Recognition of Epidemic Cases in Social Web texts

Eleftherios Alexiou, Apostolos Antonakakis, Nemanja Jevtic, Georgios Sideras, Eftichia Farmaki, Sofronia Foutsitzi, Katia Kermanidis

Recognition of epidemic cases in social web texts

Alexiou Eleftherios[1], Antonakakis Apostolos[1], Jevtic Nemanja[1], Sideras Georgios[1], Farmaki Eftichia[1], Foutsitzi Sofronia[1], and Kermanidis Katia - Lida[1]

Ionian University, Department of Informatics, 7 Tsirigoti Square, Corfu, Greece
{p17alex, p17anto2, p17pevt, p17side, p17farm, p17fout, kerman}@ionio.gr

Abstract. Since December 2019, Covid-19 has been spreading rapidly across the world. Unsurprisingly, conversation in social networks about Covid-19 is increasing as well. The aim of this study is to identify tentative Covid-19 infection cases through social networks and, specifically, on Twitter, using machine learning techniques. Tweets were collected using the data set "Covid-19 Twitter", between November 1, 2020 and December 30, 2020, and manually marked by the authors of this study as positive (describing a tentative Covid-19 infection case) or negative (pertaining to any other Covid-19 related issue) cases of Covid-19, creating a smaller but more focused dataset. This study was conducted in three phases: a. data collection and data cleaning, b. processing and analysis of tweets by machine learning techniques, and c. evaluation and qualitative/quantitative analysis of the achieved results. The implementation was based on Gradient Boosting Decision Trees, Support Vector Machines (SVM) and Deep Learning algorithms.

Keywords: covid · machine learning · twitter · natural language processing

1 Introduction

Social media is a way to connect people around the world with extreme ease. As a result, users share a lot of information about their social life, transforming social channels into a powerful data collection tool for scientific purposes. During the current state of the Covid-19 pandemic global spreading, the processing of textual information shared on social media can help us identify tentative infection cases. This knowledge can help authorities to timely analyse activities and plan optimal solutions for addressing the virus outbreak.

Bearing this in mind, this study aims to use collected data from the social media platform Twitter [3], based on basic keywords related to the current pandemic outbreak of the virus SARS-CoV-2, and automatically identify positive (describing a tentative infection case) from negative (pertaining to any other Covid-related issue) tweets. The automatic identification was achieved with machine learning algorithms such as Gradient Boosting Decision Trees, Support Vector Machines (SVM) and Deep Learning, using RapidMiner Studio[1], the

[1] https://rapidminer.com/products/studio/

machine learning workbench. In order to implement this framework, data was firstly collected from GitHub repositories and filtered through new keywords, creating a new dataset, more focused towards the needs of this study. The new dataset was then annotated, whether each tweet was referencing a probabilistic case or not. Finally, RapidMiner was used as a machine learning interface for Gradient Boosting Decision Tree, Support Vector Machines (SVM) and Deep Learning algorithms.

The remainder of this paper is organized as follows: Section 2 describes the related work that helped define the purpose of this study. Section 3 presents the selection and the usage of the data. Section 4 explains the methods, the algorithms and the evaluation metrics employed, Section 5 displays and explains the results that were achieved. Finally, Section 6 concludes the work, and suggests directions for future research.

2 Related Work

This section covers the literature review which is related to Twitter, NLP (Natural Language Processing) and machine learning methods for text mining. Most of the studies collect data in the English language [10], however, there are approaches based on multilingual data sets [3][18]. Data sets are filtered with keywords which pertain to Covid-19 ("covid-19", "coronavirus " "epidemic", "pandemic", "fever") [3], and are collected from Twitter [10][16][12] and other social networks, like weibo [17]. Wakamiya et. al. [19] chose to manually write a large amount of twitter-like messages in multiple languages, labelling them for more than one disease. Sick Posts are defined as posts that report any symptoms or diagnoses that are likely to be related to COVID-19, based on published research [17]. For data analysis, machine learning techniques have been used to classify whether a text is related to the pandemic. Linear Regression, Naive Bayes Classification, Logistic Regression, k-nearest neighbours are some of the methods that studies have used to identify posts on social media which describe a virus infection case [16].

The focus of this paper is on identifying positive test cases by user announcements on social media by creating a new dataset based on an already existing one [3]. This dataset, albeit smaller, is less generic and more suited for the purposes of this study.

3 Data

In the next sub-sections, the methodology for data collection and annotation is presented.

3.1 Data Collection

Initially, the data in the present work is gathered from "Covid-19 Twitter" [3], which is available on the COVID-19-TweetIDs GitHub repository. This dataset

is using Twitter's search API and Tweepy to collect real time multilingual tweets published from verified users that mention specific keywords. These tweets are compartmentalized in file names that follow the same structure (year-month-day-hour). Afterwards, Hydrator[2] was used, a tool for hydrating tweet IDs, as in collecting and storing data and metadata of each tweet. A Python script, implemented by the authors of this paper was used in order to filter out languages other than English, and to filter the tweets through new keywords, chosen for the purpose of creating the new dataset. Table 1 shows the keywords used.

Table 1. Keywords used to filter the existing dataset.

Positive	tested	self-isolation	dry cough
i have covid	symptoms	has covid	covid test

3.2 Data Annotation

The dataset was annotated by the authors, with 2 annotators on each tweet and in constant communication. Tweets that describe tentative Covid-19 infection cases related to the author himself or to his surroundings (family, friends, colleagues or even famous people) were labeled as **True**, and every tweet that is Covid-19 related, but does not describe a tentative infection as **False**. This resulted in a dataset of nearly 4000 entries.

Some examples of the annotation follow:

- Weekly covid tests are a drag but it brings a piece of mind: **False**
- Positive of covid: seeing more kids walking home after school with a parent: **False**
- Nothing will traumatize you like receiving a text informing you that, one of the learners in your child's class was tested positive of covid-19!! text i received from my daughter's teacher yesterday: **True**

3.3 Data Transformation

Afterwards, the dataset was imported into RapidMiner. Before actually applying any classification algorithm, we applied a few operators available in the Text Processing extension of RapidMiner on our dataset:

- **Tokenization:** The process of splitting a text into a sequence of tokens, spanning a word each.

[2] https://github.com/DocNow/hydrator

- **Stemming:** The process of combining all words of the same root into one. For the purposes of this study, we used the Porter Stemmer [14] for English words, an algorithm that applies an iterative and rule-based replacement of word suffixes.
- **Filter Stopwords:** Removes english stopwords, words that don't convey important information, such as articles, from the dataset.
- **TF-IDF:** A weight factor for each of the tokens based on the term frequency, offset by the number of tweets that contain the word.
- **Pruning**: Removes tokens that appear only once in the entire dataset.

The result was a dataset of 3947 attributes, word roots with a weight variable representing their importance in the overall context, some of which are language usage conventions of the Twitter community (hashtags, abbreviations or emojis), or even spelling mistakes. The machine learning algorithms that were used did a adequate job of ruling those out.

4 Methods

4.1 Algorithms

RapidMiner gives a handful of machine learning services and tools to work with. After the pre-processing of the data, several algorithms were experimented with for automatic classification. More specifically, RapidMiner's stock implementations of Gradient Boosted Decision Trees, Support Vector Machines and Deep Learning classifiers were employed. A short briefing follows on each of them in the next sub-sections.

Gradient Boosted Decision Trees (GBDT) [4], is a well known algorithm used in machine learning, which plays a vital role in multi-class classification [11], click prediction [15], and the ability to rank its learnings [2]. It starts with an ensemble of weak trees, gradually increasing their estimation accuracy through Boosting [9]. While boosting trees increases their accuracy, it also decreases speed and human interpretability. The gradient boosting method generalizes tree boosting to minimize these issues. RapidMiner's implementation of GBDT is based on the H2O framework. Contrary to the original implementation of H2O, which uses a distributed cluster of nodes representing trees [13], RapidMiner creates a 1-node cluster, supporting parallelism through threads [6].

Support Vector Machines (SVM) Algorithms like support vector machines can be used for both regression and classification. Classification is derived from the statistical learning theory [1]. To give an illustration, a set of already trained examples are marked in two categories and then the SVM assigns new input into one category based on his algorithm thus making it a non-probabilistic binary linear classifier. In other words, it maps the training examples to points in the

threshold space between the two categories, in order to maximize the gap between them. When new examples are imported, they are placed onto the same space, and then they are categorized into one category based on the side of the gap on which they fall. In addition to this, the SVM algorithm can also perform a non-linear classification with the help of kernel functions. Then again, the mapping is happening implicitly into high-dimensional feature spaces. RapidMiner's implementation of SVM is based on the internal Java implementation mySVM [8].

Deep Learning (DL) is not one single algorithm, but a family of different machine learning algorithms based on neural networks. Its differentiating attribute is the fact that there are many layers involved, each responsible for learning one piece of the model knowledge. The learning method can be either supervised, semi-supervised or unsupervised. Getting into the specifics, a deep neural network always consists of the following components: neurons, synapses, weights, biases and functions. They try to mimic a human brain and that's where the name comes from. Ultimately, the DNNs are feedforward networks, where data flows from the first layer or the input layer to multiple hidden layers connected with each other, as a map with different weights on each connection. The weights are initialized with uniform random values, but these are updated during the training process until the network achieves optimal performance. More specifically, if the network does not recognize the pattern to be learned correctly, the weights are adjusted using back propagation, and the input is fed again into the network, defining a new learning epoch [9]. RapidMiner's implementation of Deep Learning is based on the H2O framework. It creates a 1-node cluster and supports parallelism through threads [5].

4.2 Performance Metrics

Three metrics were used to evaluate the algorithms: accuracy, precision and recall. Firstly, the results are loaded on a 2×2 confusion matrix with the following measures [7]:

- **True Positives:** Correctly identified positive examples.
- **False Positives:** Incorrectly identified positive examples.
- **True Negatives:** Correctly identified negative examples.
- **False Negatives:** Incorrectly identified negative examples.

Then, the metrics are calculated:

- **Accuracy:**
$$\frac{TP+TN}{TP+FP+FN+TN}$$
- **Precision:**
$$\frac{TP}{TP+FP}$$
- **Recall:**
$$\frac{TP}{TP+FN}$$

4.3 Optimal Parameters

Table 2 presents the optimal parameters found.

Table 2. The optimal parameters of each algorithm

Support Vector Machines	**Kernel Type** **C**	Dot 0
Gradient Boosted Trees	**Number Of Trees** **Maximal Depth** **Learning Rate**	50 8 0.01
Deep Learning	**Activation** **Hidden Layers** **HL 1 Size** **HL 2 Size** **Epochs**	Tanh 2 200 100 50

In RapidMiner, the kernel type Dot in SVMs is the equivalent of Linear [8], and the activation Tanh in DL the equivalent of a shifted and scaled Sigmoid [5]. In addition, in RapidMiner, the output layer of DL uses Softmax activation by default.

5 Results

As seen in Table 3, the Gradient Boosted Decision Trees algorithm has better accuracy and better recall, meaning that it identified the most True labels, but it identified as True many False labels in comparison to Deep Learning. Deep Learning follows with a better Precision but a worse Accuracy, which means that it was more selective in its True identification.

Table 3. Effectiveness of the algorithms

Operator	**Accuracy**	**Precision**	**Recall**
Gradient Boosted Trees	80.93%	75.81%	85.10%
Support Vector Machines	75.58%	74.16%	70.60%
Deep Learning	78.50%	77.07%	74.91%

In RapidMiner, the Result Analysis of SVMs outputs the weight towards a True and a False identification, while the GBDT and DL equivalent outputs

only the Importance Factor, i.e. the overall weight without an orientation towards True or False identification. The Weight and Importance Factors follow in Table 4.

Table 4. Weight and Importance Factors

SVM		GBDT	DL
Positive	**Negative**	**Importance Factor**	
posit	neg	posit	posit
ha	get	test	covid
i	symptom	covid	test
player	free	fals	i
mom	line	symptom	neg
week	vaccin	i	ha

The greyed out words are words used to filter the dataset

The keyword symptom is a negative weight factor meaning that when someone mentions symptoms he has, it is usually in the context of getting tested because of it, or expressing relief when he gets tested negative. This is also supported by the fact that the keyword "test" exists in both algorithms' most important factors, but it doesn't appear as a positive or a negative factor. The keywords "player" and "mom" are positive weight factors, because when someone refers to family members or athletes in a "Covid-19" context, it is usually to announce that they tested positive.

6 Conclusions and Future Work

In this study, a novel approach is described for automatically identifying tweet text that describes a tentative Covid-19 infection. The models and algorithms used here can be found in most of the related work. Our contribution lies mainly in the labelling of the already existing dataset, resulting in a new dataset, and in making the basic implementation of a classification model for monitoring the self-reporting in the pandemic, as well as analysing the results' weight factors. Furthermore, if data is gathered from different social media sources (e.g. Facebook, Instagram, LinkedIn) and more generic keywords are used, more accurate classification may be achieved, especially by Deep Learning and Neural Networks in general, and medical intervention as well as government regulations can be performed on time.

Future work may include the revaluation of used algorithms, as well an expanded dataset. Use of the model presented could also be used to examine the correlation of self-reporting on social media in an area and the area wide positive cases. Integration in an external monitoring, visualisation and geospatial analysis application should also be considered.

References

1. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory. p. 144–152. Association for Computing Machinery (1992). https://doi.org/10.1145/130385.130401
2. Burges, C.J.: From ranknet to lambdarank to lambdamart: An overview. Tech. Rep. MSR-TR-2010-82 (2010), https://www.microsoft.com/en-us/research/publication/from-ranknet-to-lambdarank-to-lambdamart-an-overview/
3. Chen, E., Lerman, K., Ferrara, E.: Tracking Social Media Discourse About the COVID-19 Pandemic: Development of a Public Coronavirus Twitter Data Set. JMIR Public Health and Surveillance **6**(2), e19273 (2020). https://doi.org/10.2196/19273
4. Friedman, J.H.: Greedy Function Approximation: A Gradient Boosting Machine. The Annals of Statistics **29**(5) (2001). https://doi.org/10.1214/aos/1013203451
5. GmbH, R.: Deep Learning, https://docs.rapidminer.com/latest/studio/operators/modeling/predictive/neural_nets/deep_learning.html
6. GmbH, R.: Gradient Boosted Trees, https://docs.rapidminer.com/latest/studio/operators/modeling/predictive/trees/gradient_boosted_trees.html
7. GmbH, R.: Performance (Binominal Classification), https://docs.rapidminer.com/latest/studio/operators/validation/performance/predictive/performance_binominal_classification.html
8. GmbH, R.: Support Vector Machine, https://docs.rapidminer.com/latest/studio/operators/modeling/predictive/support_vector_machines/support_vector_machine.html
9. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition, pp. 337–380. Springer, 2nd edn. (2021)
10. Kabir, M.Y., Madria, S.: Coronavis: A real-time covid-19 tweets data analyzer and data repository (2020), preprint available at https://arxiv.org/abs/2004.13932
11. Li, P.: Robust logitboost and adaptive base class (abc) logitboost. In: Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence. p. 302–311. UAI'10, AUAI Press, Arlington, Virginia, USA (2010)
12. Mackey, T., Purushothaman, V., Li, J., Shah, N., Nali, M., Bardier, C., Liang, B., Cai, M., Cuomo, R.: Machine Learning to Detect Self-Reporting of Symptoms, Testing Access, and Recovery Associated With COVID-19 on Twitter: Retrospective Big Data Infoveillance Study. JMIR Public Health and Surveillance **6**(2), e19509 (2020). https://doi.org/10.2196/19509
13. Malohlava, M., Candel, A.: Gradient Boosting Machine with H2O (2021), https://www.h2o.ai/resources/booklet/gradient-boosting-machine-with-h2o/
14. Porter, M.: An algorithm for suffix stripping. Program **14**(3), 130–137 (1980). https://doi.org/10.1108/eb046814
15. Richardson, M., Dominowska, E., Ragno, R.: Predicting clicks: Estimating the click-through rate for new ads. In: Proceedings of the 16th International World Wide Web Conference(WWW-2007) (January 2007), https://www.microsoft.com/en-us/research/publication/predicting-clicks-estimating-the-click-through-rate-for-new-ads/
16. Samuel, J., Ali, G.G.M.N., Rahman, M.M., Esawi, E., Samuel, Y.: COVID-19 Public Sentiment Insights and Machine Learning for Tweets Classification. SSRN Electronic Journal (2020). https://doi.org/10.2139/ssrn.3584990

17. Shen, C., Chen, A., Luo, C., Zhang, J., Feng, B., Liao, W.: Using Reports of Symptoms and Diagnoses on Social Media to Predict COVID-19 Case Counts in Mainland China: Observational Infoveillance Study. Journal of Medical Internet Research **22**(5), e19421 (2020). https://doi.org/10.2196/19421
18. Singh, L., Bansal, S., Bode, L., Budak, C., Chi, G., Kawintiranon, K., Padden, C., Vanarsdall, R., Vraga, E., Wang, Y.: A first look at covid-19 information and misinformation sharing on twitter (2020), preprint available at https://arxiv.org/abs/2003.13907
19. Wakamiya, S., Morita, M., Kano, Y., Ohkuma, T., Aramaki, E.: Overview of the ntcir-13: Medweb task. In: Proceedings of the 13th NTCIR Conference on Evaluation of Information Access Technologies (2017)