0.350	0.500	0.412	0.629	0.406	0.641	0.497	0.679
2018_11	0.475	0.319	0.375	0.639	0.454	0.513	0.448	0.686
2018_12	0.427	0.230	0.299	0.600	0.491	0.447	0.445	0.694
2019_01	0.331	0.214	0.260	0.598	0.425	0.367	0.360	0.661
2019_02	0.382	0.169	0.234	0.592	0.421	0.342	0.330	0.673

Table 5. Numerical results of the evaluation of the model trained on Incremental (no Haspeede+) dataset.

Test Set	SVM				AlBERTo			
	Prec.	**Rec.**	**F1**	**F1 macro**	**Prec.**	**Rec.**	**F1**	**F1 macro**
2018_10	0.350	0.500	0.412	0.629	0.406	0.641	0.497	0.679
2018_11	0.343	0.435	0.384	0.624	0.415	0.694	0.519	0.694
2018_12	0.389	0.387	0.388	0.636	0.464	0.627	0.533	0.704
2019_01	0.273	0.362	0.311	0.611	0.436	0.434	0.435	0.687
2019_02	0.266	0.448	0.334	0.624	0.356	0.539	0.429	0.679

Table 6. Numerical results of the evaluation of the model trained on Sliding Window and Haspeede+ dataset.

Test Set	SVM				AlBERTo			
	Prec.	**Rec.**	**F1**	**F1 macro**	**Prec.**	**Rec.**	**F1**	**F1 macro**
2018_10	0.343	0.649	0.449	0.634	0.387	0.764	0.514	0.677
2018_11	0.329	0.571	0.418	0.628	0.445	0.479	0.461	0.684
2018_12	0.347	0.569	0.431	0.640	0.415	0.507	0.456	0.665
2019_01	0.239	0.551	0.334	0.603	0.315	0.525	0.394	0.649
2019_02	0.195	0.494	0.280	0.575	0.283	0.514	0.365	0.636

Table 7. Numerical results of the evaluation of the model trained on Incremental and Haspeede+ dataset.

Test Set	**SVM**				**AlBERTo**			
	Prec.	**Rec.**	**F1**	**F1 macro**	**Prec.**	**Rec.**	**F1**	**F1 macro**
2018_10	0.343	0.649	0.449	0.634	0.439	0.672	0.524	0.694
2018_11	0.335	0.587	0.427	0.633	0.415	0.616	0.493	0.684
2018_12	0.360	0.535	0.430	0.645	0.471	0.516	0.470	0.680
2019_01	0.243	0.464	0.319	0.603	0.352	0.505	0.412	0.666
2019_02	0.239	0.529	0.329	0.611	0.339	0.523	0.407	0.667

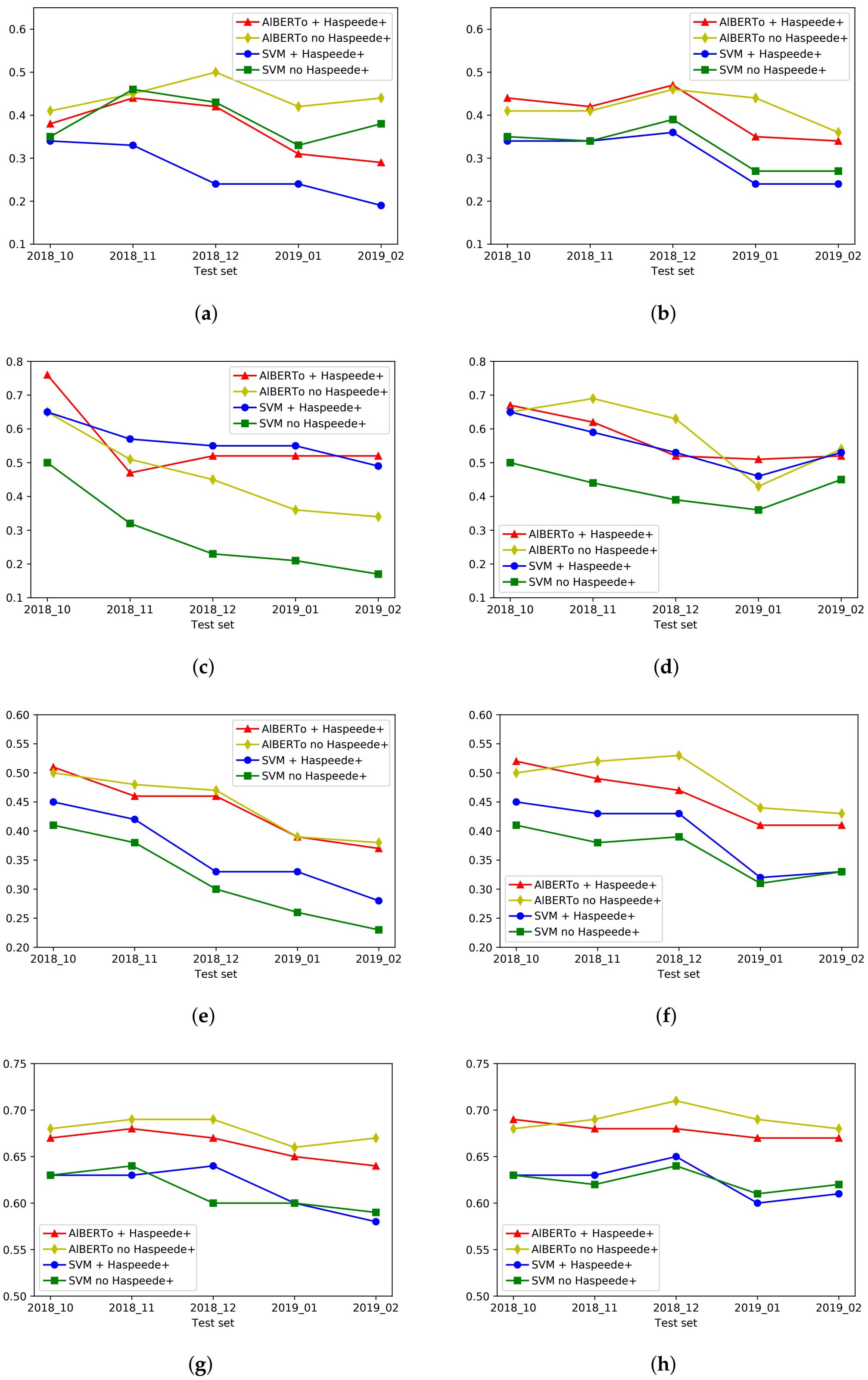
Our most meaningful results are presented in Figure 3f: AlBERTo overcomes the performance obtained using a fixed training set (Figure 2c). It can successfully mitigate the decay of the performance with the passage of time as shown in Figure 3g–h. Moreover, AlBERTo trained on an incremental dataset performs better than the same model trained on an incremental scheme built using only on the more recent data, and better than SVM as well.

We can observe that using both a sliding-window and incremental training strategy, the models' performance tends to reduce over time. Nevertheless, the drop in performance, in both the approaches, is smaller compared to the one obtained using a fixed training dataset. This observation demonstrates the importance of the diachronic training. This behavior is especially evident if we look at the F1 of the positive class, apart from small irregularities. As an example, the trend of both the F1-score shows an inversion around December in the incremental setup. An additional factor impacting the performance of all classifiers on the last two test sets is likely the lower relative rate of HS messages (see Table 2). However, other reasons concur in the specificity of these monthly samples, in particular lexical and topical features, as explained in the next section.

The strategy based on incremental training set generally works better than the one based on sliding window as a consequence of the largest amount of recent data available for training. The key to a successfully fine-tuning of AlBERTo is the use of data that are not too distant in time from the test set: we estimate a max value of six months. As proof of our claim, when adding Haspeede+, model performance tends to decrease. This behavior is a consequence of its internal algorithm that uses fine-tuning to focus the model on many specific and timely aspects. Consequently, older data addition can introduce noise that does not help the model to converge better. The SVM strategy has a similar behavior of AlBERTo when comparing the two strategies of training. The main difference is that SVM is more sensitive to the quantity of data than AlBERTo, and consequently, it performs better if Haspeede+ is included in the training set. As a general claim, we can then affirm that the best strategy to train models for hate speech detection is to use a large amount of data as updated as possible because both these aspects influence machine learning models.

Table 8 shows the results of the fixed and incremental windows experiments in comparison. In order to understand the significance of our results, we performed a paired Wilcoxon non-parametric test. This analysis shows statistical confidence for the results of the two different experiments for $p < 0.01$.

To support our hypothesis about the importance of updated data for reducing the negative influence of time factor on machine learning models, we decided to train both models on a new dataset injected with Haspeede+ data, which are temporally very distant from the test set data. We can observe that the performance of both models in this condition tend to decrease over times, which is a proof of our claim: an injection of timely distant data introduce a degree of noise that ultimately leads to a decrease of the model performance, in both cases similarly. All the results of this experiment and their statistical significance (tested as before with a Wilcoxon test) are listed in Table 9.



(**a**) (**b**)

(**c**) (**d**)

(**e**) (**f**)

(**g**) (**h**)

Figure 3. Evaluation of the models trained on a Sliding Windows (left columns) and Incremental dataset (right column). (**a**,**b**) Precision on the positive class. (**c**,**d**) Recall on the positive class. (**e**,**f**) F1-score on the positive class. (**g**,**h**) Macro-averaged F1-score.

Table 8. Comparison of the macro F1 scores between the fixed and incremental windows experiments.

Test Set	SVM no Haspeede+				AlBERTo no Haspeede+			
	Fixed	**Incremental**	**Δ**	***p*-Value**	**Fixed**	**Incremental**	**Δ**	***p*-Value**
2018_10	0.617	0.629	+0.012	4.1×10^{-18}	0.674	0.679	+0.005	7.1×10^{-7}
2018_11	0.579	0.624	+0.045	3.6×10^{-38}	0.642	.694	+0.052	3.4×10^{-14}
2018_12	0.578	0.636	+0.058	8.8×10^{-63}	0.628	0.704	+0.076	1.2×10^{-8}
2019_01	0.560	0.611	+0.051	2.3×10^{-56}	0.610	0.687	+0.077	8.8×10^{-11}
2019_02	0.517	0.624	+0.107	4.8×10^{-62}	0.587	0.679	+0.092	3.3×10^{-14}

Table 9. Wilcoxon Test *p*-values.

Sliding Window			
Months: Training->Test	**Macro-F Score Linear SVM**	**Macro-F Score BERT**	***p*-Value**
9->10	0.629	0.679	7.6×10^{-02}
10->11	0.639	0.686	7.5×10^{-10}
11->12	0.600	0.694	2.4×10^{-8}
12->1	0.598	0.661	1.8×10^{-2}
1->2	0.592	0.673	1.1×10^{-2}
Sliding Window + Haspeede+			
Months: Training->Test	**Macro-F Score Linear SVM**	**Macro-F Score BERT**	***p*-Value**
9->10	0.634	0.677	1.8×10^{-1}
10->11	0.628	0.684	3.7×10^{-16}
11->12	0.640	0.665	2.2×10^{-6}
12->1	0.603	0.649	7.3×10^{-7}
1->2	0.575	0.636	3.1×10^{-7}
Incremental Window + Haspeede+			
Months: Training->Test	**Macro-F Score Linear SVM**	**Macro-F Score BERT**	***p*-Value**
9->10	0.634	0.694	7.6×10^{-2}
9+10->11	0.633	0.684	8.4×10^{-8}
9+10+11->12	0.645	0.680	2.3×10^{-6}
9+10+11+12->1	0.603	0.666	1.3×10^{-6}
9+10+11+12+1->2	0.611	0.667	3.0×10^{-12}

Consequently, we can affirm that machine learning techniques are affected in performance by a bias consequent of the change of language over time in new text analyzed, especially in a domain of hate speech. The issue is strongly related to the amount of data provided at the model for the training phase, and consequently, the use of data updated and large enough is the best option for preserving good performance of an automatic hate speech detection model. In the event, it is difficult to obtain frequently enough updated data, a possible strategy to use for mitigating the issue is to use an incremental training set that merges old and new data in order to guarantee the model enough data for generalizing correctly and some new examples that include the updated vocabulary.

6. Lexical Analysis

To gain a more in-depth insight into the phenomena causing the prediction performance described in the previous section, we performed an additional set of experiments aiming at understanding the topics of discussion emerging from the data, and their diachronic properties. Our main statistical tool is the *weirdness index* [48], an automatic metric to retrieve words characteristic of a *special language* with respect to their typical usage.

In practice, given a *specialist* text corpus and a *general* text corpus, the weirdness index of a word is the ratio of its relative frequencies in the respective corpora. Calling w_s the frequency of the word w in the specialist language corpus, w_g the frequency of the word w in the general language corpus, and t_s and t_g the total count of words the specialist and general language corpora respectively, the weirdness index of w is computed as:

$$Weirdness(w) = \frac{w_s/t_s}{w_g/t_g}$$

When applied to an annotated corpus of hate speech, we expect that the words with high WI will reflect the most characteristic concepts in that corpus, those who distinguish it most from generic language. By analyzing the words with the highest weirdness index in each test set (treated as specialized corpora) against the training set Haspeede+ (treated as the general corpus), we aim at discovering patterns among the emerging topics that are novel with respect to the original training set. Table 10 shows the top ranked words by Weirdness Index from each of our test sets. Please note that words occurring only once in the data set were filtered out before the computation of the index. Indeed, at the top of each ranked list of words by weirdness, words appear that refer to specific events. For instance, the test set from January 2019 is dominated by the topic of the Sea Watch NGO ship and the refusal of the Italian government to let it enter their ports (https://en.wikipedia.org/wiki/Sea-Watch). In almost all cases, the topics emerging from the weirdness analysis are different from one month to the following. In rare occurrences, the echo of an event on social media spans two months, as is the case of the political discussion around the Global Compact for Migration pact (https://en.wikipedia.org/wiki/Global_Compact_for_Migration), observed among the top ranked words in November as well as December 2018.

Table 10. Top 20 words by Weirdness Index in each test set.

September 2018	October 2018	November 2018	December 2018	January 2019	February 2019
dalai	cialtronaggine	credito	strasburgo	sea	sea
lama	@giovanniproto67	global	global	47	#salvininonmollare
l'escamotage	all'opposizione	carte	@lavaligiadianna	#salvininonmollare	47
applicare	eurotassa	moavero	giuseppe	siracusa	recessione
slavi	#leu	#baobab	sea	#portichiusi	emirati
#deluca	l'illegalità	▪	:/	#restiamoumani	@danilotoninelli
ℹ	@gbongiorno66	assegni	@europarl_it	battisti	@openarms_it
@time	incompetente	ruspe	open	49	processare
magazine	#unhcr	@lavaligiadianna	versato	#giornatadellamemoria	#portichiusi
costituirsi	aste	flessibilità	@openarms_fund	valdese	@medhope_fcei
abramo	@tgrsicilia	polonia	venuto	olandese	2019
luisa	867	peschereccio	#bergoglio	totalmente	#bergoglio
ranieri	#voisapete	unhcr	antonio	#fakenews	tav
sfavore	avessimo	firmare	babbo	magistris	febbraio
gyatso	paladino	@baobabexp	emendamento	#cesarebattisti	@rescuemed
xiaomi	#iostoconmimmolucano	eletta	international	palermitani	#catania
profetessa	organizzava	#pakistan	praticano	disumane	@openarms_fund
giudea	riacesi	dell'onu	#manovra	tedesche	fazio
busto	donano	meningite	natale	claudio	#martina
asselborn	combinato	hiv	presepe	chiedendo	laureato

We then apply the weirdness index to the same sets in a different way, to gauge the topics most associated with the hateful language in the labeled dataset. The mechanism is straightforward: instead of comparing the relative frequencies of a word in a special language corpus (the test set, in the previous experiment) against a general language corpus (the training set), we compare the relative frequencies of a word as it occurs in the subset of the labeled datasets identified by one value of the label against its complement. We refer to such variant as *Polarized Weirdness Index* (PWI). Formally, consider a labeled corpus $C = \{(e_1, l_1), (e_2, l_2), ...\}$ where $e_i = \{w_1, w_2, ...\}$ is an instance of text, and l_i is the label associated with the text where e_i occurs, belonging to a fixed set L (e.g., $\{HS, not - HS\}$). The *polarized weirdness* of w with respect to the label $l*$ is the ratio of the relative frequency of w in the subset $\{e_i \in C : l_i = l*\}$ over the relative frequency of w in the subset $\{e_i \in C : l_i \neq l*\}$ The outcome of the calculation of the Polarized Weirdness index is again a ranking over the words contained in the subset of each test set identified by the hateful label. Words occurring only once in each test set were again filtered out before computing the index. High-PWI words from a class will give a strong indication of the most characteristic words to distinguish that class (e.g., hate speech) from its complement (e.g., not hate speech).

Following this analysis, whose results are shown in Table 11, we found many action verbs among the top-ranking words in all the test sets. Such verbs refer to negative, in particular criminal, actions such as *killing* or *robbing*, indicating a strong link between the topics emerging in the messages labeled as hateful and events in the news. However, the main verbs are different from month to month. For instance, verbs related to *drug dealing* are prevalent in November 2018, while verbs related to *rape* are relevant from October 2018 to January 2019 with a peak in December 2018, and verbs related to killing are mostly concentrated in December 2018.

Table 11. Top 20 words by Polarized Weirdness Index in each test set.

September 2018	October 2018	November 2018	December 2018	January 2019	February 2019
delinquere	parassiti	zingari	feccia	#primagliitaliani	incompatibile
zingari	stupratori	stupri	parassiti	👿	rotto
barconi	stuprare	parassiti	assassini	invasori	fanculo
auto	pamela	stuprano	negri	stupri	stupratori
biglietto	violentata	bambine	civiltà	#rai	vergognatevi
#stopinvasione	ns	💩	moderato	infami	esistono
hotel	strade	uccidono	cacciati	auto	ladri
calci	dell'islam	intanto	stupratori	pamela	etnie
clandestino	feccia	🤮	venire	visti	bus
famiglie	nomadi	cesso	infedeli	autoctoni	siriani
studenti	merde	ladri	💩	paghiamo	pensionati
modello	cani	#pakistan	#primagliitaliani	film	nullafacenti
assistenza	dobbiamo	etc	cancro	recessione	negri
#movimentonesti	farci	buonismo	onesti	spacciatori	l'invasione
ladri	abusivi	tramite	assassino	chiese	forze
feccia	assassini	strade	ospiti	invasione	nonni
subito	campi	moderato	rispetta	merde	maledetti
rapine	dovete	spaccio	#allah	ospite	bestie
pagano	mantenerli	diventata	#corano	tale	90
cinesi	rimpatriare	stupro	#stopinvasione	stuprata	pago

Finally, our considerations are confirmed by the chart in Figure 4, showing a simpler frequency-based analysis provided by Sketch Engine (https://www.sketchengine.eu/). Here, the vertical axis shows the frequency of the items in each test set relative to the average of all six sets. This score is higher than 100% when an item occurs more often than the average in a month, e.g., "compact" occurs almost four times the average in December 2018). The lemmas related to criminal activity, "rubare" (*to steal*), "stuprare" (*to rape*), and "uccidere" (*to kill*) show different patterns, likely linked to events in the news. The effect is even more prominent with more topical words, such as "porti" (*harbors*, referring to the Sea Watch event) and "compact" (from the aforementioned Global Compact), showing clear peaks in specific months.

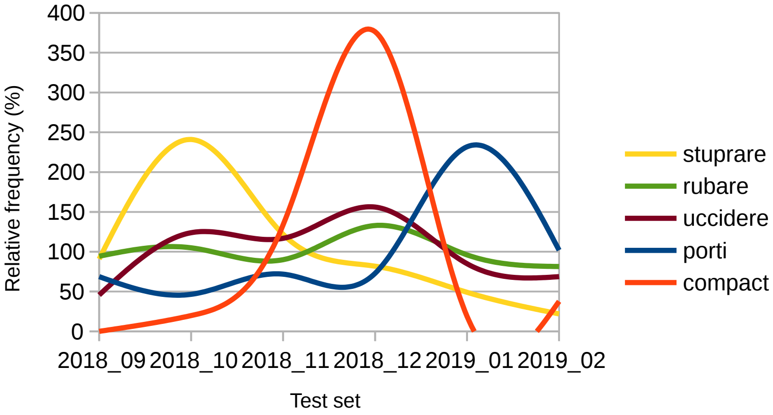


Figure 4. Relative frequencies of topical words and lemmas over time.

7. Conclusions and Future Work

In this work, we explored the temporal robustness of prediction systems for hate speech detection on Twitter. We evaluated the temporal robustness of different hate speech prediction systems, with respect to language and topic change over time (RQ1), by designing two different experiments: in the first case, we trained the models on data from a single month and tested it on the following month. In the second case, we injected information on the recent past (thus increasing the size of the training set) by using data from all the months preceding the one from which we draw the test sample (All codes and data are publicly available to the community here: https://github.com/komal83/timeofyourhatepaper). Unsurprisingly, injecting training data temporally closer to the test set sharply improves the prediction performance of AlBERTo compared with the SVM (partly answering RQ2), since the training data are very similar to the test data form a linguistic and topic perspective. On the contrary, our experiments show that increasing the size of the training set does not necessarily lead to equally improved performance.

To provide a more complete answer to RQ2, we also repeated the experiments adding a larger training set from a distant time span. Our results show how this setting has a beneficial effect on the SVM, but a negative effect on the performance of the transformer model. To gain a better understanding of the linguistic differences between our monthly samples, we also ran a statistical analysis of the topics from a temporal perspective. The analysis confirms that there is a relatively fast shift in topics in the online discourse, and this constitutes the main challenge to overcome in order to improve the robustness over time of the predicting systems for hate speech detection. We applied our methodology to a real Italian case study. However, the experimental design is agnostic with respect to the language. Therefore, the work can be expanded from a multilingual perspective, provided the development of suitable diachronic corpora. Moreover, we would like to investigate more the possibility to use strategies of data balancing. Our annotated data are naturally very unbalanced, with non-hate speech examples representing most of the dataset. It is commonly known that the performance of machine learning approaches is strongly influenced by the class unbalance, and consequently, we would like to investigate the impact of automatic balancing techniques or the addition of new training data on the robustness observed in the models we analyzed.

Author Contributions: All authors contributed equally to the work. All authors have read and agreed to the published version of the manuscript.

Funding: The work of all the authors was partially funded by Italian Ministry of Labor (Contro l'odio: tecnologie informatiche, percorsi formativi e storytelling partecipativo per combattere l'intolleranza, avviso n.1/2017 per il finanziamento di iniziative e progetti di rilevanza nazionale ai sensi dell'art. 72 del decreto legislativo 3 luglio 2017, n. 117 - anno 2017). The work of Valerio Basile and Viviana Patti was partially funded by the project "Be Positive!" (2020-22 under the "Google.org Impact Challenge on Safety" call). The work of Marco Polignano is funded by the project "DECiSION" (code: BQS5153), under the Apulian INNONETWORK program, Italy.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Capozzi, A.T.; Lai, M.; Basile, V.; Poletto, F.; Sanguinetti, M.; Bosco, C.; Patti, V.; Ruffo, G.; Musto, C.; Polignano, M.; et al. Computational linguistics against hate: Hate speech detection and visualization on social media in the "Contro L'Odio" project. In Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2019), Bari, Italy, 13–15 November 2019; CEUR Workshop Proceedings; CEUR-WS.org: Bari, Italy, 2019; Volume 2481, pp. 1–6.
2. Menini, S.; Moretti, G.; Corazza, M.; Cabrio, E.; Tonelli, S.; Villata, S. A System to Monitor Cyberbullying based on Message Classification and Social Network Analysis. In Proceedings of the Third Workshop on Abusive Language Online, Florence, Italy, 1 August 2019; Association for Computational Linguistics: Florence, Italy, 2019; pp. 105–110. [CrossRef]

3. Paschalides, D.; Stephanidis, D.; Andreou, A.; Orphanou, K.; Pallis, G.; Dikaiakos, M.D.; Markatos, E. MANDOLA: A Big-Data Processing and Visualization Platform for Monitoring and Detecting Online Hate Speech. *ACM Trans. Internet Technol.* **2020**, *20*. [CrossRef]
4. EU Commission. *Code of Conduct on Countering Illegal Hate Speech Online*; European Commission, Bruxelles, Belgium, 2016.
5. Shen, Q.; Rose, C. The Discourse of Online Content Moderation: Investigating Polarized User Responses to Changes in Reddit's Quarantine Policy. In Proceedings of the Third Workshop on Abusive Language Online, Florence, Italy, 1 August 2019; Association for Computational Linguistics: Florence, Italy, 2019; pp. 58–69. [CrossRef]
6. Florio, K.; Basile, V.; Lai, M.; Patti, V. Leveraging Hate Speech Detection to Investigate Immigration-related Phenomena in Italy. In Proceedings of the 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW), Cambridge, UK, 3–6 September 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–7. [CrossRef]
7. Olteanu, A.; Castillo, C.; Boy, J.; Varshney, K.R. The effect of extremist violence on hateful speech online. In Proceedings of the Twelfth International Conference on Web and Social Media (ICWSM 2018), Stanford, CA, USA, 25–28 June 2018; AAAI Press: Menlo Park, CA, USA, 2018; pp. 221–230.
8. Fortuna, P.; Nunes, S. A survey on automatic detection of hate speech in text. *ACM Comput. Surv. (CSUR)* **2018**, *51*, 85. [CrossRef]
9. MacAvaney, S.; Yao, H.R.; Yang, E.; Russell, K.; Goharian, N.; Frieder, O. Hate speech detection: Challenges and solutions. *PLoS ONE* **2019**, *14*, e0221152. [CrossRef] [PubMed]
10. Basile, V.; Bosco, C.; Fersini, E.; Nozza, D.; Patti, V.; Pardo, F.M.R.; Rosso, P.; Sanguinetti, M. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In Proceedings of the 13th International Workshop on Semantic Evaluation, Minneapolis, MN, USA, 6–7 June 2019; Association for Computational Linguistics: Minneapolis, MN, USA, 2019; pp. 54–63. [CrossRef]
11. Struß, J.M.; Siegel, M.; Ruppenhofer, J.; Wiegand, M.; Klenner, M. Overview of GermEval Task 2, 2019 Shared Task on the Identification of Offensive Language. In Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019), Erlangen, Germany, 8–11 October 2019; German Society for Computational Linguistics & Language Technology: Erlangen, Germany, 2019; pp. 354–365.
12. Bosco, C.; Felice, D.; Poletto, F.; Sanguinetti, M.; Maurizio, T. Overview of the EVALITA 2018 Hate Speech Detection Task. In Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Final Workshop (EVALITA 2018), Turin, Italy, 12–13 December 2018; CEUR Workshop Proceedings; CEUR-WS.org: Torino, Italy, 2018; Volume 2263, pp. 1–9.
13. Corazza, M.; Menini, S.; Cabrio, E.; Tonelli, S.; Villata, S. A Multilingual Evaluation for Online Hate Speech Detection. *ACM Trans. Internet Technol.* **2020**, *20*. [CrossRef]
14. Pamungkas, E.W.; Patti, V. Cross-domain and Cross-lingual Abusive Language Detection: A Hybrid Approach with Deep Learning and a Multilingual Lexicon. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, Florence, Italy, 28 July–2 August 2019; Association for Computational Linguistics: Florence, Italy, 2019; pp. 363–370. [CrossRef]
15. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; Association for Computational Linguistics: Minneapolis, MN, USA, 2019; Volume 1 (Long and Short Papers), pp. 4171–4186.
16. Dai, A.M.; Le, Q.V. Semi-supervised Sequence Learning. In Proceedings of the Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, Montreal, QC, Canada, 7–12 December 2015; Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R., Eds.; Curran Associates, Inc.: Dutchess County, NY, USA, 2015; pp. 3079–3087.
17. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep Contextualized Word Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, LA, USA, 1–6 June 2018; Walker, M.A., Ji, H., Stent, A., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; Volume 1 (Long Papers), pp. 2227–2237. [CrossRef]

18. Howard, J.; Ruder, S. Universal Language Model Fine-tuning for Text Classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, 15–20 July 2018; Volume 1: Long Papers; Gurevych, I., Miyao, Y., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 328–339. [CrossRef]
19. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. 2018. Available online: https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf (accessed on 6 June 2020).
20. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.U.; Polosukhin, I. Attention is All you Need. In *Advances in Neural Information Processing Systems 30*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Long Beach, CA, USA, 2017; pp. 5998–6008.
21. Polignano, M.; Basile, P.; de Gemmis, M.; Semeraro, G.; Basile, V. 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), Bari, Italy, 13–15 November 2019; CEUR Workshop Proceedings; CEUR-WS.org: Bari, Italy, 2019; Volume 2481.
22. Basile, V.; Lai, M.; Sanguinetti, M. Long-term Social Media Data Collection at the University of Turin. In Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Torino, Italy, 10–12 December 2018; CEUR Workshop Proceedings; CEUR-WS.org: Torino, Italy, 2018; Volume 2253, pp. 1–6.
23. Polignano, M.; Basile, P.; de Gemmis, M.; Semeraro, G. Hate Speech Detection through AlBERTo Italian Language Understanding Model. In Proceedings of the 3rd Workshop on Natural Language for Artificial Intelligence co-located with AI*IA 2019, Rende, Italy, 19–22 November 2019; CEUR Workshop Proceedings; CEUR-WS.org: Rende, Italy, 2019; Volume 2521.
24. Tahmasebi, N.; Borin, L.; Jatowt, A. Survey of Computational Approaches to Lexical Semantic Change. *arXiv* **2019**, arXiv:1811.06278v2.
25. Goel, R.; Soni, S.; Goyal, N.; Paparrizos, J.; Wallach, H.; Diaz, F.; Eisenstein, J. The social dynamics of language change in online networks. In Proceedings of the International Conference on Social Informatics, Bellevue, WA, USA, 11–14 November 2016; Lecture Notes in Computer Science; Springer: Berlin, Germany, 2016; Volume 10046, pp. 41–57. [CrossRef]
26. Clyne, M.; Clyne, M.G.; Michael, C. *Dynamics of Language Contact: English and Immigrant Languages*; Cambridge University Press: Cambridge, UK, 2003.
27. Lieberman, E.; Michel, J.B.; Jackson, J.; Tang, T.; Nowak, M.A. Quantifying the evolutionary dynamics of language. *Nature* **2007**, *449*, 713–716. [CrossRef] [PubMed]
28. Donoso, G.; Sánchez, D. Dialectometric analysis of language variation in Twitter. In Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial), Valencia, Spain, 3 April 2017; Association for Computational Linguistics: Valencia, Spain, 2017; pp. 16–25. [CrossRef]
29. Basile, P.; Caputo, A.; Semeraro, G. TRI: A tool for the diachronic analysis of large corpora and social media. In Proceedings of the 7th AIUCD Annual Conference Cultural Heritage in the Digital Age. Memory, Humanities and Technologies, Bari, Italy, 30 January–2 February 2018.
30. Jaidka, K.; Chhaya, N.; Ungar, L. Diachronic degradation of language models: Insights from social media. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Melbourne, Australia, 15–20 July 2018; Association for Computational Linguistics: Melbourne, Australia, 2018; pp. 195–200.
31. Hellrich, J. *Word Embeddings: Reliability & Semantic Change*; IOS Press: Amsterdam, The Netherlands, 2019.
32. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
33. Kao, A.; Poteet, S.R. *Natural lAnguage Processing and Text Mining*; Springer: Berlin, Germany, 2007.
34. Vangara, R.V.B.; Vangara, S.P.; Thirupathur, V.K. A Survey on Natural Language Processing in context with Machine Learning. *Int. J. Anal. Exp. Modal Anal.* **2020**, *XII*, 1390–1395.
35. Ramos, J. Using tf-idf to determine word relevance in document queries. In Proceedings of the First Instructional Conference on Machine Learning, Piscataway, NJ, USA, 3–8 December 2003; Volume 242, pp. 133–142.

36. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and Their Compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems—Volume 2, NIPS'13, Lake Tahoe, NV, USA, 5–10 December 2013; Curran Associates Inc.: Red Hook, NY, USA, 2013; pp. 3111–3119.
37. Cherkassky, V.; Ma, Y. Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Netw.* **2004**, *17*, 113–126. [CrossRef]
38. Gopi, A.P.; Jyothi, R.N.S.; Narayana, V.L.; Sandeep, K.S. Classification of tweets data based on polarity using improved RBF kernel of SVM. *Int. J. Inf. Technol.* **2020**, 1–16.
39. Kaur, G.; Kaur, E.P. Novel approach to text classification by SVM-RBF kernel and linear SVC. *Int. J. Adv. Res. Ideas Innov. Technol.* **2017**, *3*, 1014–1047.
40. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog* **2019**, *1*, 9.
41. Zhu, Y.; Kiros, R.; Zemel, R.; Salakhutdinov, R.; Urtasun, R.; Torralba, A.; Fidler, S. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In Proceedings of the IEEE International Conference on Computer Vision, Las Condes, Chile, 11–18 December 2015; pp. 19–27.
42. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF models for sequence tagging. *arXiv* **2015**, arXiv:1508.01991.
43. Polignano, M.; Basile, V.; Basile, P.; de Gemmis, M.; Semeraro, G. AlBERTo: Modeling Italian Social Media Language with BERT. *Ital. J. Comput. Linguist.* **2019**, *2*, 11–32.
44. Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–27. [CrossRef]
45. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. Tensorflow: A system for large-scale machine learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI'16), Savannah, GA, USA, 2–4 November 2016; pp. 265–283.
46. Poletto, F.; Stranisci, M.; Sanguinetti, M.; Patti, V.; Bosco, C. Hate speech annotation: Analysis of an Italian Twitter corpus. In Proceedings of the 4th Italian Conference on Computational Linguistics (CLiC-it 2017), Rome, Italy, 11–13 December 2017; CEUR Workshop Proceedings; CEUR-WS.org: Rome, Italy, 2017; Volume 2006, pp. 1–6.
47. Sanguinetti, M.; Poletto, F.; Bosco, C.; Patti, V.; Marco, S. An italian Twitter corpus of hate speech against immigrants. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, 7–12 May 2018; European Language Resources Association (ELRA): Miyazaki, Japan, 2018; pp. 1–8.
48. Ahmad, K.; Gillam, L.; Tostevin, L. *University of Surrey Participation in TREC 8: Weirdness Indexing for Logical Document Extrapolation and Retrieval (WILDER)*; TREC: Gaithersburg, MD, USA, 1999; pp. 1–8.