Analysis of COVID-19 Data

Alessandra Amato¹, Giovanni Cozzolino²(✉), Alessandro Maisto³, and Serena Pelosi³

¹ University of Napoli Federico II, Naples, Italy
alessandra.amato@studenti.unina.it
² DIETI, University of Napoli Federico II, Via Claudio 21, Naples, Italy
giovanni.cozzolino@unina.it
³ DISPC, University of Salerno, Via Giovanni Paolo II, 132, Fisciano, SA, Italy
{amaisto,spelosi}@unisa.it

Abstract. A lot of research has been done during the first months of 2020 regarding the Covid-19. Researchers of different fields worked and cooperated to understand the virus better, in order to manage the pandemic and to model its spread. A series of tools have been developed in this sense, but there is a lack of work with regards to what has been developed from the scientific community. We would like to, at least partially, summarise the results obtained so far by analysing some of the published papers on the matter. To achieve such a result, we are going to use different python libraries that allow analysing texts. The entire work has been done with python on the Google Colaboratory platform.

1 Introduction

Analyse the results of thousands of researchers around the world about the Covid-19 is not an easy task for anyone, and it would be far easier if there could be a summary of the main points reached by the scientific community [1].

We propose an analysis of COVID related literature, aiming at extracting information and classify papers by using their significant topic and results. Not all the features that we aimed to include in the analysis are considered since some hiccups were met given the amount of data and the hardware/software limitations in which we operated [2]. The obtained results can constitute a good starting point for a future, more comprehensive analysis [3–6] and further exploitation, such as suggestions systems [7,8].

2 Data and Tools

The data was retrieved from the website of the European Bioinformatics Institute (EBI) that is part of the European Molecular Biology Laboratory (EMBL), Europe’s flagship laboratory for the life sciences. From this website, we retrieved about 73000 articles of various types, considering 60 different attributes. We used the attributes in order to select a subset of the data so that we could consider:
Published journal articles; 
in English; 
open access; 
with PDF available

Out of this subset of the data, we decided to concentrate our analysis only on the articles that were cited more than 100 times [9–11]. The retrieval has been conducted using the requests and pandas libraries of python. The data was in JSON format that was first converted to CSV using pandas and then to string using PyPDF2. This last operation was possible by using the DOI of each open-access article to download their PDFs. This task was achieved by using the scidownl library, which lets users download papers from the Sci-Hub repository. Finally, they are converted to plain text using the PyPDF2. Since PyPDF2 most of the times make mistakes in reading the PDFs, some articles were corrupted. The resulting articles used in the analysis are 326. For the core, analysis tasks required a variety of libraries in python for different reasons. The nltk library was used in order to both clean the data and analyse it. We also used regular expressions to clean the data further, implemented in python with the re library. Lastly, we used the enchant library to check for the validity of the words, thus extrapolated and its suggest method to try to correct the corrupted words [12].

To plot the data the libraries matplotlib, seaborn and networkx were used. To facilitate the representation of the results the library collections was used.

2.1 Preparing the Data

Before going into the core of the analysis, we show how we proceeded to clean the data for the subsequent analysis. We used regular expressions in order to remove from the papers URLs and words that were not helpful for the review, such as all the words that contain the terms research, publish, scholar and some other composed words that belonged to the header of the papers or to the preamble as well as all the digits.

After having applied these expressions we removed the stopwords using the ones that were given from the nltk library. In addition to the removal of the stop words, we decided to remove other words that occurred frequently, but that were not considered useful for the goal of the analysis. The list of the words we decided to exclude from the papers is as follows: ‘covid19’, ‘covid’, ‘coronavirus’, ‘covid2019’, ‘virus’, ‘covid-19’, ‘human’, ‘may’, ‘also’, ‘version’, ‘article’, ‘acad’, ‘material’, ‘available’, ‘online’, ‘usa’, ‘received’, ‘proc’, ‘natl’, ‘sci’, ‘limited’, ‘right’, ‘reserved’, ‘national’, ‘institute’, ‘health’, ‘university’, ‘department’, ‘nature’, ‘volume’, ‘issue’, ‘fig’, ‘figure’.

After removing the words cited above, we checked if the remaining words were meaningful, that is to say, check if they were in the English vocabulary. This step was necessary for two reasons: we did not want words that had no interpretable meaning; the PyPDF2 library could have mixed up some words.

This last step was pursued using the word module of the nltk library To compare the results that we obtained from the use of these words with a better
representation of the content of the articles, we decided to lemmatise them. The lemmas, that are the dictionary or citation form of a word or a set of words, were found using the WordNetLemmatizer of the nltk library. With these data we proceeded with our analysis.

3 Absolute Frequencies and N-Grams

From the frequencies of the words and the lemmas, in particular, we would like to see if it would be possible to find some information about the coronavirus and the state of the research on the matter. We start by looking only at the absolute frequencies of words and lemmas in all the papers considered.

In the following are displayed the 30 most common lemmas and words. Figures 1 and 2 are bar plots made to better visualise the results displayed above. From the plots, it can be seen that the most frequent lemma, and word, is cell, followed by protein, infection and viral.

The most indicative results are the ones obtained from the lemmas. In fact, looking at the results given using the words, we can see that there are repetitions of words used in different ways. For instance, the first two words cell and cells are better considered as a unique word rather than separated as it is the case without a proper lemmatisation. The 20 most common lemmas all describe pretty well the Covid-19 and, at least partially, the research on the matter. Nonetheless, the result shows that some information from the papers this way can be captured, even though it seems to be a bit superficial.

![Common words in COVID-19 scientific papers](image)

**Fig. 1.** Histogram of the frequency of the 30 most common words in the 326 papers considered.
These results were obtained without looking for specific parts of speech. Likely, more desirable results would be found if we were to look only for nouns and adjectives. With the use of regular expressions, we could look for these specifications and compare the results with what we obtained above.

To find the more helpful information, we proceed by analysing bigrams and trigrams. Since it is evident from this first glance at the papers that the lemmas are a better representation of the content of the documents than the raw words, we are going to focus on the results obtained using the lemmas in following sections while still providing the ones obtained with the words before lemmatisation.

### 3.1 Biagrams and Triagrams

In this part of the analysis, instead of considering the frequency of words by themselves, we consider the frequency of a sequence of words. In particular, we are going to focus on sequences of two and three words, respectively called bigrams and trigrams. To create these grams, we are going to use the two modules in the nltk library called bigrams and trigrams, precisely as one would expect them to be called.

The results above give us more information about the content of the papers. The differences between raw words and lemmas are an almost non-existent, sign that the n-grams are far more robust than the single words, also called unigrams. The results are also represent in the histograms of Figs. 3 and 4. These same results can also be seen using graphs. In this way, it is possible to see the connections between the different words used in the papers considered, at least for the ones considered in the plots, that are the 20 most common ones.
Fig. 3. Histogram of the frequency of the 30 most common bigrams before lemmatising in the 326 papers considered.

Fig. 4. Histogram of the frequency of the 30 most common bigrams after lemmatising in the 326 papers considered.

The graphs in Figs. 5 and 6 show as nodes the single words of the bigrams while the edges connect the name that belongs to the same bigram at least once.

The results of the trigrams are even more useful for the characterisation of the studies on the Covid-19 and on the virus itself. The results are as follows and show an even deeper understanding of the content of the papers:

What observed so far can be verified also in Figs. 7 and 8. In Figs. 9 and 10 are shown the graphs for the corresponding histograms in the same fashion as for the bigrams.
The graphs show which words fell into an n-gram, that is to say, which words happen in sequence among these most frequent n-grams. This shows how useful n-grams are in summarising, and eventually evaluating, the content of a collection of papers or texts in general.
Fig. 7. Histogram of the frequency of the 30 most common trigrams before lemmatising in the 326 papers considered.

Fig. 8. Histogram of the frequency of the 30 most common trigrams after lemmatising in the 326 papers considered.

4 IDF Analysis

The Inverse Document Frequency (IDF) measures how rare a word is in a collection of documents. This translates into a measure of how much information, or in other words, how rare or common a word is as a whole across all the papers.
considered. The IDF has been computed using the module TextCollection of the nltk library. The formula for the IDF is as follows:

$$idf(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|} \quad (1)$$

On the numerator there is $N$, that is the number of documents considered; while at the denominator, there is the number of records where the term $t$ appears. At the denominator 1 can be added since it can happen that a term is not in the collection of documents considered.
In the following are shown the top 30 words on the basis of the IDF:

Even more interesting would be to search for specific words in order to assess their informational importance as a whole. For this reason, we made a function to search for a particular word in the documents and compute its IDF, so that it would be possible to look for the importance of a certain word of interest, as we have done for the word skin above, which IDF is equal to 0.983.

5 Conclusion

In this work, we applied a TF-IDF transformation on text extracted from scientific papers dealing with COVID analysis. After having identified and trained some categories of interest, one of the objectives of the analysis was the implementation of a recommending system exploiting the TF-IDF values and the citations. Another focus of the analysis was the performance of the program. At the moment, it takes a long time to clean the whole paper, mainly because of the addition of the enchant library methods. The use of spark with Map-Reduce methods would have helped in improving the performances, but some difficulties came up in the implementation on the Google Colaboratory platform. Lastly, it would have been interesting to add the IDF values for the n-grams, but the performances of the program would have made it challenging to obtain the results in a reasonable time. Therefore, the main focus in the future will be dedicated to improving the performance of the program in order to add the other more sophisticated features.

References

1. Amato, F., Cozzolino, G., Maisto, A., Mazzeo, A., Moscato, V., Pelosi, S., Picariello, A., Romano, S., Sansone, C.: ABC: a knowledge based collaborative framework for e-health. In: 2015 IEEE 1st International Forum on Research and Technologies for Society and Industry Leveraging a Better Tomorrow (RTSI), pp. 258–263. IEEE (2015)
2. Alicante, A., Amato, F., Cozzolino, G., Gargiulo, F., Improda, N., Mazzeo, A.: A study on textual features for medical records classification. In: Innovation in Medicine and Healthcare 2014, vol. 207, p. 370 (2015)
3. Amato, F., Cozzolino, G., Giacalone, M., Moscato, F., Romeo, F., Xhafa, F.: A hybrid approach for document analysis in digital forensic domain. In: International Conference on Emerging Internetworking, Data & Web Technologies, pp. 170–179. Springer (2019)
4. Amato, A., Cozzolino, G.: Trust analysis for information concerning food-related risks. In: International Conference on Emerging Internetworking, Data & Web Technologies, pp. 344–354. Springer (2019)
5. Amato, A., Cozzolino, G., Moscato, V.: Big data analytics for traceability in food supply chain. In: Workshops of the International Conference on Advanced Information Networking and Applications, pp. 880–884. Springer (2019)
6. Castiglione, A., Cozzolino, G., Moscato, V., Sperlì, G.: Analysis of community in social networks based on game theory. In: 2019 IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Cloud and Big Data Computing, International Conference on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), pp. 619–626. IEEE (2019)

7. Amato, F., Cozzolino, G., Mazzeo, A., Romano, S.: Intelligent medical record management: a diagnosis support system. Int. J. High Perform. Comput. Networking 12(4), 391–399 (2018)

8. Canonico, R., Cozzolino, G., Ferraro, A., Moscato, V., Picariello, A., Sorrentino, F.R., Sperlì, G.: A smart chatbot for specialist domains. In: Workshops of the International Conference on Advanced Information Networking and Applications, pp. 1003–1010. Springer (2020)

9. Cozzolino, G.: Using semantic tools to represent data extracted from mobile devices. In: 2018 IEEE International Conference on Information Reuse and Integration (IRI), pp. 530–536. IEEE (2018)

10. Amato, A., Cozzolino, G., Giacalone, M.: Opinion mining in consumers food choice and quality perception. In: International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp. 310–317. Springer (2019)

11. Amato, A., Balzano, W., Cozzolino, G., Moscato, F.: Analysis of consumers perceptions of food safety risk in social networks. In: International Conference on Advanced Information Networking and Applications, pp. 1217–1227. Springer (2019)

12. Di Cicco Mattia Fonisto, F.: Covid-19 papers analysis