

JVLC

**Journal of
Visual Language and
Computing**

Volume 2021, Number 1

Copyright © 2021 by KSI Research Inc.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of the publisher.

DOI: 10.18293/JVLC2021-N1

Journal preparation, editing and printing are sponsored by KSI Research Inc.

**Journal of
Visual Language and Computing**

Editor-in-Chief

Shi-Kuo Chang, University of Pittsburgh, USA

Co-Editors-in-Chief

Gennaro Costagliola, University of Salerno, Italy

Paolo Nesi, University of Florence, Italy

Gem Stapleton, University of Brighton, UK

Franklyn Turbak, Wellesley College, USA

An Open Access Journal published by

KSI Research Inc.

156 Park Square Lane, Pittsburgh, PA 15238 USA

JVLC Editorial Board

Tim Arndt, Cleveland State University, USA

Paolo Bottoni, University of Rome, Italy

Francesco Colace, University of Salerno, Italy

Maria Francesca Costabile, University of Bari, Italy

Martin Erwig, Oregon State University, USA

Andrew Fish, University of Brighton, United Kingdom

Vittorio Fuccella, University of Salerno, Italy

Angela Guercio, Kent State University, USA

Erland Jungert, Swedish Defence Research Establishment, Sweden

Kamen Kanev, Shizuoka University, Japan

Robert Laurini, University of Lyon, France

Jennifer Leopold, Missouri University of Science & Technology, USA

Mark Minas, University of Munich, Germany

Brad A. Myers, Carnegie Mellon University, USA

Joseph J. Pfeiffer, Jr., New Mexico State University, USA

Yong Qin, Beijing JiaoTung University, China

Genny Tortora, University of Salerno, Italy

Kang Zhang, University of Texas at Dallas, USA

Journal Production Associate Editors

Jorge-Luis Pérez-Medina, Universidad de Las Américas, Ecuador

Yang Zou, Hohai University, China

Journal of Visual Language and Computing

Volume 2021, Number 1

August 2021

Table of Contents

Regular Papers

SENECA: A Pedagogical Tool supporting Remote Teaching and Learning	1
<i>Alessia Auriemma Citarella, Luigi Di Biasi, Stefano Piotto, Michele Risi and Genoveffa Tortora</i>	
YouCare: A Cross-Platform Telehealth App for COVID-19.	11
<i>Gennaro Costagliola, Mattia De Rosa, Vittorio Fuccella and Francesco Vitale</i>	

Research Notes

PADD: Dynamic Distance-Graph based on Similarity Measures for GO Terms Visualization of Alzheimer and Parkinson Diseases	19
<i>Alessia Auriemma Citarella, Fabiola De Marco, Luigi Di Biasi, Michele Risi and Genoveffa Tortora</i>	
Use of Natural language inference in optimizing reviews and providing insights to end consumers	29
<i>Tandon Chahat, Jayesh Bongale Pratiksha, T M Arpita, R R Sanjana, Palivela Hemant and C R Nirmala</i>	
Long-Term Predictions of Bike-Sharing Stations' Bikes Availability..	35
<i>Enrico Collini, Paolo Nesi and Gianni Pantaleo</i>	

Journal of Visual Language and Computing

journal homepage: www.ksiresearch.org/jvlc

SENECA: A Pedagogical Tool Supporting Remote Teaching and Learning

Alessia Auriemma Citarella^{a,*}, Luigi Di Biasi^a, Stefano Piotto^b, Michele Risi^a and Genoveffa Tortora^a

^aDepartment of Computer Science, University of Salerno, 84084 Fisciano (SA), Italy

^bDepartment of Pharmacy, University of Salerno, 84084 Fisciano (SA), Italy

ARTICLE INFO

Article History:

Submitted 7.31.2021

Revised 8.10.2021

Accepted 8.17.2021

Keywords:

Natural Language Processing

Distance Education

Learning

Attention

Content

ABSTRACT

In this paper, we suggest SENECA, a tool that attempts to assist students who follow remote classes in maintaining/capturing attention, allowing them to focus on context-driven learning. Distance education has a number of disadvantages, including a lack of physical interaction between students and teachers, emotional and motivational isolation as a result of this strategy, and a reduction in active engagement. All of these things have an impact on student learning abilities. The largest distractions at home are considered among these disadvantages of distant education, particularly for subjects with low awareness. These distractions cause a movement of the student's attention from the current lesson to disturbing events. For this reason, there is a need to experiment with new solutions also linked to *Information Technology* (IT) to improve the focused learning during distance education. Our tool's technical idea is to create a real-time summary of the topic treated by the teacher. The system captures the text every five minutes, generates outlines, and browses them to eliminate repetitive portions after each survey. We looked at two different sorts of filters, semantic and summary, to see if the first could distinguish between topics and the second could evaluate the topic's highlights. Natural Language Processing algorithms are used to extract categories and keywords from the general generated summary. The latter will emphasize the most important points of the speech, while the keywords will be utilized to extract the candidate literature about the discussed topics.

© 2021 KSI Research


1. Introduction

Into the current pandemic situation generated by the SARS-CoV-2 coronavirus, the educational environment around the world is faced with several problems and challenges in order to continue teaching in schools and universities.

Recent work has addressed the issue of distance education by administering questionnaires to both teachers and pupils. The most variable answers to the questions were also obtained on the degree of students' participation in distance lessons, emphasizing a wide range of behaviors. Furthermore, perception of difficulty during remote lessons was

found to be linked to many factors: access to technology, motivation and support with a greater presence of negative experiences [28]. The increased educational needs of online teaching, as well as the shifting learning styles of the students, impede comprehensive and effective knowledge transmission. Many dysfunctional behaviors can be developed as a result of the detachment of presence predicted by distance teaching. Among them we include loss of interest, attention, and motivation due to psycho-physical causes and non-adaptation to an abnormal setting. Distance learning has a disadvantage in terms of distractions as compared to traditional education environment. It's a solitary experience with no direct communication, which makes participation much more active [15]. It would be beneficial to overcome the difficulties of keeping the attention of a student, whether or not a circumstance necessitates the use of distant learning resources. Overcoming these obstacles would aid in refin-

*Corresponding author

 aauriemmacitarella@unisa.it (A. Auriemma Citarella);

ldibiasi@unisa.it (L. Di Biasi); piotto@unisa.it (S. Piotto);

mrisi@unisa.it (M. Risi); tortora@unisa.it (G. Tortora)

ORCID(s): 0000-0002-6525-0217 (A. Auriemma Citarella);

0000-0002-9583-6681 (L. Di Biasi); 0000-0002-3102-1918 (S. Piotto);

0000-0003-1114-3480 (M. Risi); 0000-0003-4765-8371 (G. Tortora)

DOI reference number: 10-18293/JVLC2021-N1-006



Figure 1: A standard bidirectional media streaming.

ing each student’s strategic learning styles and ensuring a meta-cognitive self-assessment approach to one’s limits and abilities, all of which would be aided by technology.

A student’s ability to become aware of his or her ability to “*learning how to learn*” is another recent meta-cognitive skill. This ability means recognizing and then consciously applying appropriate behaviors and strategies useful for a more effective learning process [31].

This paper proposes a Distributed Multimedia System for support learning, designed to face the loss of attention during distance education. The purpose of the system is to be able to reawaken or maintain attention to the context (topic) that is being experienced during the activity in progress (in *real time*) to reduce the negative effect of distractions. The system also aims to provide the possibility of an in-depth analysis at the end of the lesson through auto-generated hyperlinks to lesson-related content. The architecture proposed rely on *Speech-to-text*, *Natural Language Processing (NLP)*, *Text Summarization* [19] and *Semantic Analysis technology*.

This paper is an extension of the work *SENECA: An Attention Support Tool for Context-related Content Learning* [3]. For this contribution, we enriched the Section 7 with a new experimentation for the SUMMARY filter of the SENECA pipeline. Specifically, we conducted a validation with the support of three independent and expert reviewers who assessed the consistency and degree of consistency of the SUMMARY filter’s output on several lessons. Three criteria were evaluated for coherence with the topic of the lessons: the summary, the keywords extracted and the insights.

This document is organized as follows. In Section 2, we described the most important related works about the technological systems that help in learning. In Section 3 we introduced the used methodologies. In Section 4 we have highlighted the working hypotheses on which we based our work. In Section 5 we presented the system architecture and in Section 6 we have detailed the performed experiments and the related results (Section 7). Finally, in Section 8 we will discuss research directions and future development of our work.

2. Related works

In this section, we present some related works about the use of semantic and NLP analysis technologies. Specifically,

we will discuss different application contexts of these techniques. We also presented an overview of how the working memory works and the impact of technology on it.

One of the most important aspects of cognitive function is the “*ability to keep*” relevant information in mind. *Working memory* is a system dedicated to the maintenance and temporary processing of information during cognitive processes. One of the components is represented by the *central executive* which carries out the coordination of subordinate systems, coordination of execution of tasks and recovery of strategies and attentional functions of both selection and inhibition [4]. The central executive controls the *phonological loop* which contains verbal and auditory information, the *visuo-spatial sketchpad* engaged in spatial representation and the *episodic buffer* which has a limited ability to link information from different sources with spatial and temporal parameters.

Specifically, each attentive act is divided into three phases: the orientation and perception towards the different stimuli; the processing phase that presents the function of selectivity and sustained attention overtime on a task or activity, the shift to move the focus quickly and the ability to pay attention to use the right cognitive resources in different situations; the specific response concerning the input stimuli [5].

Different studies have focused on the impact of technologies on cognitive functions in the present digitized era, both from the perspective of the benefits and disadvantages [40]. *Lodge and Harrison* [27] stressed as attention is subject to complex dynamics that impact learning, especially in educational contexts. The most important part of a sentence, oral or written, is the focus. Recent articles have demonstrated the importance of *marking elements* as a *guide* for better information exchange [26] between speakers and listeners. In particular, these studies argue that focus marking captures the listener’s attention to what the speaker considers the most relevant part of the message. At the same time, this strategy aids in maintaining focus on the highlighted element, allowing for its representation [34].

Most of a student’s effort is to transfer information from working memory to long-term memory to acquire and memorize key concepts. Two strategies can be used: dual coding and chunking [11]. In cognitive psychology, a *chunk* is nothing more than a unit of information, and chunking is the operating mode in which this unit of data is recovered.

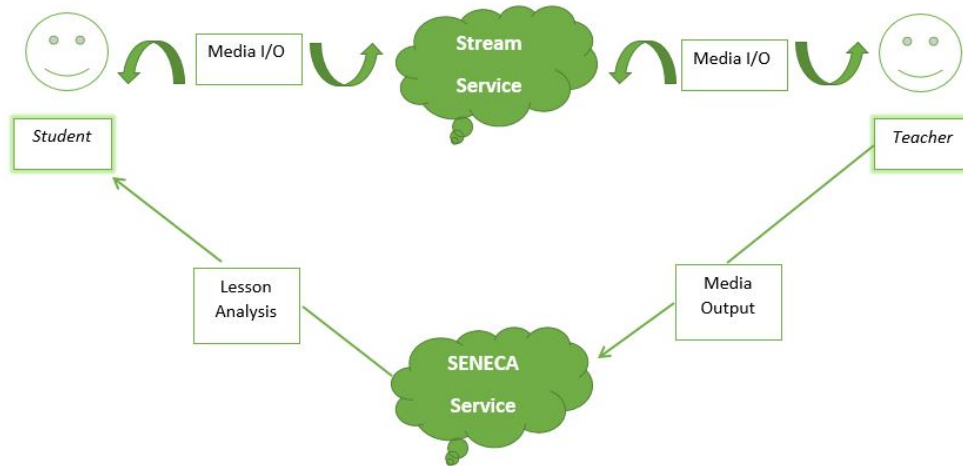


Figure 2: The additive layer for multimedia analysis.

When faced with new knowledge, the individual can grasp the relative chunk of information and bring it back to light later when recalling a similar situation or concept. Then, the initial piece can be expanded into more complex pieces following the management control and understanding of the flows of one’s knowledge [37].

The standard structure of a *Distance Educational System* can be generalized as reported in Fig. 1. This kind of system relies on the classic bidirectional multimedia connection, like the common video-chat system based on SIP/VOIP system, such as Microsoft Teams [25]. However, a Distance Educational System supports multiple bidirectional connections between students and teachers and allows channel moderation.

Nowadays, the cloud’s audio/video stream transfer services are implemented by the major world providers (*Amazon, Microsoft, Zoom*). A teacher can teach remotely by transmitting an audio/video stream from their home to one of these providers. Then these last provide a broadcast service to the students.

NLP techniques have been widely used in intelligent tutoring systems that helped acquire content knowledge [9]. For example, in Guzmán-García *et al.* [22], the analysis of the speaking of surgeons into the operating room through NLP techniques is proposed to obtain a deeper vision of intraoperative decision-making processes. The goal of this study was to create a technique for identifying and analyzing the various surgical phases, as well as a workflow that was equivalent to the procedure’s framework, in order to improve surgical learning in The Educational Operating Room.

Recent studies have emphasized the importance of identifying the main contents in order to better understand a topic, particularly in students with cognitive disabilities and attention or memory problems. The ability to take advantage of text summarization techniques by explaining the main idea allows students to interface with the limits of their working memory and have a tool to overcome their difficulties [36].

Today, many educational and academic institutions benefit from the *Learning Management Systems (LMS)* to support and improve teaching processes [16][20]. Most LMS are software application systems that allow teachers to manage and deliver educational courses [2]. One of the requirements for the success of distance education is traceable in the self-management of learning which is the starting point for self-discipline in autonomous learning [38].

In 2019, Cobos *et al.* [13] have developed EdX-CAS, a content analyzer system for edX MOOCs, using NLP techniques for the Spanish language. The program accepts video transcripts from courses as input and allows users to interact with them specifically. It allows us to extract the text’s main terms, the vector representation for each of the terms in the text, the linguistic diversity to understand how many different words are used, indications on the subjective opinion on the text and the representation with word clouds. The EdX-CAS tool is oriented to Sentiment Analysis Opinion Mining for Detecting Subjectivity and Polarity Detection in Online Courses related to Madrid’s Universidad Autónoma.

On the topic of the educational distance imposed by the Covid-19, to support students in self-training, a chatbot was proposed using NLP techniques [17]. The proposed solution involves sending a message to Moodle [1] by the student. An accompanying plugin tries to decipher the text and provides feedback. Based on the degree of assessment achieved by the student, the chatbot makes suggestions for the chapters where the evaluation is insufficient. The system presupposes the memorization of the student evaluation outcomes, accessible to teachers. In this context, the chatbox, acts as a tutor and allows us to fill the gaps of the students.

3. Methods

We propose a new tool called SENECA (Support IEarning coNtEnt Context Attention), which involves using a new layer dedicated exclusively to analyzing the audio/video streams

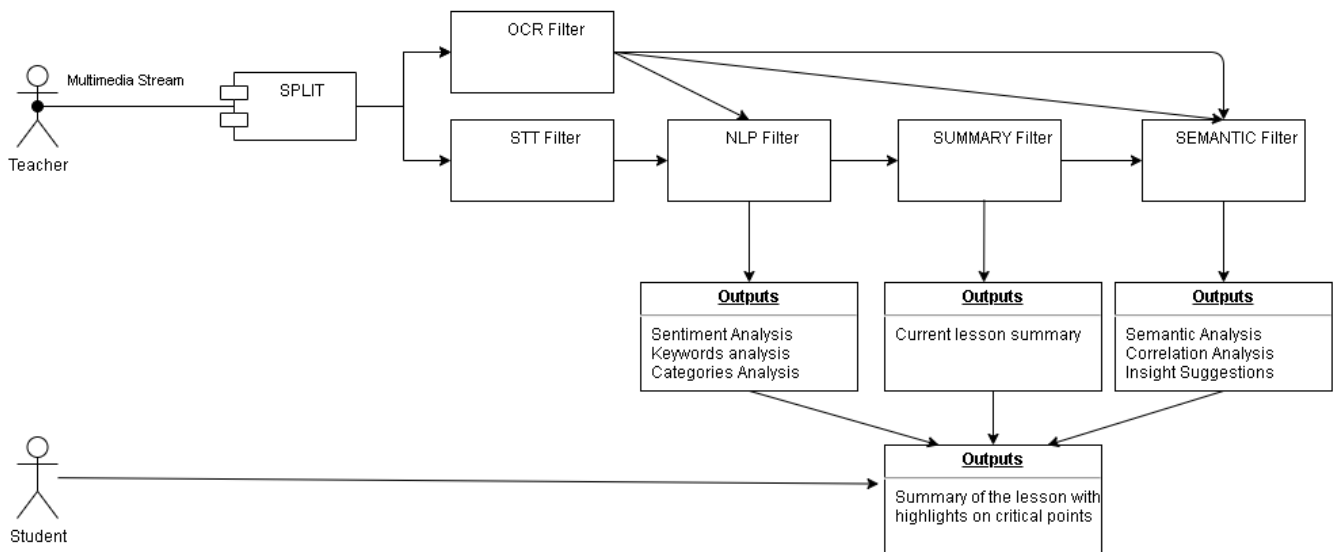


Figure 3: A basic SENECA module.

generated by the “teachers”. By integrating this new layer, we hope to preserve or quickly recover student focus. The proposed architecture is to be considered feasible for real-time distance lessons and not to the MOOC [6] or the on-demand recorded lessons. We did not consider the capabilities related to file upload, file sharing and homework as an added values.

The new layer is identified in the Fig. 2 as Seneca Service. In this proposal, we refer only to real-time audio/video streams, i.e., not recorded lessons held at a distance. The presence of the real-time component allows us to immerse ourselves in a learning environment that is subject to disruptions that distract the individual student.

SENECA’s major goal is to help students avoid losing concentration by providing multiple of information that can help them stay focused on the argument or return their attention to the context, if they become distracted. Another key purpose is to encourage the use of a variety of text analysis approaches to improve the learning quality of the topic under study and integrate it. For this purpose, the system supplies additional information that can be used to recover information at the end of the session.

4. Working hypothesis

In this section, we will highlight some working assumptions that will help us gain a better understanding of the SENECA tool’s initial concept:

- a) The real-time audio/video stream (from here called STREAM) generated by a remote lesson can be split into two unique sub-streams: VIDEO flow, containing the video frames and AUDIO flow, containing audio buffer.
- b) The VIDEO flow will contain information from slides or, in any case, projected material to provide a conceptual map to students.

- c) The AUDIO flow will contain the lesson audio, and it is expected to add information on both the context under study and in-depth study (as well as student questions or others).

In this context, we assume that the VIDEO stream contains *information already summarized on the subject*. In our experiment, we considered the data extracted from VIDEO as already cleaned. On the other hand, the presence of heterogeneous data in AUDIO streams will require a more accurate analysis of the content.

The extracted data from VIDEO and AUDIO is referred following as WORD STREAMS.

5. System architecture

We designed a prototype architecture based on a pipeline approach, like Microsoft DirectShow ¹ or ffmpeg ².

In SENECA each computational block is called *Filter*.

An overview of a complete SENECA architecture is shown in Fig. 3. For this proposal, we implemented only the following filters: SPLIT, OCR, STT, SUMMARY and SEMANTICS.

The filters are defined as following:

$$SPLIT(STREAM) \rightarrow \{AUDIO, VIDEO\}$$

$$OCR(VIDEO) \rightarrow \{WS\}$$

$$STT(AUDIO) \rightarrow \{WS\}$$

$$SUMMARY(WS, GLS) \rightarrow \{NEW GLS\}$$

$$SEMANTIC(GLS) \rightarrow SUGGESTIONS$$

The SPLIT filter takes as input the STREAM and splits it into two separate flows, called AUDIO and VIDEO.

¹<https://docs.microsoft.com/en-us/windows/win32/directshow/directshow>

²<https://ffmpeg.org/developer.html>

Table 1
Partial subset of lessons MEF.

Topics	MEFs
<i>Cancer</i>	Smoking, Colon Cancer, Surgery, Risk Factor
<i>Diabetes</i>	Beta Cell, Interleukin, Inflammatory, Physic
<i>Evolution</i>	Selection, Heritability, Billion Years, Coevolution
<i>Terrorism</i>	Terror, Poverly, Success, Politican
<i>Chemistry</i>	Compound, Energy, Element, Electron

The OCR technique allows the detection and extraction of text from images [30]. The SENECA OCR Filter takes as input a single frame video at a time. We used the `videocr` python module (v. 0.1.6)³ for our experiment purpose. That module lies on Tesseract OCR 4.1.1.⁴ This filter analyzed each video frame from the pipeline and stored the detected text (handwritten and block letters) into a word stream (WS). Each word stream was enqueued into the next pipeline filter.

In SENECA, the STT filter performs a speech-to-text routine. Speech-to-text is a technique that allows the detection and the extraction of phrases from an audio flow WS [12]. Probably, the most commonly known example is Amazon Alexa or Google Assistant. Into our prototype, we used the Google Cloud Speech API⁵. For each audio frame extracted by the SPLIT, the STT filter generated a word-stream (WS) that was enqueued to the NPL filter.

One of the project goals is to provide a way to regain the attention on the topic focus after a distraction. In SENECA, one of the tips is to allow users to summarize the lesson in real-time. As shown in Fig. 3, the media flow comes into the SPLIT filter that separates audio from video. For each video frame extracted, the OCR retrieves the identified sentences and STT does the same for the audio frame. These word streams, in particular, entered the following filter, the SUMMARY filter, which is a delegate for creating partial summaries from the word streams that entered the filter. Into our prototype, the SUMMARY filter computes a summary for the WSs using MEAD [19]. These multiple summaries are merged every 5 minutes into the Global Lesson Summary (GLS) that is processed again by MEAD. We have chosen the five minutes interval using the mean lesson length. Consequently, SENECA builds and refreshes a GLS by using SUMMARY filter output for each real-time lesson. Into our prototype, GLS is composed of phrases generated by applying MEDA text summarization algorithm on WSs.

The SEMANTIC filter extracts the MEF from the GLS every time a new GLS is deployed from the SUMMARY filter. We identify with the term MEF or *Most Expressed Features* of a string S, the dictionary D(S) of all possible *k*-mers extracted from S, using substrings length between m and M. The dictionary is ordered in a decreasing way, compared to the number of occurrences of each *k*-mers. The MEFs represent the object's functional parts, such as words or portions

of sentences of a text repeated several times. SEMANTIC is designed to make suggestions by probing one or more scientific databases using the MEF. In particular, it performs a combined NO-SQL alignment-free search into the pre-processed PubMed database (see next paragraph). For each GLS, SEMANTIC can extract the candidate literature papers indexed by MEF. It uses two metrics to compute (see experiment one) the suggested papers based on the semantic distance between GLS and candidate paper set.

For this prototype, we used the PubMed database [10], due to the higher prevalence of selected biomedical topics. We executed the SEMANTIC filter on the entire PubMed dataset, and we have extracted the MEFs, using substrings length interval between $m=3$ and $M=15$.

Due to the PubMed dataset size, we used Amazon EC2 and Amazon RDS services [32] to distribute MEFs extraction and storage.

6. Experiment Execution

We simulated real-time lessons by using public videos from Coursera⁶. We selected five free courses, recovering from each it, the text subscription using STT. The main topics of the lessons are:

- Cancer;
- Diabetes;
- Evolution;
- Terrorism;
- Chemistry.

We implemented two experiments. Our goal was to study the SUMMARY and SEMANTIC filter performances.

6.1. First Experiment

Into the first experiment, we sent all the entire lessons (5 merged videos per lesson) into the SENECA pipeline, sequentially, to compute separate GLS output for each entire lesson. Also, we sent each video (one at a time, not merged) into the pipeline to generate the GLS for each video.

We wanted to study if the SEMANTIC filter was able to discriminate between lesson topics. We applied the SEMANTIC filter on each GLS to extract the MEFs for each of them.

³<https://pypi.org/project/videocr/>

⁴<https://github.com/tesseract-ocr/tesseract>

⁵<https://cloud.google.com/speech-to-text/>

⁶<https://www.coursera.org/>

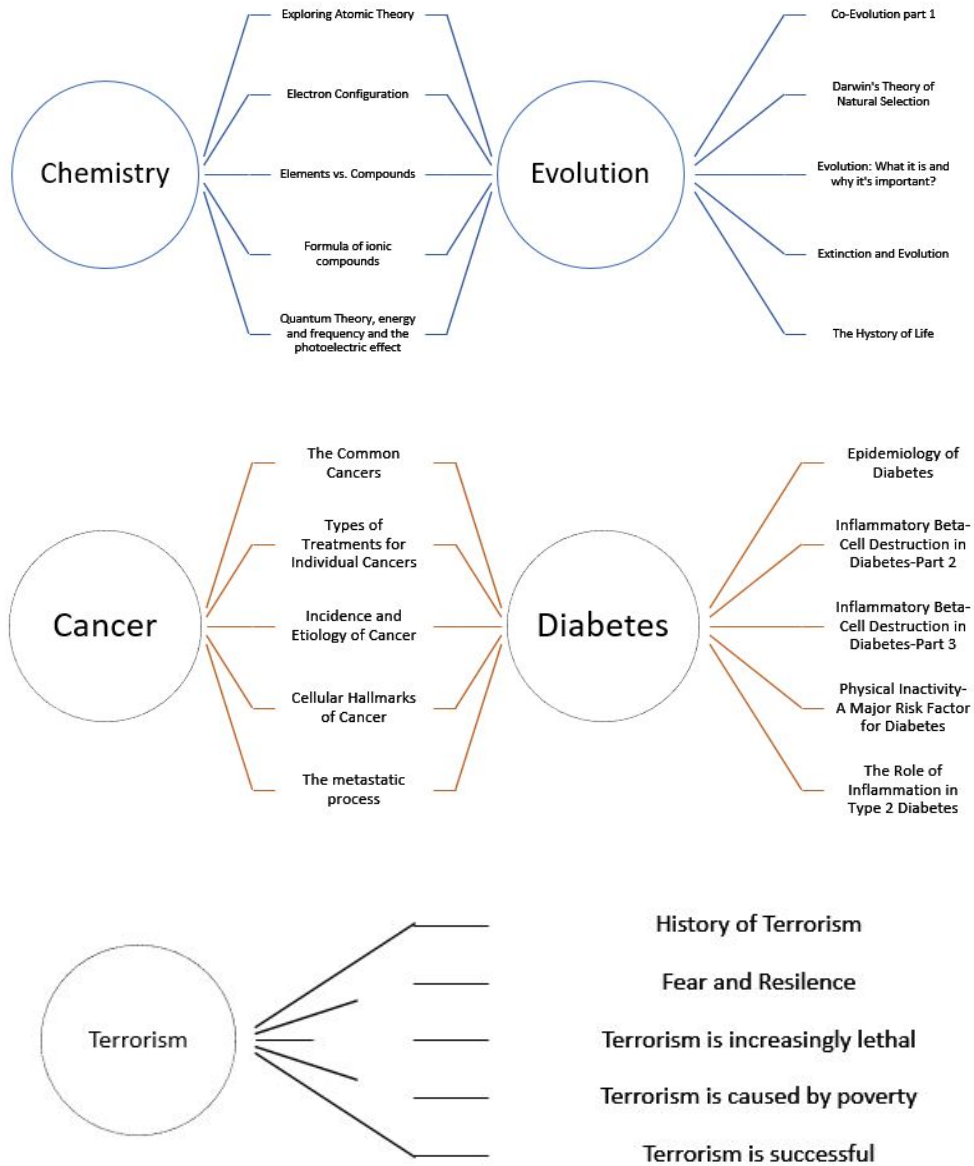


Figure 4: Lesson Topics clustering using Jaccard distance.

Using the MEFs, we were able to use two different distance metrics. The distances were calculated using the *Jaccard Index* [33] (see Eq. 1) and *Szymkiewicz–Simpson coefficient (SSC)* [41] (see Eq. 2).

Both metrics allow us to compute the similarity between pairs of MEF dictionary. SSC is often identified as the “*overlap coefficient*.”

The Jaccard index is defined as:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

where A and B are two different datasets, whilst the $|\cdot|$ operator computes the size of a set. In particular, the Jaccard

index is represented as the size of the intersection divided by the size of the union of the datasets.

Given the two dictionaries A and B , the overlap coefficient is a measure that returns the overlap between them and it is defined as the intersection divided by the smaller of the size of the two sets, as shown in Eq. 2.

$$SSC(A, B) = \frac{|A \cap B|}{\min(|A|, |B|)} \quad (2)$$

We used the dictionaries and the distance matrices to extract the most expressed keywords and cluster the GLS.

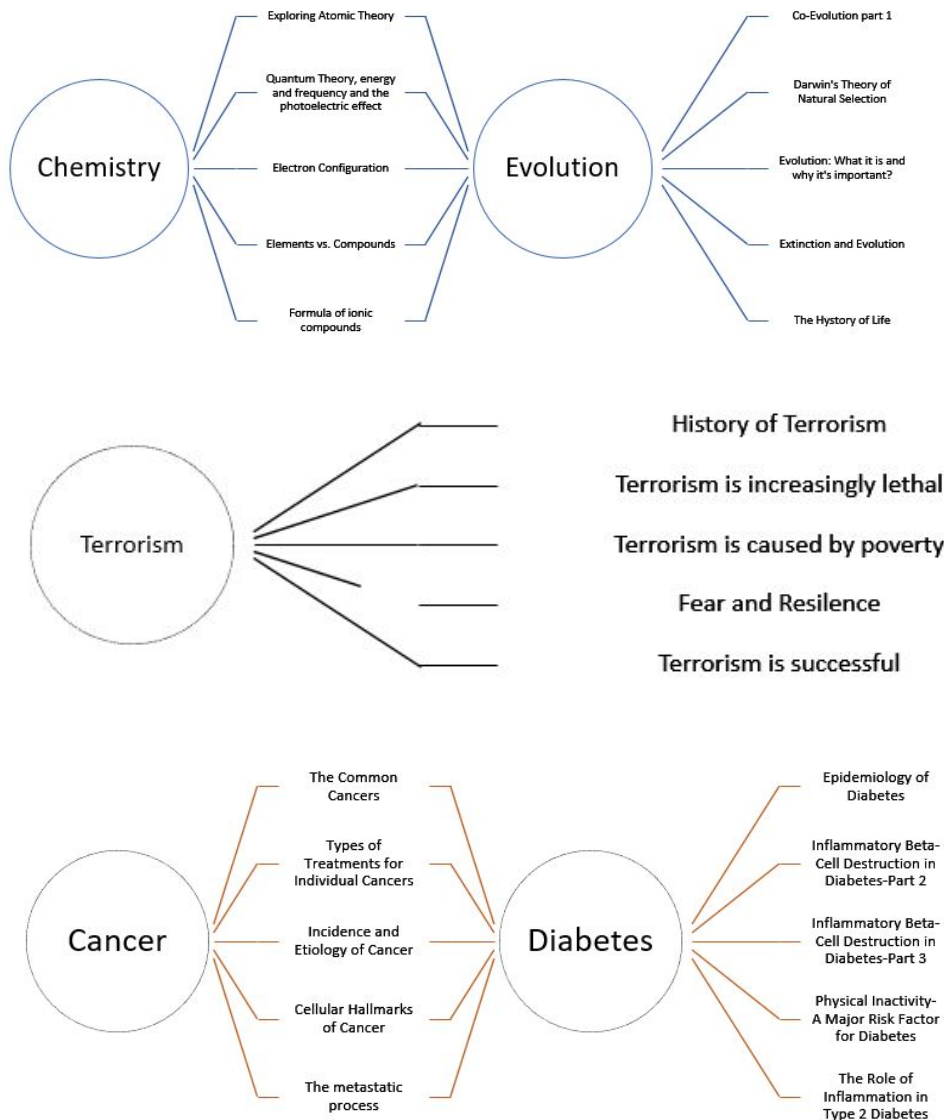


Figure 5: Lesson topics clustering using SSC.

6.2. Second Experiment

Into the second experiment, we sent each lesson, each in turn, to the pipeline to generate its GLS. For each GLS, we applied the SEMANTIC filter in order to extract the related MEFs to use them as probes for the subsequent insights.

We applied Jaccard and SSC metrics between each GLS and the PubMed-MEF dataset.

For each GLS, we computed ten distance matrices and selected the first ten most similar results (see Table 2). Due to the high memory requirements of these tasks, we were forced to employ Apache Spark [39] to distribute this job across multiple slaves.

7. Experimental Results

The Table 1 shows the first four MEFs for each lesson. The SEMANTIC filter was able to detect keywords related to lesson context. Due to the GLS and MEF definitions, the MEF dictionaries contain up to 140K kmers for each GLS. We reported the most expressed that refer to complete word.

The results of topics clustering are available in Fig. 4 with the Jaccard index and Fig. 5 with the overlap coefficient.

We used the same color to identify topics that belong to the same branch of the tree and different colors to identify subgroups of each topic. With only minor discrepancies in lesson aggregation levels, the filter SEMANTIC successfully separated the five treated themes. Specifically, it is interesting to note how the system is able to cluster together the *Cancer* and *Diabetes* and *Chemistry* with *Evolution* topics.

In this scenario, there appears to be a common thread connecting the two diseases and chemical topics with evolution, with the basic premise that the latter is the branch of natural sciences at the foundation of life and recognized material transformations.

In the second experiment, we used the extracted MEFs as if they represented ‘tags’ to recover suggested papers. For convenience, we have used only the MEFs of the lessons on the topics *Cancer* to show the results.

SEMANTIC identified 274 correlated documents recovered by PubMed-MEF dataset. For example, in Table 2, we showed the first ten recovered papers with a score value greater than 0.80. In Fig. 6, we represented the position of suggested papers graphically compared to the target query, keeping in mind that the score of the distance from the target query is representative of the similarity between the set of MEFs of the *Cancer* topic and the individual retrieved papers.

We chose one topic, *Pharmacology*, consisting of five lessons from COURSERA. We invited three independent and expert reviewers, which submitted the output of SUMMARY filter in order to validate our SENECA pipeline. For each lecture, three parameters are assessed: summary, keyword, and insight. The initial assessment is based on the consistency of these three factors, with a binary Yes/No (indicated as Y/N) response. The second evaluation of the experts is assigning a consistency rating between 0 and 5 to each of the three aspects under consideration. The results are shown in Table 3. Expert evaluators performed a manual evaluation, giving a direct score on the coherence of the summary. Other factors, such as the quality of the summary or language comprehension were not considered. In Table 4, we reported the average scores reached for each lessons based on the votes assigned by the reviewers. For the five lessons, we can see how, on average for the threshold score, the SUMMARY filter fluctuates between a minimum of 2 and a maximum of 3. The value for keywords is around 2-4 and the value for the insights is between 0.3 and 3.7. The first two lessons are ones in which the reviewers had differing perspectives on all three aspects assessed, while the subsequent lessons have similar values assigned. On average, the SUMMARY filter and keywords extraction perform better than the insights recovery phase. The discordance of score on the first two lessons suggests that the results are in any case influenced by subjective elements linked to the identity of the reviewer. The main emerged considerations from this first examination are that the keywords may be improved, as they are currently too generic in some circumstances.

8. Conclusions

The change to the basis of remote teaching is the transition from traditional education to smart education. The teacher is responsible for managing class to be student-centered, which involves greater responsibility and awareness of their limits and potential in self-learning behind a display. It is not always possible because disturbing environmental factors may cause the student’s attention to be diverted. This is

Table 2

Partial subset of suggested papers and their distance score.

Paper ID	Author	Value
P1	Belsky <i>et al.</i> [8]	0.91
P2	Huang <i>et al.</i> [24]	0.81
P3	Wu [43]	0.93
P4	Gaitanidis <i>et al.</i> [18]	0.83
P5	Bauer <i>et al.</i> [7]	0.88
P6	Wang <i>et al.</i> [42]	0.95
P7	Mays <i>et al.</i> [29]	0.94
P8	Schuck <i>et al.</i> [35]	0.93
P9	Guo <i>et al.</i> [21]	0.92
P10	Corsi <i>et al.</i> [14]	0.90

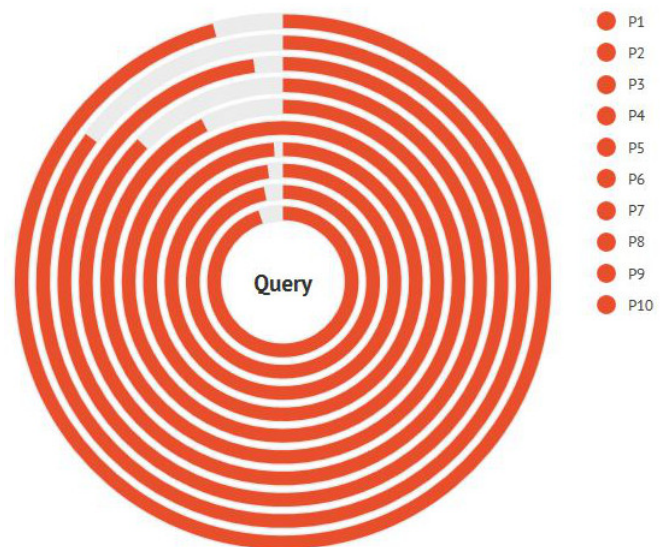


Figure 6: Position of suggested papers compared to the target query.

one of the disadvantages of remote education, which intervene in maintaining the focus. In an era in which new teaching tools are proposed, managing personal learning is also changed. In this work, we have proposed a tool that aims to maintain attention of students on topics covered in the lessons held by teachers. The technology operates in real-time and allows us to get additional knowledge by allowing us to conduct in-depth research using hyperlinks to related topics. Preliminary testing indicates that the framework can provide insight and assist in refocusing attention on the lesson. We used independent experts to perform a first evaluation phase of our pipeline. This step showed how SENECA can be further improved in the appropriateness of insight research. Future enhancements may possibly include different applications of summarization technologies. Manual evaluation necessitates the utilization of external resources, which should not be underestimated. As a result of this issue, we may meet this demand by developing future automatic assessment approaches that compare the quality of the summary to the Gold Standard Reference, which was prepared by an expert. In this case, a score could be assigned based on

Table 3
Evaluation of SUMMARY filter of SENECA pipeline

Lessons	Coherence			Coherence Threshold			
	Summary	Keywords	Insight	Summary	Keywords	Insight	
Pharmacology 1	Reviewer 1	Y	Y	Y	3	3	2
	Reviewer 2	Y	Y	N	3	3	0
	Reviewer 3	N	Y	N	0	2	0
Pharmacology 2	Reviewer 1	Y	Y	Y	3	3	1
	Reviewer 2	Y	Y	N	3	3	0
	Reviewer 3	N	N	N	0	0	0
Pharmacology 3	Reviewer 1	Y	Y	Y	3	2	4
	Reviewer 2	Y	Y	Y	3	2	4
	Reviewer 3	Y	Y	Y	3	2	4
Pharmacology 4	Reviewer 1	Y	Y	Y	3	4	3
	Reviewer 2	Y	Y	Y	3	4	3
	Reviewer 3	Y	Y	Y	2	4	2
Pharmacology 5	Reviewer 1	Y	Y	Y	3	3	3
	Reviewer 2	Y	Y	Y	3	3	3
	Reviewer 3	Y	Y	Y	3	3	5

Table 4
Average coherence threshold scores for summary, keywords and insights.

Lesson	Summary	Keywords	Insight
Pharmacology 1	2	2.7	0.7
Pharmacology 2	2	2	0.3
Pharmacology 3	3	2	4
Pharmacology 4	2.7	4	2.7
Pharmacology 5	3	3	3.7

the extent to which the subject is covered. *Human generated* models can also be considered. Important elements of the semantic level, referred to as *Summary Content Units(SCU)*, are the annotations that expresses the content unit’s semantic meaning. In this pyramidal model, each SCU is assigned to a weight based on the number of models that contain it. A perfect summary is made up of a subset of the entire SCUs made up of those with the highest index [23].

To determine the framework’s impact on student attentiveness in remote lessons, real-world testing should be done. Among the future directions’ objectives, we expect to explore techniques that improve text summarization and the extension of architecture for multilingual texts. An additional module for the real-time creation of concept maps may be provided. As goals of future direction, we expect to investigate approaches to improve text summarization and the extension of architecture for multilingual texts. In addition, a separate module for creating idea maps in real time will be provided. In this way, students will be able to structure and organize material more effectively. This would allow for better assimilation of the information provided in the lessons.

References

[1] Al-Ajlan, A., Zedan, H., 2008. Why Moodle, in: Procs. of the 12th IEEE International Workshop on Future Trends of Distributed Computing Systems, IEEE. pp. 58–64.
 [2] Alias, N.A., Zainuddin, A.M., 2005. Innovation for better teaching

and learning: Adopting the learning management system. Malaysian Online Journal of Instructional Technology 2, 27–40.
 [3] Auriemma Citarella, A., De Marco, F., Di Biasi, L., Risi, M., Tortora, G., 2021. Seneca: An attention support tool for context-related content learning, in: 27th International Distributed Multimedia Systems Conference on Visualization and Visual Languages, DMSVIVA 2021, Knowledge Systems Institute Graduate School, KSI Research Inc.. pp. 36–45.
 [4] Baddeley, A., 1996. Exploring the central executive. The Quarterly Journal of Experimental Psychology - Section A 49, 5–28.
 [5] Baddeley, A., 2000. The episodic buffer: a new component of working memory? Trends in cognitive sciences 4, 417–423.
 [6] Baggaley, J., 2013. Mooc rampant. Distance Education 34, 368–378.
 [7] Bauer, M., Morales-Orcajo, E., Klemm, L., Seydewitz, R., Fiebach, V., Siebert, T., Böhl, M., 2020. Biomechanical and microstructural characterisation of the porcine stomach wall: Location-and layer-dependent investigations. Acta Biomaterialia 102, 83–99.
 [8] Belsky, D.W., Moffitt, T.E., Baker, T.B., Biddle, A.K., Evans, J.P., Harrington, H., Houts, R., Meier, M., Sugden, K., Williams, B., et al., 2013. Polygenic risk and the developmental progression to heavy, persistent smoking and nicotine dependence: Evidence from a 4-decade longitudinal study. JAMA Psychiatry 70, 534–542.
 [9] Burstein, J., 2009. Opportunities for natural language processing research in education, in: Procs. of the International Conference on Intelligent Text Processing and Computational Linguistics, Springer. pp. 6–27.
 [10] Canese, K., Weis, S., 2013. Pubmed: The bibliographic database, in: The NCBI Handbook [Internet]. 2nd edition. National Center for Biotechnology Information (US).
 [11] Chew, S.L., Cerbin, W.J., 2021. The cognitive challenges of effective teaching. The Journal of Economic Education 52, 17–40.
 [12] Chiu, C.C., Sainath, T.N., Wu, Y., Prabhavalkar, R., Nguyen, P., Chen, Z., Kannan, A., Weiss, R.J., Rao, K., Gonina, E., Jaitly, N., Li, B., Chorowski, J., Bacchiani, M., 2018. State-of-the-art speech recognition with sequence-to-sequence models, in: Procs. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4774–4778.
 [13] Cobos, R., Jurado, F., Blázquez-Herranz, A., 2019. A content analysis system that supports sentiment analysis for subjectivity and polarity detection in online courses. IEEE Revista Iberoamericana de Tecnologías Del Aprendizaje 14, 177–187.
 [14] Corsi, D.J., Chow, C.K., Lear, S.A., Subramanian, S., Teo, K.K., Boyle, M.H., 2012. Smoking in context: A multilevel analysis of 49,088 communities in Canada. American Journal of Preventive Medicine 43, 601–610.

- [15] Cowan, J., 1995. The advantages and disadvantages of distance education. *Distance Education for Language Teachers: A UK Perspective*, 14–20.
- [16] Francese, R., Gravino, C., Risi, M., Scanniello, G., Tortora, G., 2015. Using project-based-learning in a mobile application development course-an experience report. *Journal of Visual Languages & Computing* 31, 196–205.
- [17] Gaglo, K., Degboe, B.M., Kossingou, G.M., Ouya, S., 2021. Proposal of conversational chatbots for educational remediation in the context of Covid-19, in: *Procs. of the 23rd International Conference on Advanced Communication Technology (ICACT)*, IEEE. pp. 354–358.
- [18] Gaitanidis, A., Machairas, N., Alevizakos, M., Tsalikidis, C., Tsaroucha, A., Pitiakoudis, M., 2019. Predictive nomograms for synchronous liver and lung metastasis in colon cancer. *Journal of Gastrointestinal Cancer*, 1–7.
- [19] Gambhir, M., Gupta, V., 2017. Recent automatic text summarization techniques: A survey. *Artificial Intelligence Review* 47, 1–66.
- [20] Gravino, C., Risi, M., Scanniello, G., Tortora, G., Francese, R., 2016. Supporting mobile development project-based learning by software project and product measures, in: *Procs. of the 22nd International Conference on Distributed Multimedia Systems (DMS)*, pp. 41–47.
- [21] Guo, Q., Unger, J.B., Palmer, P.H., Chou, C.P., Johnson, C.A., 2013. The role of cognitive attributions for smoking in subsequent smoking progression and regression among adolescents in China. *Addictive Behaviors* 38, 1493–1498.
- [22] Guzmán-García, C., Gómez-Tome, M., Sánchez-González, P., Oropesa, I., Gómez, E.J., 2021. Speech-based surgical phase recognition for non-intrusive surgical skills' assessment in educational contexts. *Sensors* 21, 1330.
- [23] Hennig, L., De Luca, E.W., Albayrak, S., 2010. Learning summary content units with topic modeling, in: *Coling 2010: Posters*, pp. 391–399.
- [24] Huang, X., Zou, Y., Lian, L., Wu, X., He, X., He, X., Wu, X., Huang, Y., Lan, P., 2013. Changes of T cells and cytokines TGF- β 1 and IL-10 in mice during liver metastasis of colon carcinoma: Implications for liver anti-tumor immunity. *Journal of Gastrointestinal Surgery* 17, 1283–1291.
- [25] Hubbard, M., Bailey, M.J., 2018. Mastering Microsoft Teams. Mastering Microsoft Teams. <https://doi.org/10.1007/978-1-4842-3670-3>.
- [26] Káldi, T., Babarczy, A., 2021. Linguistic focus guides attention during the encoding and refreshing of working memory content. *Journal of Memory and Language* 116, 104187.
- [27] Lodge, J.M., Harrison, W.J., 2019. Focus: Attention science: The role of attention in learning in the digital age. *The Yale Journal of Biology and Medicine* 92, 21.
- [28] Marek, M.W., Chew, C.S., Wu, W.c.V., 2021. Teacher experiences in converting classes to distance learning in the Covid-19 pandemic. *International Journal of Distance Education Technologies (IJDET)* 19, 40–60.
- [29] Mays, D., Luta, G., Walker, L.R., Tercyak, K.P., 2012. Exposure to peers who smoke moderates the association between sports participation and cigarette smoking behavior among non-white adolescents. *Addictive Behaviors* 37, 1114–1121.
- [30] Memon, J., Sami, M., Khan, R.A., Uddin, M., 2020. Handwritten optical character recognition (OCR): A comprehensive systematic literature review (SLR). *IEEE Access* 8, 142642–142668.
- [31] Mowling, C.M., Sims, S.K., 2021. The metacognition journey: Strategies for teacher candidate exploration of self and student metacognition. *Strategies* 34, 13–23.
- [32] Mufti, T., Mittal, P., Gupta, B., 2021. A review on Amazon web service (AWS), Microsoft azure & Google cloud platform (GCP) services. *European Union Digital Library*.
- [33] Niwattanakul, S., Singthongchai, J., Naenudorn, E., Wanapu, S., 2013. Using of Jaccard coefficient for keywords similarity, in: *Procs. of the International Multiconference of Engineers and Computer Scientists*, pp. 380–384.
- [34] Sanford, A.J., Sanford, A.J., Molle, J., Emmott, C., 2006. Shallow processing and attention capture in written and spoken discourse. *Discourse Processes* 42, 109–130.
- [35] Schuck, K., Otten, R., Engels, R.C., Kleinjan, M., 2012. The role of environmental smoking in smoking-related cognitions and susceptibility to smoking in never-smoking 9–12 year-old children. *Addictive Behaviors* 37, 1400–1405.
- [36] Shelton, A., Lemons, C.J., Wexler, J., 2021. Supporting main idea identification and text summarization in middle school co-taught classes. *Intervention in School and Clinic* 56, 217–223.
- [37] Simon, H.A., 1974. How big is a chunk?: By combining data from several experiments, a basic human memory unit can be identified and measured. *Science* 183, 482–488.
- [38] Smith, P.J., Murphy, K.L., Mahoney, S.E., 2003. Towards identifying factors underlying readiness for online learning: An exploratory study. *Distance Education* 24, 57–67.
- [39] Spark, A., 2018. Apache spark. Retrieved January 17.
- [40] Venditti, A., Fasano, F., Risi, M., Tortora, G., 2018. The importance of interaction mechanisms in blended learning courses involving problem solving e-tivities, in: *Procs. of the 13th International Conference on Digital Information Management (ICDIM)*, pp. 124–129.
- [41] Vijaymeena, M., Kavitha, K., 2016. A survey on similarity measures in text mining. *Machine Learning and Applications: An International Journal* 3, 19–28.
- [42] Wang, M.P., Ho, S.Y., Lo, W.S., Lam, T.H., 2012. Smoking family, secondhand smoke exposure at home, and nicotine addiction among adolescent smokers. *Addictive Behaviors* 37, 743–746.
- [43] Wu, C.Y., 2019. Initiatives for a healthy stomach. *Current Treatment Options in Gastroenterology* 17, 628–635.

Journal of Visual Language and Computing

journal homepage: www.ksiresearch.org/jvlc

YouCare: a Cross-platform Telehealth App for COVID-19

Gennaro Costagliola^a, Mattia De Rosa^{a,*}, Vittorio Fuccella^a and Francesco Vitale^a

^aDipartimento di Informatica, University of Salerno, Via Giovanni Paolo II, 84084 Fisciano (SA), Italy

ARTICLE INFO

Article History:

Submitted 4.23.2021
Revised 5.20.2021
Second Revision 7.30.2021
Accepted 8.15.2021

Keywords:

COVID-19
telehealth
app

ABSTRACT

The COVID-19 pandemic has caused disruption across the globe and put pressure on healthcare systems. In order to limit the use of hospital resources, the use of home care and telehealth has been very important to minimize direct human intervention in monitoring patients. The purpose of this work is to present YouCare: a cross-platform application that allows the collection of medical data on the health status of the user in order to allow physicians to efficiently monitor the status of the patient. As an important feature, it includes functions to monitor the general situation through statistics and interactions with the other users of the application. This might make the isolation period less stressful while exchanging current COVID experiences. The use of the application has been experimented with a usability test, obtaining positive feedback from the users. We also report other similar applications that have been developed and used in different parts of the world.

© 2021 KSI Research

1. Introduction

The COVID-19 pandemic, which spread in Wuhan (Hubei, China) in the early 2020s, caused a major change worldwide, strongly influenced people's lifestyle, and put a lot of pressure on national health care systems, in some cases leading them to collapse (with shortages of both hospital beds, doctors and medical supplies such as masks and oxygen [30]). In a substantial part of the world, therefore, great efforts have been made to improve the situation of healthcare systems, on the one hand, by expanding the capacity of hospital systems, although necessarily limited, and on the other, by expanding the use of home care and telehealth. In the most critical cases, the lack of healthcare personnel (or even just phone contact with a physician) has also negatively affected home care, so systems that can minimize the need for direct human intervention in monitoring patients at home can prove very valuable. However physicians' positive communication skills have a significant psychological effect on COVID-19 patients [2], so limiting direct interaction between patient and health professional may be negative from that perspective. Moreover, for such patients, in addition

to the disease, typically a quarantine at home is imposed as well. Obviously, such a state of isolation and the consequent lack/reduction of social interactions and information about the disease can negatively affect mood.

The purpose of this work is to present YouCare: a cross-platform application (Android / iOS / Web, in order to ensure the widest possible availability), to allow the collection of medical data about the health status of home patients in order to efficiently allow physicians constant monitoring of the patient's status (also providing alerts in the case of critical values). The application also provides some features intended to make the isolation period less stressful.

In particular, the user can fill out a questionnaire in which he/she reports his/her clinical data (body temperature, sore throat, headache, muscle pain, nausea, cough, shortness of breath, bad mood, oxygen saturation level, breathing rate, heart rate, and blood pressure). Moreover the user can get in direct contact with other users through a Forum section, in this way other users (often in similar situations) can offer practical help and support through reply messages. In addition, the user can check statistics about the use of the app and aggregate information about other users' daily questionnaires (so he/she can monitor the condition of the majority of users, to have an idea of the global situation). The user may request a phone contact as well. The person's emotional state may also benefit from the opportunity to interact with health

*Corresponding author

[✉ gencos@unisa.it](mailto:gencos@unisa.it) (G. Costagliola); matderosa@unisa.it (M. De Rosa); vfuccella@unisa.it (V. Fuccella)
ORCID(s): 0000-0003-3816-7765 (G. Costagliola);
0000-0002-6922-5529 (M. De Rosa); 0000-0003-3244-3650 (V. Fuccella)

DOI reference number: 10.18293/JVLC2021-N1-009

professionals and other people who are affected by COVID-19 and the ability to track overall statistics. The app also allows the user to “manage” another person, e.g. an elderly relative who does not have or is unable to use a smartphone or pc.

This paper is an extended version of work published in [12]. We extend our previous work by expanding the empirical evaluation to older people.

The paper is organized as follows: Section 2 describes previous work on COVID-19 apps; Section 3 describes the YouCare application, Section 4 shows its experimental evaluation, and Section 5 the evaluation results. Finally, Section 6 concludes the paper with a discussion on future work.

2. Related Work

Since the onset of the COVID-19 pandemic, numerous technologies and mobile applications have been proposed. Great interest has been devoted to contact tracing apps, with the study of different technologies and architectures, each with different privacy implications [1]. The use of this kind of technology has not been limited to contact tracing, but uses for telemedicine and remote diagnosis have also been explored, since in recent years, devices such as smartphones, smartwatches, and smart bands have seen an increase in usage in the medical and assistive technology fields [19, 10, 31, 4]. Some contact tracing apps have also included such features [3]. In this section, we will briefly focus on applications that offer these functionalities.

Among COVID-19 contact tracing apps, some include the ability for the user to report their health status (self-assessment). As an example, COVID Tracker is an app used by the Irish Health Service Executive for contact tracing on a national basis [14]. COVID Tracker monitors contacts between nearby devices via Bluetooth. Upon detection of contact with a positive, it alerts affected users via anonymous notifications. If the user expresses consent, instead of notification, they can receive a direct phone call from a health care provider. Daily, the user can fill in a short questionnaire in which he/she communicates his/her health situation based on a few parameters (fever, breathing problems, cough, taste, and smell problems). This application also has an informative side, in fact, on the homepage the citizen can read statistics on the progress of the infection in Ireland. Among the available statistics, there are: number of installations of the application, number of tests performed, number of symptomatic and asymptomatic infected people. Other examples of applications that include the facility for self-assessment used in different parts of the world include Mawid (Saudi Arabia) [5], Aarogya Setu (India) [18], NHS COVID-19 (UK) [23], PathCheck SafePlaces (USA) [24]. Another case is the Health Code system for contact tracing that has been integrated in China into two of the most popular apps used by the population, WeChat and Alipay [22]. According to the health code rules peoples are required to enter their personal information, including medical and health status information, into these apps. Based on contact tracing and entered

symptom information the app assigns a health risk status to the person, which regulates his or her movement options and the possible need for medical/health intervention.

Other applications, on the other hand, are designed explicitly to allow the user to report their health status, both in the case of people without a COVID-19 diagnosis (preventive monitoring) and people with a COVID-19 diagnosis (home care). As an example, the Cvm-Health [27] web app, distributed by Sensyne Health in the UK and US, is a COVID-19 monitoring app. Users can sign up for the platform and daily record their vital signs and any COVID-19-related symptoms. The app creates a personalized digital medical record based on the recorded vital sign data and symptom information. This app also offers a social aspect: it allows users to help family and friends who are digitally disconnected by monitoring their health. The platform interacts directly with the NHS (UK National Health System) by sharing the data entered by users. Another example is the e-Covid SINFONIA app [28], used in Campania (Italy), which notifies the user of the outcome of COVID-19 tests performed in regional laboratories, and also allows the user to submit a questionnaire with the main infection indicators. These data are then made available to the general practitioners. Within the application, it is possible to register family members in order to view or receive notifications of their test results. Other examples include [8] in which teleconsultation is performed through a mobile application, tablet, or web browser, and [20] in which the approach of the Cleveland Clinic incorporates a self-monitoring app for patient engagement, monitors symptoms for early intervention.

Some applications, in addition to self-reporting capabilities, make use of wearable devices to detect the person's vital signs [7]. Examples include [16], where wearable devices were used to collect vital signs of people in quarantine, e.g., before or after a trip abroad. This data is monitored by algorithms and a medical team so that signs of deteriorating health can be detected and people can be transferred to a hospital if necessary. In [15], instead, wearable devices were used to monitor patients with chronic illnesses or who are recovering from a COVID-19, so that some patients who would have to be hospitalized could be allowed to stay at home instead. Another example is the ZCare Monitor [32], a home monitoring system for COVID-19 patients in the non-acute phase developed by the Zucchetti Group in collaboration with Doctors Without Borders for the Lodi Hospital, among the first to address the health emergency related to Coronavirus, and used by 11 different hospital facilities, for a total of more than 10,000 patients. The center contacts COVID-19 positive patients who are quarantined at home by phone. Patients, daily, enter their physiological data through the platform. In addition, a pulse oximeter, connected via Bluetooth, sends the following data to the platform twice a day: body temperature, blood saturation level, maximum blood pressure, pulse, and heart rate. This data is collected directly by the hospital platforms and monitored daily, also with the help of software for predicting clinical status (using machine-learning algorithms).

Other research has focused on the possibility of using mobile devices for diagnosis. As an example, in [21], the authors discuss the pre-symptomatic detection of COVID-19 using smartwatch data, also focusing on asymptomatic patients. Given that the asymptomatic status of the disease does not preclude the possibility of infection, it would be helpful to know about this condition. The authors asked study participants to record daily symptoms and share fitness tracker data. The types of data collected included heart rate, steps, and sleep over a period of several months. Two infection detection algorithms (RHR-Diff and HROS-AD) were developed, and based on these data, it was analyzed that 63% of COVID-19 cases could be detected prior to symptom on-set via a two-level alert system based on the occurrence of extreme increases in resting heart rate relative to the individual baseline. Such an alarm would allow, even several days in advance, the recognition of a possible asymptomatic infection and the ability to proceed with standard tests. Other examples include *AI4COVID-19* [17], in which a preliminary COVID-19 diagnosis is performed from cough samples captured through the app, and [25], an app that collects self-reported symptoms, diagnostic testing results, and smartwatch and activity tracker data. This data, collected over time, is used to help identify subtle changes indicating an infection.

3. YouCare

YouCare consists of three components: a cross-platform mobile application (Android/iOs/Web) used by the system's users, a server that receives/responds to requests from that application and stores user data, and a web platform that allows authorized physicians to access their patients' data. Based on what has been seen in the literature and from interviews with healthcare professionals and COVID-19 patients, the functional requirements of the application have been defined. Specifically, the application must allow:

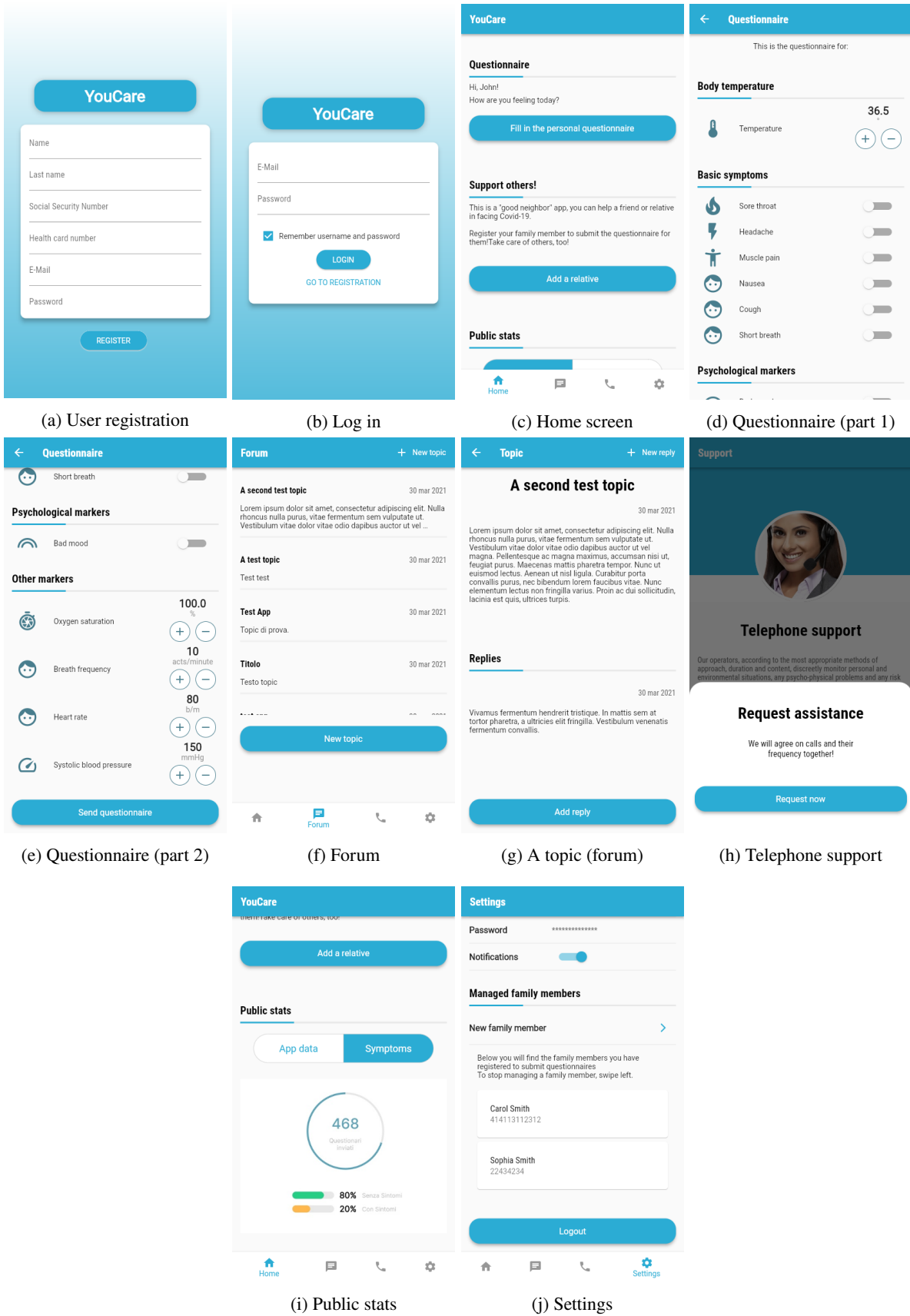
- Registration and log in. The unauthenticated user must be able to login into the app. There must also be the ability to subscribe themselves. The app functionalities should only be available to people who have COVID-19 (or people with suspected COVID-19 and with a scheduled test or waiting for a test result).
- It should be possible to use the app's functionalities on behalf of another person. This is very important given the lower penetration of digital technologies among older people, who are also the category most at risk for COVID-19.
- Symptoms entry and statistics. The user must be able to enter his physiological data, useful for diagnosis, directly in his user area of the system on a daily basis. The application must allow the filling in of the questionnaire also for other users registered as relatives. The user shall also be able to consult global statistics about the users of the system.

- Forum. The user should have access to a Forum section in the application where he can read the topics of other users, reply to the topics and create his own. This should be possible in an anonymous way, possibly even restricting the geographic area in which messages can be viewed.
- Telephone support. The user should have the option of requesting telephone assistance. These requests will be handled by healthcare providers who, by accessing the request list, can make phone calls to the application users.
- View of users' status. The physician accessing the system with his credentials (provided by the administrator) can view the data collected with the questionnaires of his patients. Alerts and statistics about his patients should also be available.

Non-functional requirements include:

- The registration process should be as fast as possible even in cases where the user is subscribing other people.
- The design of the system, especially with regard to the mobile application, must be as intuitive as possible due to the very diverse user base.
- The application must be available for as many devices as possible (both mobile and desktop).
- Access to the data and services offered by the system must meet the most common security standards. No user should be allowed to access the confidential data of other users to which he/she is not entitled. Access to the patient's data should be restricted to their physician.
- The system must be always online.

Based on the identified requirements we proceeded with the development of the app. The app development was performed using the Flutter framework, which allowed the development of a cross-platform Android, iOS, and Web application with a single codebase in Dart. As shown in Figure 1a, the user, in order to access the application must register to the platform by entering his personal data. In order to validate his identity, adapting the app to the Italian national health system, the number printed on the Italian health card is requested (a confidential number of which only the regional health authority should be aware; this validation should be adapted to the health systems of the various countries). This same check is used to allow one person to use the app on behalf of another. However, it is always possible for a person to register themselves, in which case they can decide whether to still allow the other person to use the app on their behalf or to remove that ability. Finally, the app's functionality will only be accessible to individuals for whom there is a positive (or scheduled/pending) COVID-19 test in the regional health



(a) User registration

(b) Log in

(c) Home screen

(d) Questionnaire (part 1)

(e) Questionnaire (part 2)

(f) Forum

(g) A topic (forum)

(h) Telephone support

(i) Public stats

(j) Settings

Figure 1: Application screenshots.

system (again, this also needs to be adapted to the country’s health systems).

After registration, the user can log in (see Figure 1b) and access the home screen of the application (see Figure 1c). From the home screen, the user can access the filling of the questionnaire (see Figure 1d and Figure 1e) that allows the user to enter his medical data daily and forward them to the server, thus allowing their vision to his physician. YouCare also allows the user to fill out questionnaires for other people (for example relatives who do not have a device with Internet access), in fact, the home screen presents a section for handling relatives. It is possible to add a relative by clicking the relative button (see Figure 1c) on the home screen (the number of the health card of the person to be added is required). After that, it is possible to fill in the daily questionnaire with the medical data. At any time, a person added by other users can register as a new user to the platform, thus taking possession of their user. The user, who previously managed the daily questionnaires, is notified of the registration by a notification on the device. YouCare offers features aimed at improving the psychological well-being of quarantined patients. These features include a forum where users can interact with other users (see Figures 1f and 1g), and a “Telephone support” section where users can request a telephone contact from their physicians or a health care professional (see Figure 1h). On the home screen the user can also view some global statistics about the application and the daily questionnaires of the users (see Figure 1i). Together these social and informational functions can be helpful to the person’s emotive state. This information could be of moral support since it will show the run-time collective status of people in the same condition. Finally, there is a “Settings” section (see Figure 1j).

The application communicates with a server that provides a JSON API for the functionality needed by the app and takes care of storing the data entered by the users. The server uses PHP and MySQL technologies and uses the JSON WEB TOKEN standard (JWT - RFC 7519 [6] standard) for authentication management. The Google Firebase Cloud Messaging service [13] is also used to manage YouCare push notifications.

A web-based platform, shown in Figure 2, is also available for physicians to view patient data and medical information. The system administrator can add a physician to the system, and the association between a General Practitioner and his patients can be automatically performed thanks to the health card numbers (at least in countries that provide this association, such as Italy). The doctor can view the data entered by each patient through the daily questionnaires and statistics on the questionnaires. A MEWS scale [29] score is also provided for each patient in order to immediately alert the physician of a possible clinical instability of the patient. The score obtained from the scale ranges from a minimum of 0 to a maximum of 14. Above level 5 the patient is critical and unstable. For all other patients with normal values, however, the MEWS is an important tool for early detection of worsening clinical conditions.

Table 1
Tasks performed by the experiment participants.

	Task
1	Register a personal account from the app
2	Log in
3	Look at the public statistics from the app
4	Fill out a questionnaire for themselves
5	Add a first relative/family member
6	Add a second relative/family member
7	Remove one of the two relatives/family members
8	Fill out a questionnaire for a relative/family member
9	Add a new topic to the forum
10	Reply to one of the existing topics
11	Request phone support from the application

4. Evaluation

We carried out a user study aimed at evaluating the usability of YouCare. In the experiment, we asked participants to use the app’s features and then evaluate the app by filling out a questionnaire.

4.1. Participants

For the experiment, 23 participants who decided to participate for free were recruited. They were divided into two groups based on age. The 20-29 group consisted of 15 participants (2 women), all college students between 20 and 28 years old ($M = 22.3$, $SD = 2.6$), regular users of computers and smartphones. Two of them had personally dealt with COVID-19. The 55-64 group consisted of 8 participants (2 female) between 55 and 64 years old ($M = 58.1$, $SD = 2.9$), all regular smartphone users.

4.2. Apparatus

The experiment was conducted on the individual participants’ smartphones due to restrictions due to COVID-19 that prevented direct contact. All devices are recent Android smartphones from different manufacturers (Samsung, Xiaomi, etc.). The server was running an 8-core server with 32 GB of RAM running Ubuntu 20.04 and with a gigabit internet connection.

4.3. Procedure

Before starting the experiment, the participants filled out a questionnaire with the following information: personal data (age, gender), previous experiences with smartphones, personal experience with COVID-19.

Participants were asked to use the app, performing a list of the most representative tasks (see Table 1). In order to assess the intuitiveness of the app, no explanations were given about it or on how to perform the tasks.

At the end of the experiment, they were asked to submit a form in which they could write free-form comments, both about the app in general and about the performed tasks.

Moreover, they were asked to complete a System Usability Scale (SUS) [9] questionnaire. SUS includes ten statements (see Table 2), to which respondents had to specify

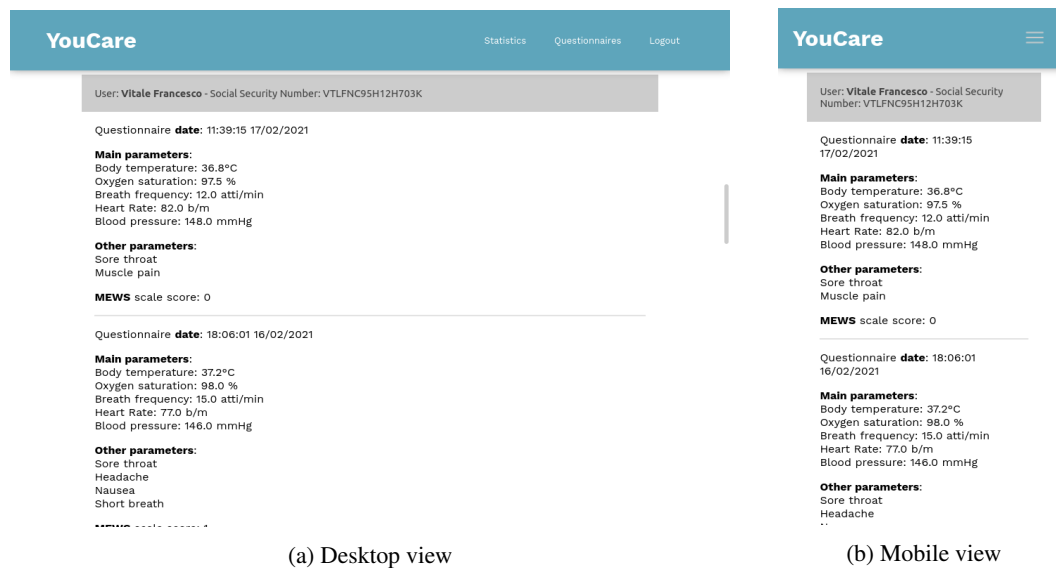


Figure 2: Web-based platform for physicians.

Table 2
SUS questionnaire.

	Question
1	I think that I would like to use this system frequently
2	I found the system unnecessarily complex.
3	I thought the system was easy to use
4	I think that I would need the support of a technical person to be able to use this system
5	I found the various functions in this system were well-integrated
6	I thought there was too much inconsistency in this system
7	I would imagine that most people would learn to use this system very quickly
8	I found the system very cumbersome to use
9	I felt very confident using the system
10	I needed to learn a lot of things before I could get going with this system

their level of agreement using a five-point Likert scale. The questions alternate between positive and negative (since they are in a rather standard form we do not include them here). Each SUS questionnaire has a score between 0 and 100, of which we then calculated the averages on all participants.

Finally, we also collected further feedback through verbal interaction.

5. Results and discussion

All participants completed the experiment. For each participant, the experiment lasted about 20 minutes.

From the server logs, we were able to verify that all participants in both groups successfully completed all tasks.

For the 20-29 group, the average SUS score was 83.3 ($SD = 9.6$), while for the 55-64 group the score was 73.8 ($SD = 11.8$), which are both good values [26].

Regarding free form comments and interviews, for both groups the feedback was generally positive about both the usability of the app and its usefulness. The most common suggestion concerned the ability to view the status (confirmation) of the telephone support request. Other suggestions include adding reminders, integration with smart devices for health monitoring, keeping the name of sections always visible in the UI, adding more information about the general statistics.

Most of the difficulties in using the application were reported by the 55-64 participants, primarily in the interaction required to remove family members. Some participants did not notice at first the navigation buttons at the bottom of the app screen, trying to perform all tasks from within the home screen and thus having difficulty when performing the first task for which their use was needed. A single participant had difficulty performing 4 tasks, finding the interaction mode not intuitive. Finally, a bug that occasionally occurred when adding/removing family members was reported.

The obtained results may be regarded as good, since even if the 55-64 group had more difficulties in carrying out some tasks, all participants managed to successfully complete all tasks. It must be noted, nonetheless, that all participants were regular smartphone users. People with little or no experience with them, or older people, might have more difficulties. However, this is mitigated by the fact that one person can use the app in behalf of another person.

6. Conclusions and further works

In this paper, we presented YouCare, a COVID-19 patient telehealth application whose application design was influenced by the analysis of existing applications and interviews with COVID-19 patients.

Future work includes the possibility of directly interfacing the application with Bluetooth wearable devices in or-

der to allow for the collection of some of the medical data without the need for the user to fill out the questionnaire in an ongoing and more reliable manner. It is also intended to integrate a video call feature to complement the phone contact and allow users to receive COVID-19 test reports and medication prescriptions from their doctor directly from the application.

7. Acknowledgment

This work was partially supported by the grant “Fondo FSC 2014 2020 per il Piano Stralcio Ricerca ed Innovazione 2015-2017 - MIUR - progetto MEDIA” (project code: PON03PE_00060_5/12 - D54G14000020005).

References

- [1] Ahmed, N., Michelin, R.A., Xue, W., Ruj, S., Malaney, R., Kanhere, S.S., Seneviratne, A., Hu, W., Janicke, H., Jha, S.K., 2020. A survey of COVID-19 contact tracing apps. *IEEE Access* 8, 134577–134601. doi:10.1109/ACCESS.2020.3010226.
- [2] Al-Zyouid, W., Oweis, T., Al-Thawabih, H., Al-Saqqar, F., Al-Kazwini, A., Al-Hammouri, F., 2021. The psychological effects of physicians’ communication skills on COVID-19 patients. *Patient Preference and Adherence* Volume 15, 677–690. doi:10.2147/ppa.s303869.
- [3] Alanzi, T., 2021. A review of mobile applications available in the app and google play stores used during the COVID-19 outbreak. *Journal of Multidisciplinary Healthcare* Volume 14, 45–57. doi:10.2147/jmdh.s285014.
- [4] Angellotti, F., Costagliola, G., De Rosa, M., Fuccella, V., 2020. ifree: design and evaluation of a pointing method for disabled users on mobile devices, in: 2020 IEEE International Conference on Human-Machine Systems (ICHMS), pp. 1–6. doi:10.1109/ICHMS49158.2020.9209537.
- [5] Arab News, 2020. Saudi arabia’s mawid smartphone app offers coronavirus self-assessment. URL: <https://www.arabnews.com/node/1652171/saudi-arabia>.
- [6] Auth0. Json web tokens. URL: <https://jwt.io>.
- [7] Best, J., 2021. Wearable technology: covid-19 and the rise of remote clinical monitoring. *BMJ* 372. URL: <https://www.bmj.com/content/372/bmj.n413>, doi:10.1136/bmj.n413.
- [8] Bourdon, H., Jaillant, R., Ballino, A., El Kaim, P., Debillon, L., Bodin, S., N’Kosi, L., 2020. Teleconsultation in primary ophthalmic emergencies during the COVID-19 lockdown in paris: Experience with 500 patients in march and april 2020. *Journal Français d’Ophthalmologie* 43, 577–585. URL: <https://www.sciencedirect.com/science/article/pii/S0181551220302497>, doi:10.1016/j.jfo.2020.05.005.
- [9] Brooke, J., et al., 1996. Sus-a quick and dirty usability scale. *Usability evaluation in industry* 189, 4–7.
- [10] Cipriano, M., Costagliola, G., De Rosa, M., Fuccella, V., Shevchenko, S., 2021. Recent advancements on smartwatches and smartbands in healthcare, in: *Innovation in Medicine and Healthcare*, Springer Singapore. pp. 117–127. doi:10.1007/978-981-16-3013-2_10.
- [11] Costagliola, G., De Rosa, M., Fuccella, V., 2018. A technique for improving text editing on touchscreen devices. *Journal of Visual Languages & Computing* 47, 1–8. URL: <https://www.sciencedirect.com/science/article/pii/S1045926X17302422>, doi:10.1016/j.jvlc.2018.04.002.
- [12] Costagliola, G., De Rosa, M., Fuccella, V., Vitale, F., 2021. YouCare: a COVID-19 telehealth app, in: *The 27th International DMS Conference on Visualization and Visual Languages*, pp. 55–62. doi:10.18293/DMSVIVA2021-009.
- [13] Google. Firebase cloud messaging. URL: <https://firebase.google.com/docs/cloud-messaging>.
- [14] Health Service Executive. COVID Tracker app - Ireland’s coronavirus contact tracing app. URL: <https://covidtracker.gov.ie>.
- [15] Hospital Times, 2020. Northampton nhs at forefront of innovation for virtual ward tech. URL: <https://www.hospitaltimes.co.uk/northampton-nhs-at-forefront-of-innovation-for-virtual-tech/>.
- [16] Imperial College London, 2020. Wearable sensor trialled for remote COVID-19 monitoring. URL: <https://www.imperial.ac.uk/news/196973/wearable-sensor-trialled-remote-covid-19-monitoring/>.
- [17] Imran, A., Posokhova, I., Qureshi, H.N., Masood, U., Riaz, M.S., Ali, K., John, C.N., Hussain, M.I., Nabeel, M., 2020. AI4COVID-19: Ai enabled preliminary diagnosis for COVID-19 from cough samples via an app. *Informatics in Medicine Unlocked* 20, 100378. URL: <https://www.sciencedirect.com/science/article/pii/S2352914820303026>, doi:10.1016/j.imu.2020.100378.
- [18] Jhunjhunwala, A., 2020. Role of telecom network to manage COVID-19 in india: Aarogya setu. *Transactions of the Indian National Academy of Engineering* 5, 157–161. doi:10.1007/s41403-020-00109-7.
- [19] Lee, J.H., 2016. Future of the smartphone for patients and healthcare providers. *Health Inform Res* 22, 1–2. URL: <http://e-hir.org/journal/view.php?number=855>, doi:10.4258/hir.2016.22.1.1.
- [20] Medina, M., Babiuch, C., Card, M., Gavrilesco, R., Zafirau, W., Boose, E., Giuliano, K., Kim, A., Jones, R., Boissy, A., 2020. Home monitoring for COVID-19. *Cleveland Clinic Journal of Medicine* doi:10.3949/ccjm.87a.ccc028.
- [21] Mishra, T., Wang, M., Metwally, A., Bogu, G., Brooks, A., Bahmani, A., Alavi, A., Celli, A., Higgs, E., Dagan-Rosenfeld, O., Fay, B., Kirkpatrick, S., Kellogg, R., Gibson, M., Wang, T., Hunting, E., Mamic, P., Ganz, A., Rolnik, B., Li, X., Snyder, M., 2020. Pre-symptomatic detection of COVID-19 from smartwatch data. *Nature Biomedical Engineering* 4. doi:10.1038/s41551-020-00640-6.
- [22] Mozur, P., Zhong, R., Krolik, A., 2020. In coronavirus fight, china gives citizens a color code, with red flags. URL: <https://www.nytimes.com/2020/03/01/business/china-coronavirus-surveillance.html>.
- [23] NHS UK. The NHS test and trace app support website. URL: <https://covid19.nhs.uk>.
- [24] Pathcheck Foundation. PathCheck foundation | COVID-19 technology & research. URL: <https://pathcheck.org>.
- [25] Quer, G., Radin, J.M., Gadaleta, M., Baca-Motes, K., Ariniello, L., Ramos, E., Kheterpal, V., Topol, E.J., Steinhilb, S.R., 2020. Wearable sensor data and self-reported symptoms for COVID-19 detection. *Nature Medicine* 27, 73–77. doi:10.1038/s41591-020-1123-x.
- [26] Sauro, J., 2018. 5 ways to interpret a sus score. URL: <https://measuringu.com/interpret-sus-score/>.
- [27] Sensyne Health. Cvm-health. URL: <https://www.cvm-health.com>.
- [28] So.Re.Sa. S.p.a.. e-Covid SINFONIA app. URL: https://www.soresa.it/Pagine/e-covid_sinfonia.aspx.
- [29] Subbe, C., Kruger, M., Rutherford, P., Gemmel, L., 2001. Validation of a modified Early Warning Score in medical admissions. *QJM: An International Journal of Medicine* 94, 521–526. URL: <https://academic.oup.com/qjmed/article-pdf/94/10/521/4444063/940521.pdf>, doi:10.1093/qjmed/94.10.521.
- [30] Taylor, L., 2021. COVID-19: Brazil’s hospitals close to collapse as cases reach record high. *BMJ* 372. URL: <https://www.bmj.com/content/372/bmj.n800>, doi:10.1136/bmj.n800.
- [31] Vitiello, G., Sebillio, M., Fornaro, L., Di Gregorio, M., Cirillo, S., De Rosa, M., Fuccella, V., Costagliola, G., 2018. Do you like my outfit? cromnia, a mobile assistant for blind users, in: *Proceedings of the 4th EAI International Conference on Smart Objects and Technologies for Social Good*, Association for Computing Machinery, New York, NY, USA. pp. 249–254. doi:10.1145/3284869.3284908.
- [32] Zucchetti s.p.a., 2020. Zucchetti continues its solidarity commitment with zcare monitor. URL: <https://www.zucchetti.it/website/cms/comunicato-stampa-dettaglio/8661-zucchetti-continua-il-suo-impegno-solidale-con-zcare-monitor.html>.

Journal of Visual Language and Computing

journal homepage: www.ksiresearch.org/jvlc

PADD: Dynamic Distance-Graph based on Similarity Measures for GO Terms Visualization of Alzheimer and Parkinson diseases

Alessia Auriemma Citarella^{a,*}, Fabiola De Marco^a, Luigi Di Biasi^a, Michele Risi^a and Genoveffa Tortora^a

^aDepartment of Computer Science, University of Salerno, 84084 Fisciano (SA), Italy

ARTICLE INFO

Article History:

Submitted 7.31.2021
Revised 8.11.2021
Accepted 8.20.2021

Keywords:

Protein Visualization
Gene Ontology
Clustering

ABSTRACT

In the biological field, having a visual and interactive representation of data is useful, particularly when there is a need to investigate a large amount of multilevel data. It is advantageous to communicate this knowledge intuitively because it helps the users to perceive the dynamic structure in which the correct connections are present and can be extrapolated. In this work, we propose a human-interaction system to view similarity data based on the functions of the *Gene Ontology* (Cellular Component, Molecular Function, and Biological Process) of the proteins/genes for Alzheimer disease and Parkinson disease. The similarity data was built with the Lin and Wang measures for all three areas of Gene Ontology. We clustered data with the K-means algorithm in order to demonstrate how information derived from data can only be partial when using traditional display methods. Then, we have suggested a dynamic and interactive view based on SigmaJS with the aim of allowing customization in the interactive mode of the analysis workflow by users. To this aim, we have developed a first prototype to obtain a more immediate visualization to capture the most relevant information within the three vocabularies of Gene Ontology. This facilitates the creation of an omic view and the ability to perform a multilevel analysis with more details which is much more valuable for the understanding of knowledge by the end users.

© 2021 KSI Research

1. Introduction

In the latest years, it is becoming increasingly vital to have an omic vision in order to define biological systems at an ever-increasingly granular level. The goal of omic sciences is to generate useful knowledge which can be utilized to feature and interpret biological systems [18].


For omic sciences we refer to the wide range of biomolecular disciplines characterized by the suffix -omics including genomics, transcriptomics, proteomics, and metabolomics. In this perspective, technological innovation aids the growth of complex system biology by allowing researchers to investigate various intrinsic and extrinsic influences and events

at the base of life. Biological data is multidimensional and highly interdependent. The current challenge is to acquire a more detailed integrative view of the dynamics of cellular processes in a cell or organism enriched in biological and spatial-temporal information [19]. For this purpose, clear visualization methods can provide more immediate access to their content information.

The visualization of biological data has become increasingly relevant in Biosciences, as O'Donoghue *et al.* [14] point out because it helps researchers to interpret heterogeneous data more quickly and easily. One of the most current issues in omic data analysis is the inability to investigate relationships between multi-omic states to incorporate them and combine higher-level expertise [23].

In this paper, we report the preliminary results achieved regards visualization of the similarity of the proteins based on the protein annotations. Protein similarity visualization not based on sequence alignment can be tricky due to inter-class dissimilarities and inter-class similarity [1]. Clustering

*Corresponding author

 aauriemmacitarella@unisa.it (A. Auriemma Citarella);

fdemarco@unisa.it (F. De Marco); ldibiasi@unisa.it (L. Di Biasi);

mrisi@unisa.it (M. Risi); tortora@unisa.it (G. Tortora)

ORCID(s): [0000-0002-6525-0217](https://orcid.org/0000-0002-6525-0217) (A. Auriemma Citarella);

[0000-0003-4285-9502](https://orcid.org/0000-0003-4285-9502) (F. De Marco); [0000-0002-9583-6681](https://orcid.org/0000-0002-9583-6681) (L. Di Biasi);

[0000-0003-1114-3480](https://orcid.org/0000-0003-1114-3480) (M. Risi); [0000-0003-4765-8371](https://orcid.org/0000-0003-4765-8371) (G. Tortora)

DOI reference number: 10.18293/JVLC2021-N1-013

and Machine Learning methods may not be able to extract interdependencies between objects effectively [9]. This fact often does not allow us to generate a clear visual representation of the information.

Our goal is to demonstrate how a human-assisted dynamic graph construction can help abstract functional relationships between proteins in order to generate a clear data visualization when a traditional clustering technique fails. For this contribution, we focused on two diseases: *Alzheimer* and *Parkinson*, the two most common neurodegenerative conditions. Alzheimer's disease (AD) is a form of degenerative dementia that occurs after 65 years. In this pathology, there is a deposition of an $A\beta$ peptide B with the formation of senile plaques and the intracellular aggregation of *tau* protein [5]. Parkinson's disease (PD) is the second most common neurodegenerative disorder in the senile age in which neuronal loss is found in the substance nigra and formation of α -synuclein aggregates that are neuropathological [15].

These pathologies show similar neurodegeneration mechanisms supported by scientific evidence with genetic, biochemical, and molecular studies. Pathological pathways involving α -synuclein and *tau* proteins, oxidative stress, mitochondrial dysfunction, iron pathway, and *locus coeruleus* are among these findings [22]. Because of the overlap in their pathogenic mechanisms, they were chosen as an example for our search workflow. This feature introduces intra- and extra-class overlaps which can deceive typical clustering algorithms.

This paper is an extension of the work *Gene Ontology Terms Visualization with Dynamic Distance-Graph and Similarity Measures* [2]. We have restructured some sections of the paper, enriching the description of the approach with more details. We included two new figures (Figure 4 and Figure 5) which depict the graphical representation of the molecular function of AD and PD proteins, respectively. In addition, the chord diagrams of the recoverable information following the usage of similarity matrices have been provided as an overview (Figures 9-12). We also added further results in Section 5. In particular, we calculated the similarity between all the proteins of both diseases for the molecular function, the biological process and the cellular component. We also extracted the proteins in common to AD and PD, giving an overview of the information that can be recovered from these findings.

The paper is structured as follows. In Section 2 we describe the most important related works in the examined field. In Section 3 we discuss respectively the datasets, methodology, and performance measures which we have used in our research. Finally, we expose the visual results in Section 4 and overall results in Section 5. The conclusions with future work are outlined in Section 6.

2. Related Work

In the literature, several web interfaces can query the terms of the Gene Ontology. The *Gene Ontology* (GO) is a bioinformatics project which uses ontologies to enable the standardization of biological information regarding gene and

gene products properties. It is structured as an acyclic oriented graph where each GO-term is identified by a word or strings and a unique alphanumeric code [8]. The GO database is the most widely utilized resource for enrichment analysis.

QuickGO allows us to find and display GO terms and generate a list of correspondence results based on the user's question. This tool returns a directed acyclic graph (DAG) containing a single GO term and its associated terms and annotations. It is designed with JavaScript, Ajax, and HTML. Statistics with interactive graphs and views of term location tables are available on the fly, indicating which words are frequently noted simultaneously. The user can create a subset of annotations based on different parameters (Specific protein, Evidence Codes, Qualifier Data, Taxonomic Data, Go Terms) and download them [3].

*Gorilla*¹ identifies enriched GO terms in ordered lists of genes using simple, intuitive, and informative graphics, without explicitly requiring the user to provide targets or background sets. It is a GO analysis tool that employs a statistical approach with flexible thresholds to identify GO terms significantly enriched at the top of a classified gene (very useful when genomic data can be represented as a classified list of genes). The analysis's results are presented in the form of a hierarchical structure that allows for a clear view of the GO terms [6].

Blast2GO (B2G)² is an interactive platform that supports non-model species functional genomic research. It is a data sequence-based tool that combines high-performance analysis techniques and evaluation statistics with a high degree of user interaction. Similarity searches produce results on direct acyclic graphs [4].

*NaviGO*³, in order to measure the similarity or relation between the terms of the GO, use six different scores: Resnik, Lin, and the relevant semantic Similarity score for semantic similarity, and *Co-occurrence Association Score* (CAS), *PubMed Association Score* (PAS), and *Interaction Association Score* (IAS) for GO associations. A *Funsim* score for functional similarity is also introduced [21].

More recently, the open-source software *AEGIS* allows us to visually explore the GO data in real-time, taking into input the entire dataset GO. Any Go terms can be chosen as the anchor and have a root, leaf, or waypoint, represented with a DAG. Each source can include all the descendants of the anchor term, the leaves will only include the ancestors, and the Waypoint anchors will constitute a DAG consisted of both ancestors and descendants [25].

3. Methods

In this work, we have used the R environment⁴, a free software environment for statistical computing and graphics, and SigmaJS, a JavaScript library dedicated to graph draw-

¹Gorilla: <http://cbl-gorilla.cs.technion.ac.il>

²Blast2GO: <https://www.biobam.com>

³NaviGO: <https://kiharalab.org/web/navigo/views/goset.php>

⁴R: <https://www.r-project.org>

ing⁵. We used the standard SigmaJS renderer to show the graph view.

3.1. Datasets

Protein datasets for AD and PD, belonging to *Homo Sapiens*, were downloaded from UNIPROT [17]. Data cleaning has been carried out, removing all duplicates. Furthermore, for each UNIPROT ID, the reference gene has been obtained and linked to the STRING. STRING database allows us to consider any protein-protein interactions (PPI) based on a score calculated on experimental evidences [16]. This step is required to eliminate those proteins that are not mapped in the database and do not have the protein-protein interaction that we are looking for. We have recovered a total of 216 genes for AD and 137 genes for PD.

3.2. Gene Ontology

The Gene Ontology is based on two types of relationships between objects: *instances* and *part of*. All organisms share three biological domains which can be considered as structured and controlled vocabularies:

- *Biological Process*: refers to all those events that take place within an organism resulting from an orderly set of molecular functions;
- *Cellular Component*: concerns the location of the entity in question at the level of cellular and/or subcellular structures;
- *Molecular Function*: describes the processes that occur at the molecular level.

We have identified these domains as biological process (BP), cellular component (CC), and molecular function (MF). We have recovered from UNIPROT⁶ all the GO terms belonging to these three fields both for Alzheimer’s and Parkinson’s diseases with UniProt package in R.

3.3. Experimental setup

We explored two ways to calculate semantic similarity. In the first case, we calculated the similarity between proteins of Alzheimer disease and proteins of Parkinson disease for all three ontology gene domains. We considered both Lin’s similarities and Wang’s method. For simplicity, in this work we only show the results concerning the similarity of Lin while the future tool will allow user the setting of both measures. Subsequently, we clustered the data obtained for both similarity measures in BPs, CCs, and MFs domains for AD and PD with the K-means algorithm, trying with $n=3$ and $n=5$ clusters. In the second case, we calculated the similarity based on the Wang and Lin methods between the two sets of protein data of diseases about BPs, DCs and MFs domains in order to compare these measures.

⁵SigmaJS: <https://sigmaj.s.org>

⁶UniProt: <https://www.uniprot.org>

3.4. Distance Metrics

We used two types of metric to compute pairwise semantic similarities, *Lin* and *Wang*, calculated with the GOSemSim package in R [24].

3.4.1. Lin’s measure

Lin measure is based on *information content* (IC). The negative log of a concept’s probability is formally known as IC. This method computes the ratio between the amount of “common information” and the amount of “total information” in the descriptions regards an object pair. This ratio corresponds to the similarity between two objects [12].

In this case, this approach can measure the similarity of the knowledge content of the GO terms for each protein dataset, proteins of AD e proteins of PD. The frequency of two GO words and their closest common ancestor in a particular corpus of GO annotations are used in the estimation. The term *Least Common Subsummer* (LCS) suggests the most basic definition that two concepts share as an ancestor. So, we can consider the following Equation 1:

$$sim_{lin} = \frac{2 * IC(lcs(c_1, c_2))}{IC(c_1) + IC(c_2)} \quad (1)$$

where c_1 and c_2 are two concepts, *IC* is the information content and *lcs* is the function that computes the least common subsammer. In our experiment, c_1 and c_2 reflect the concepts represented by the GO terms referring to the BP, CC, and MF domains. The similarity is calculated for both AD and PD across all proteins in the pathological reference dataset.

3.4.2. Wang measure

The Wang method is based on a *graph-based* semantic similarity. The GO terms are converted into a numeric value by aggregating the terms of their ancestors in a GO graph [20].

Given two GO terms, A and B , we can represent $DAG_A = (A, T_A, E_A)$ and $DAG_B = (B, T_B, E_B)$, where T_n is the set of GO terms including the term n and all of its ancestor terms in the GO graph while E_n are the semantic relations represented as edges between the GO terms. The semantic similarity between these two terms are calculated as in Equation 2:

$$S_{GO}(A, B) = \frac{\sum_{t \in T_A \cap T_B} S_A(t) + S_B(t)}{SV(A) + SV(B)} \quad (2)$$

where $S_A(t)$ and $S_B(t)$ denote the S-value of a GO term t related to term A and term B .

Wang measures the semantic meaning of GO term n , $SV(n)$, after obtaining the S-values for all terms in DAG_n with the Equation 3, represented below:

$$SV(n) = \sum_{t \in T_n} S_n(t) \quad (3)$$

3.5. K-means

K-means is one of the most common and widely used partitioning clustering algorithms which divides a set of objects into K clusters based on their attributes [13]. A cluster

is simply an aggregation of data based on similarities. The division into K clusters is done *a priori*, based on the goal to be achieved or using heuristic techniques and the clusters represent the number of centroids required by the dataset. A centroid is a real or imaginary point that represents the center of the cluster and it is updated with each algorithm iteration.

The procedure is composed by four steps:

- *Step 1:* determine the value of K ;
- *Step 2:* randomly select K points as initial centers of the clusters;
- *Step 3:* assign each new point to the cluster with the closest Euclidean distance to its center. Formally, if c_i is a centroid of the set of centroids C then each point x will be assigned to a cluster based on the following equation (Equation 4):

$$\arg \min_{c_i \in C} \text{dist}(c_i, x)^2 \quad (4)$$

where $\text{dist}(\cdot)$ represents the Euclidean distance;

- *Step 4:* recalculate the updated cluster centers by averaging the points associated with each cluster (Equation 5):

$$c_i = \frac{1}{|S_i|} \sum_{x_j \in S_i} x_j \quad (5)$$

where S_i is the cluster's set of points.

The procedure repeats steps 3 and 4 until a convergence is achieved. The algorithm ensures speed of execution while leaving the data free to group and move away. For the purpose of our study, the maximum number of clusters of the K-means is limited to five. No PCA techniques were used. When the concept of similarity associated to the GO is considered, this constraint is tied to the core premise that a smaller number of clusters can be useful for biological scope. At the same time, when K is less, the K-means allows us to preserve this information but not to view it intuitively. Without a clear display of the data, the end user could not correctly interpret the results. It is necessary to represent such data as clearly as possible in order to translate it into knowledge. We attempted to collect the various forms of information from the three GO domains in order to organize and view them together.

3.6. Dynamic Distance-Graph

Based on the information presented in the previous sections, we propose a *dynamic build cyclic distance graph* (DCDG) to visualize and transfer knowledge regarding the GO terms. Our goal is to provide a clearer visualization of the GO interconnections than other visualization methods like clustering or partitioning. We used a web-based workspace built with Javascript and SigmaJS to allow the user to explore this interconnection. Workspace is designed to be as clean as possible. It starts as an empty web app with

a single callable overlay menu on the upper left corner, allowing users to search the entry point protein into datasets.

The BP, CC, and MF distance matrices, calculated before the execution of the k-means algorithm, were used as input datasets. When selected, the entry protein becomes the root of the graph. Users can click on each graph node to show a context menu (as depicted in Figure 1) in which it is possible to choose extension (explosion) operation for the node itself.

We defined three kinds of extensions for this contribution, each of them related to one dataset: BP, CC and MF, whose definitions are those intended by the three vocabularies of the GO. The distance between each node pairs is written on the arcs between them. This value, which defines the similarity measure, provides the reading key to display protein through the dynamic build cyclic distance graph. Proteins are connected to each other from these values that allow us to explore the graph taking into account the resemblance values between biological process, molecular function and cellular component. Also, the distance value is used to separate nodes into spaces.

The ForceAtlas2 algorithm is used to avoid overlapping between near nodes. In particular, we used ForceAtlas2 embedded into SigmaJS [11]. ForceAtlas2 is a layout algorithm for force-directed graphs. This algorithm allows us to position each node depending on the other nodes using the distances between them as edge weights. Just because of this condition, the position of a node must always be confronted with the other nodes. The fundamental advantage of using ForceAtlas2 for the representation of protein graphs is to have an easier view of the structure because the structural proximity present in the original datasets is converted to visual proximity.

In order to better empathize the functionality distance between GO, we defined a spatial distance SD with the following equation (Equation 6). Given two nodes, A and B and their own distance d :

$$SD = \log_e(d) \quad (6)$$

where d is the distance and the \log_e is the natural logarithm with the number of Nepero as base.

Note that SD is used only for graphical purposes in the rendering routines. Figure 7 shows no linear proportionality into edge lengths: see the distance between (Q8IZY2, Q9BS0) and (Q93045, Q9BS0). Still, for graphical purposes, we defined a threshold th_i as the mean of all the distances into the dataset i used for node expansion. As an example, given the node Q9BXS0 (see Figure 7), the threshold for the protein Q9BXS0 is the mean of the edge's weight between Q9BXS0 and the related nodes. When the distance SD between two node A and B is greater than th_i , then node A and B are considered belonging to a different cluster. A dotted line renders each class separation. For the first prototype of the proposed method see the [Prototype Page](#)⁷. The input requires a symmetry (or distance) matrix in TSV format. After clustering, it is also possible to download the table of coordinates between the various proteins, represented here graph-

⁷<https://smcovid19.org/simtest/>

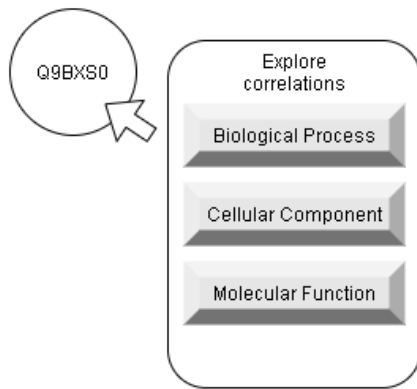


Figure 1: The contextual menu is available for each node.

ically as dynamic dots. The prototype is still being updated for further improvements to guarantee the user full control of the visualization process.

4. Results

4.1. K-means visualization

Figures 2-5 report how the GO objects are partitioned regarding the BP and MF features for AD and PD, with K equal to 3 and 5. The axis reports the distance between each item to its centroid. We used `cluster` and `factoextra` packages in R to perform clusterization. We considered only the Lin's measure for graphical example. We have found that clustering with the K-means algorithm produces visually misleading and uninformative overlaps. This is due to the density of clusters that involve very close intra-cluster distances.

4.2. DCDG visualization

To test our methods, we used protein data based on calculated similarity of Lin. In particular, we considered the G9BXS0 protein from the similarity matrices and we identified the proteins of its neighborhood to build our view of node expansion. Before testing DCDG view, we carried out a simple statistic of the common GO terms, for the only BP component, between this *root* protein and its neighbors. We represented them with a Venn diagram [10] (see Figure 6), on the basis of GO Lin's similarity matrix.

In this scenario, each protein is represented by a closed curving line in the Venn diagram (a circle). A set of GO terms is associated with each protein. In our representation the overlapping area of the circles measures the size of common GO terms for the BP among the proteins. So, this view allows us to evaluate how many common elements are among the different sets of the terms GO for all the selected proteins. It is evident that a simple analysis of terms provides no helpful information beyond the simple observation that there are terms common to all five sets of GO terms for each protein. Instead, introduce similarity based on the *information content* of the GO terms is useful for expanding knowledge regarding biological aspects that would be omitted by a simple statistical analysis.

Figure 7 shows the BP expansion with the DCDG view for the node G9BXS0, a protein produced by *COL25A1* gene for *Homo Sapiens* organism. This protein inhibits the fibrillization of β -amyloid peptide which constitutes amyloid plaques present in Alzheimer's disease. It also assembles the amyloid fibrils in aggregates which are resistant to the demerger mechanisms.

The DCDG view allows the user to see and understand immediately the proteins belonging to the two distinct BP classes: **CLASS 1**, related to many biological processes such as signaling pathway and positive and negative regulation of cellular and chemical complexes and **CLASS 2**, concerning the organization of fibrils, microtubules, and structures of the cytoskeleton.

Figure 8 highlights the successive expansion of Q8IZY2 and Q9POL2 proteins. Due to distances, a new class was identified by the system (**CLASS 3**). In terms of biological meaning, the visualization clearly shows that the additional third class emphasizes further involvement of proteins indicated in different biological processes compared to previous classes. In particular, this class intervenes in broader biological regulation processes involving energy homeostasis and cell cycle regulation systems.

5. AD and PD similarity

Diseases similarity can be determined based on three domains of the GO: molecular function (MF) similarity, biological process similarity (BP) and cellular component similarity (CC). We used the `GoSemSim` package [24]. We used Wang's technique, which leverages the graph structure topology for the GO to compute semantic similarity between the two sets of Alzheimer's and Parkinson's proteins. We have also calculated the similarity of Lin, based on the IC of the three GO domains, between AD e PD in order to compare the differences between these two used methods, as reported in Table 1. We can note as the values are similar for both similarity measure, except for a 5% waste for BP. In Table 2 we reported the common proteins between the two diseases with their ID UNIPROT and the description for each of them. Based on the similarities of BP, MF and CC, we can build a protein network for each of the three domains under consideration. This could respond to the end user request regarding the presence of similar proteins in the function, biological process or cellular location of a series of disorders. As example, in Figure 9 and Figure 10, the similarities of the BP and MF domains for the P03886 protein, present in the AD and PD, are shown. The threshold chosen for the representation is 80%. The protein in question is highlighted in the chord graph. With the threshold previously chosen for BP and MF, the similarities between proteins in PD and AD are depicted as a whole in Figure 11 and Figure 12.

6. Conclusion

Graphs are the most natural way to model interactions between entities in many fields. Dynamic graph representations result from the intrinsically dynamic character of such

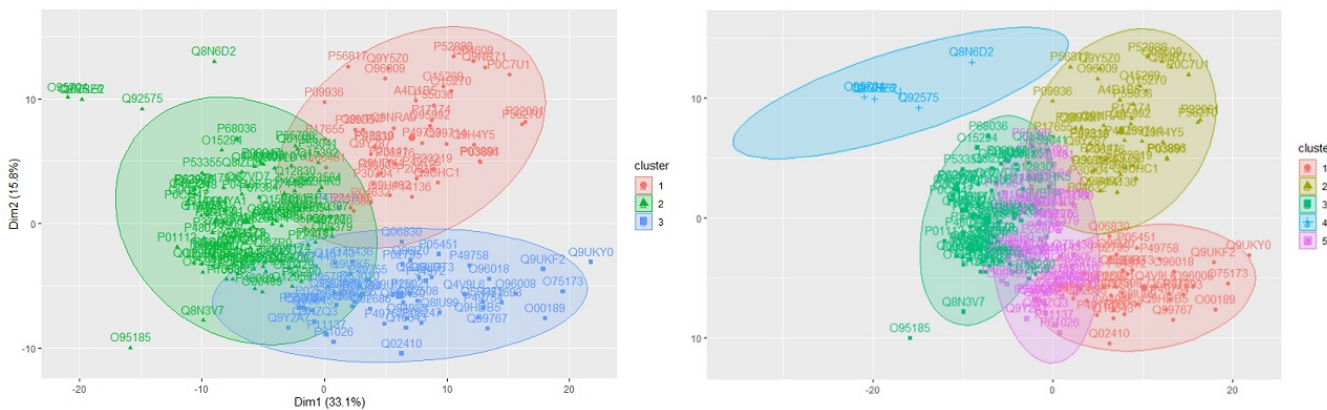


Figure 2: K-means for BP for AD with Lin's measure ($K=3$ on the left and $K=5$ on the right).

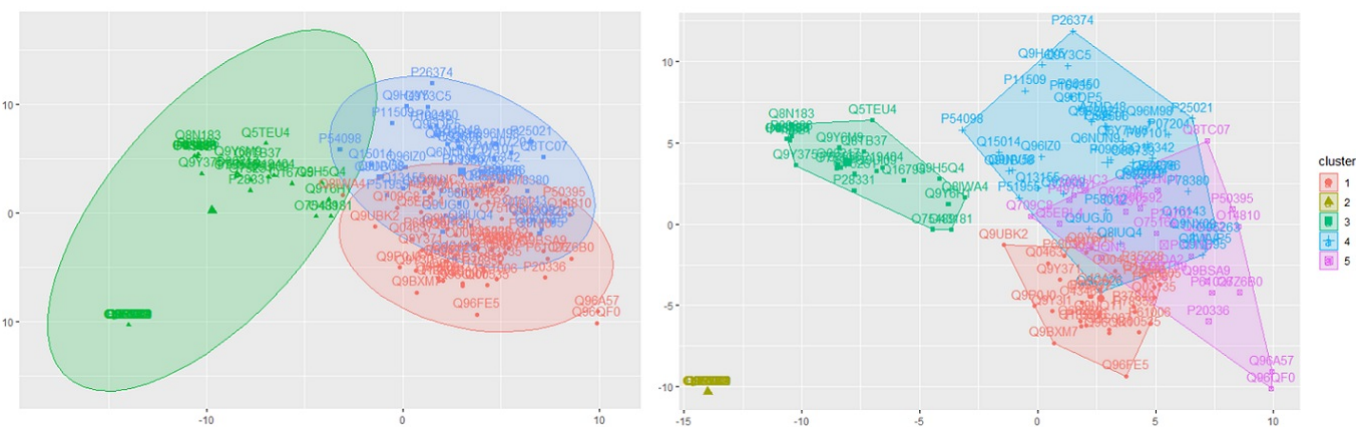


Figure 3: K-means for BP for PD with Lin's measure ($K=3$ on the left and $K=5$ on the right).

data [7]. In this paper, we explored an alternative way to graphically view the relationships between the GO terms based on their information content. In particular, we have proposed a *human interaction*-based viewing system that allows the users to have a complete omic vision of data. In particular, we ensured the direct representation of the inter-

class and intra-class correlations between involved proteins. The strategy proposes an instrument to investigate the GO with a customizable and flexible approach providing information to a more general or selective level.

We presented a distance cyclic distance graph (DCDG) as a GO terms visualization approach to immediately repre-

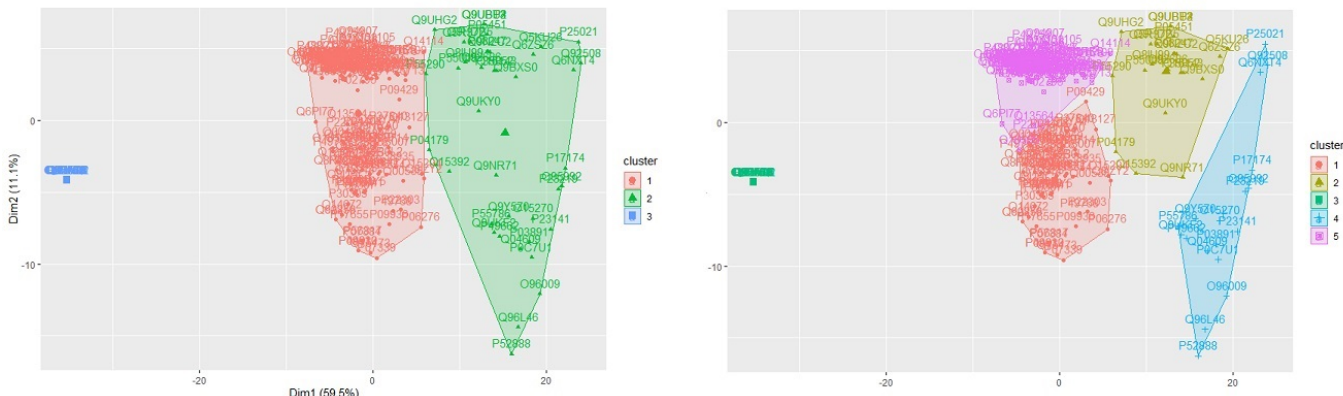


Figure 4: K-means for MF for AD with Lin's measure ($K=3$ on the left and $K=5$ on the right).

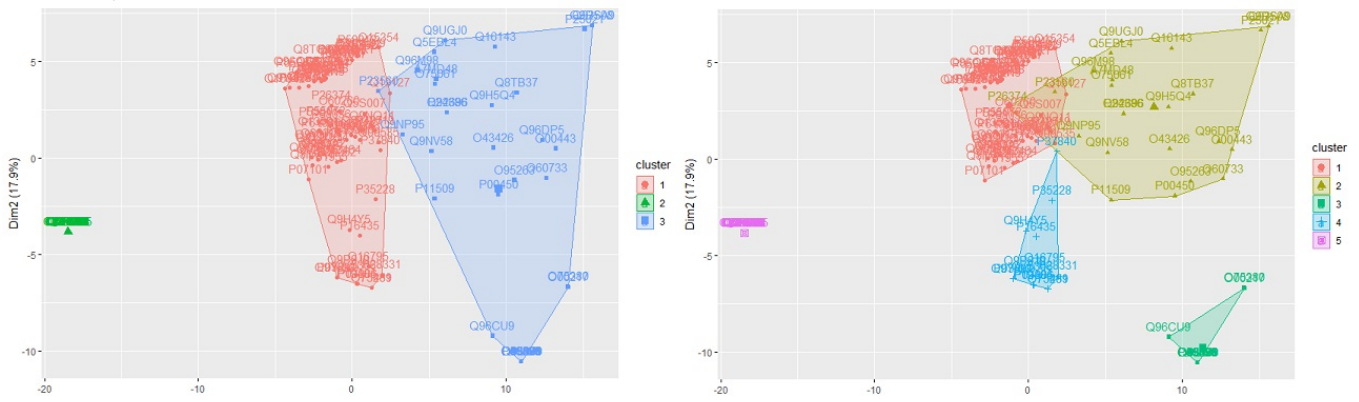


Figure 5: K-means for MF for PD with Lin's measure ($K=3$ on the left and $K=5$ on the right).

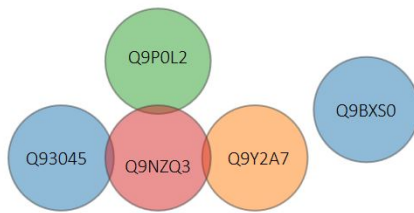


Figure 6: Venn Diagram for G9BXS0.

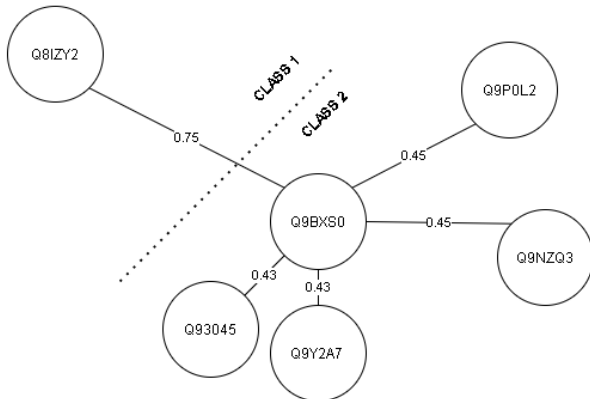


Figure 7: The result of Q9BXS0 expansion by BP dataset.

Table 1
Similarity value for AD and PD.

Measure	BP similarity	MF similarity	CC similarity
Wang	88.3%	91.3%	96.7%
Lin	93%	92%	96.6%

sent interconnection between elements. The prototype was written as a web app by using the SigmaJS framework.

We used two similarity methods, Lin's and Wang's measure, on the three GO vocabularies (*Biological Process, Cel-*

Table 2
Common proteins in AD and PD.

UNIPROT ID	
P03886	NADH-ubiquinone oxidoreductase chain 1
P05067	Amyloid-beta precursor protein
P09936	Ubiquitin carboxyl-terminal hydrolase isozyme L1
P10636	Microtubule-associated protein tau
P25021	Histamine H2 receptor
P37840	Alpha-synuclein
P49754	Vacuolar protein sorting-associated protein 41 homolog
P61026	Ras-related protein Rab-10
P68036	Ubiquitin-conjugating enzyme E2 L3
P78380	Oxidized low-density lipoprotein receptor 1
Q55007	Leucine-rich repeat serine/threonine-protein kinase 2
Q9H4Y5	Glutathione S-transferase omega-2
Q96I20	PRKC apoptosis WT1 regulator protein
Q00535	Cyclin-dependent-like kinase 5
Q13127	RE1-silencing transcription factor
Q13501	Sequestosome-1
Q16143	Beta-synuclein
Q92508	Piezo-type mechanosensitive ion channel component 1
Q92876	Kallikrein-6

ular Component and Molecular Function) for two neurodegenerative diseases, Alzheimer and Parkinson. Thanks to these metrics, we built three different distance matrices (BP, CC, and MF) for each condition.

We explored the differences between the standard cluster view and the proposed DCDG view. The datasets were clustered using the K-means algorithm to show a classic clustering plot. Also, we use the proposed DCDG method to plot the same information into a graph view.

By applying a classic display of clustering, visually was not possible to recover the information immediately, also due to the problem of overlapping of some clusters elements. On

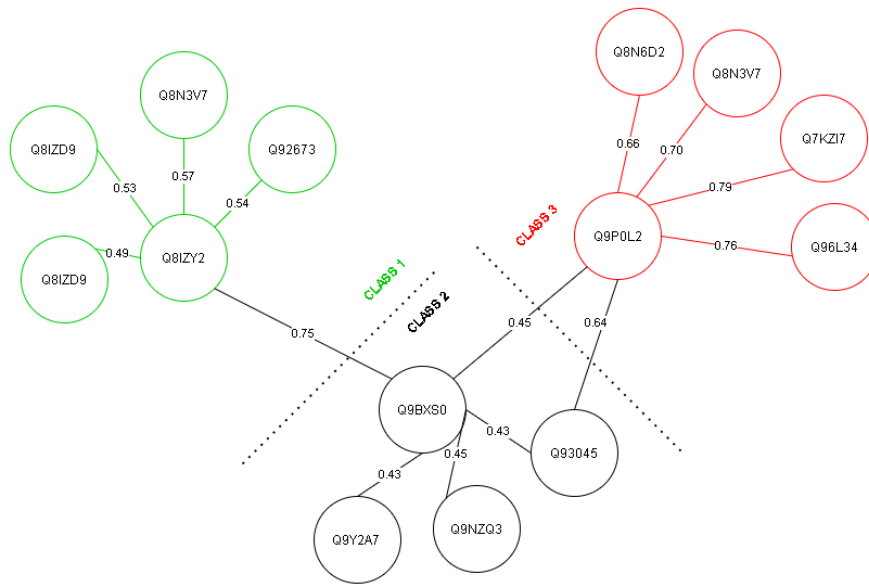


Figure 8: The result of Q8IZY2 and Q9P0L2 expansion by BP dataset.

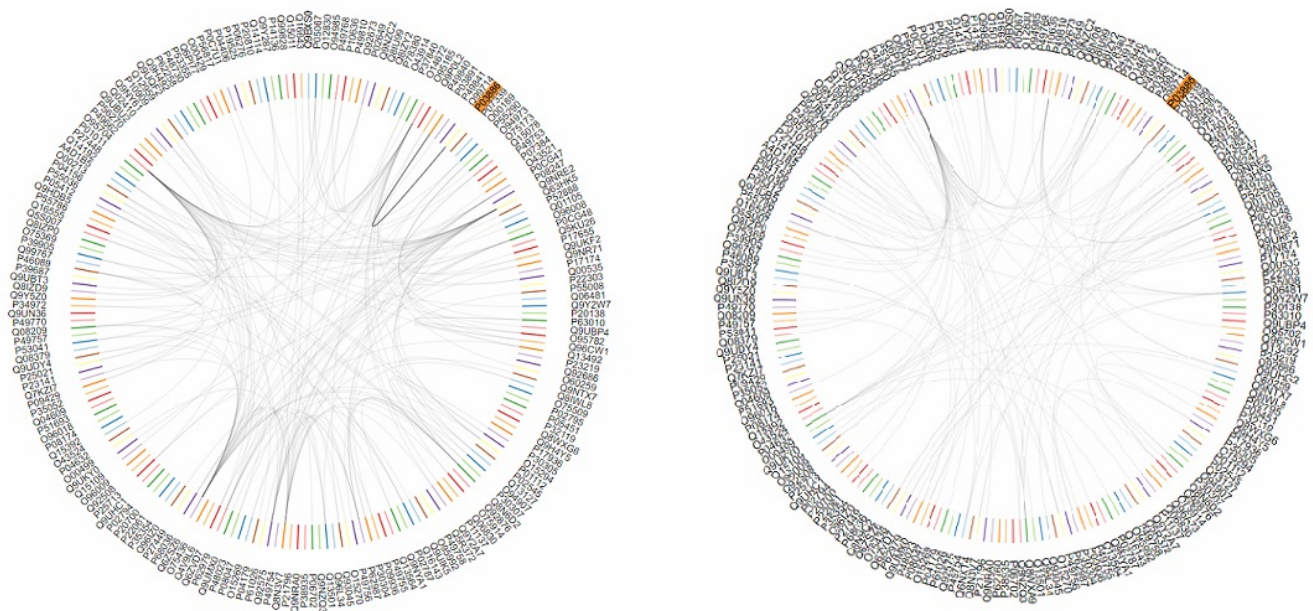


Figure 9: Similarity of BP (on left) and MF (on right) for the protein P03886 in AD.

the other hand, the display with DCDG allows a more immediate understanding of the interactions present between the proteins based on the similarity representative of the three vocabularies of the GO. The existence of well-outed protein clusters in a system is one of the purposes of our work as it represents a fundamental topological characteristic to understand the entire network of connections. This subdivision makes it possible to view the existing relationships between proteins and provides a tool which meets the need to identify and understand why some structural elements are grouped at different levels (cellular, biological and molecular) of in-

depth.

As future work, we plan to improve the web-based tool prototype into a web app with more functionality for the user for exploring protein data based on the proposed assumptions in this research study, guaranteeing user-target customization of the tools available.

References

- [1] Arif, M., 2012. Similarity-dissimilarity plot for visualization of high dimensional data in biomedical pattern classification. *Journal of Medical Systems* 36, 1173–1181.

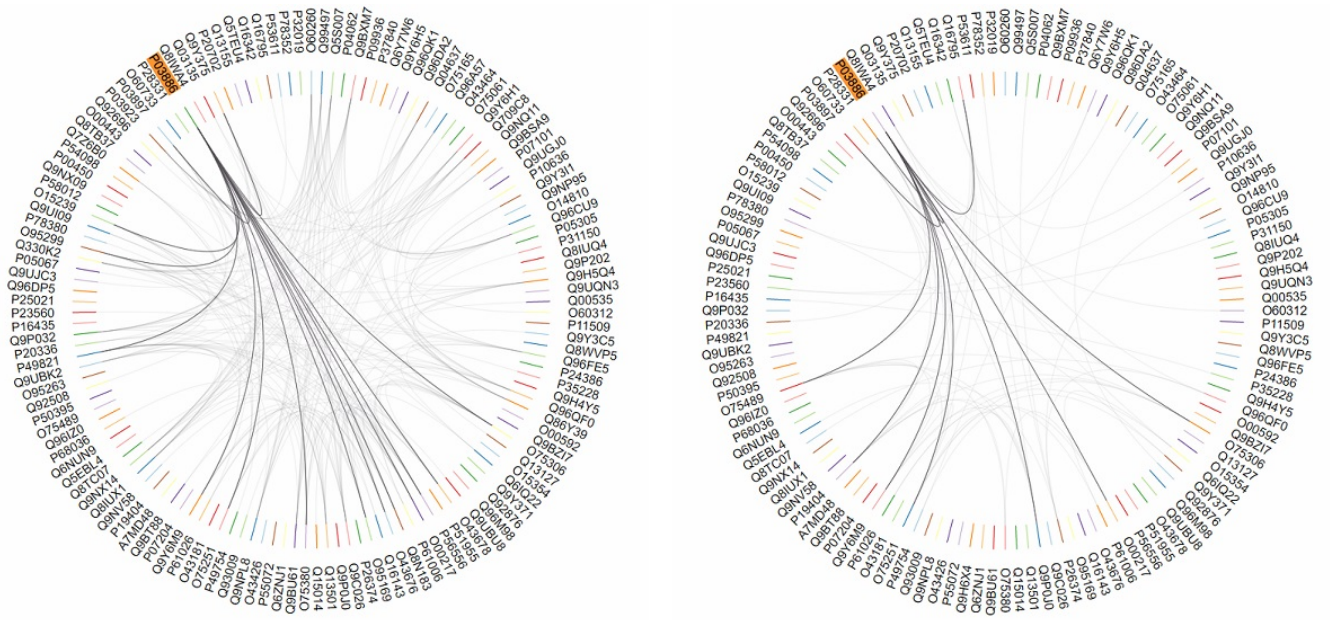


Figure 10: Similarity of BP (on left) and MF (on right) for the protein P03886 in PD.

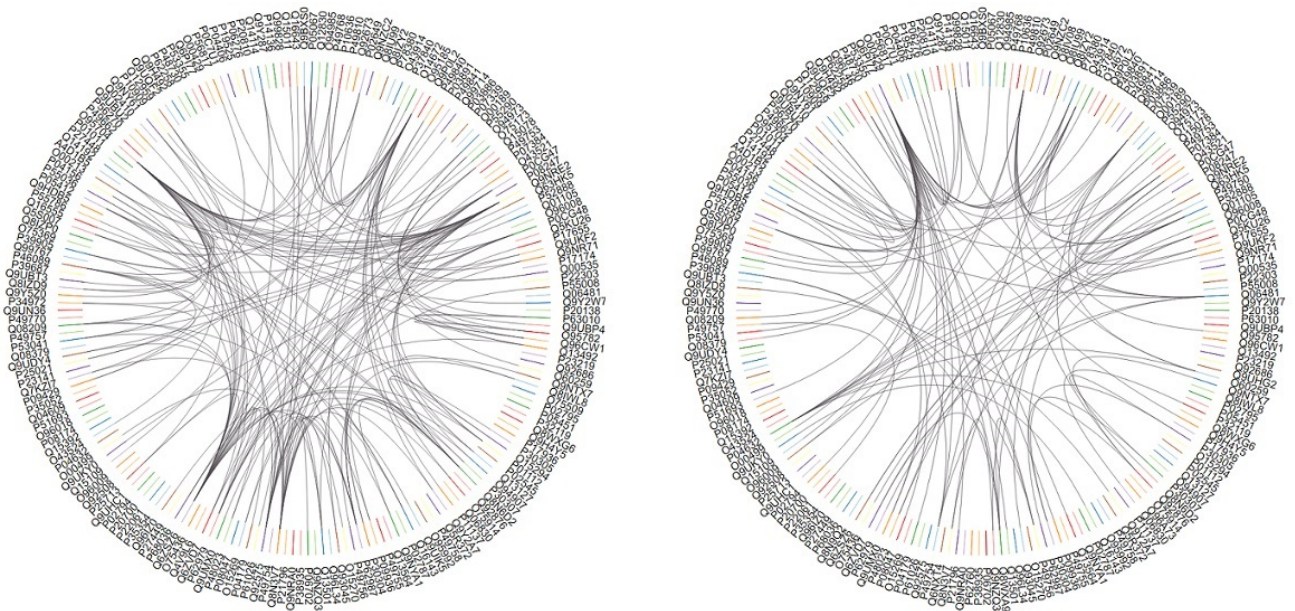


Figure 11: Similarity of BP (on left) and MF (on right) in AD.

[2] Auriemma Citarella, A., De Marco, F., Di Biasi, L., Risi, M., Tortora, G., 2021. Gene ontology terms visualization with dynamic distance-graph and similarity measures, in: 27th International Distributed Multimedia Systems Conference on Visualization and Visual Languages, DMSIVA 2021, Knowledge Systems Institute Graduate School, KSI Research Inc., pp. 85–91.

[3] Binns, D., Dimmer, E., Huntley, R., Barrell, D., O'donovan, C., Apweiler, R., 2009. QuickGO: A web-based tool for gene ontology searching. *Bioinformatics* 25, 3045–3046.

[4] Conesa, A., Götz, S., García-Gómez, J.M., Terol, J., Talón, M., Robles, M., 2005. Blast2GO: A universal tool for annotation, visualization and analysis in functional genomics research. *Bioinformatics* 21, 3674–3676.

[5] Duyckaerts, C., Delatour, B., Potier, M.C., 2009. Classification and basic pathology of Alzheimer disease. *Acta Neuropathologica* 118, 5–36.

[6] Eden, E., Navon, R., Steinfeld, I., Lipson, D., Yakhini, Z., 2009. Gorilla: A tool for discovery and visualization of enriched go terms in ranked gene lists. *BMC Bioinformatics* 10, 1–7.

[7] Fenn, D.J., Porter, M.A., Mucha, P.J., McDonald, M., Williams, S., Johnson, N.F., Jones, N.S., 2012. Dynamical clustering of exchange rates. *Quantitative Finance* 12, 1493–1520.

[8] Gene Ontology Consortium, 2008. The gene ontology project. *Nucleic Acids Research* 36, D440–D444.

[9] Goyal, M., Knackstedt, T., Yan, S., Hassanpour, S., 2020. Artificial intelligence-based image classification for diagnosis of skin cancer: Challenges and opportunities. *Computers in Biology and Medicine*, 104065.

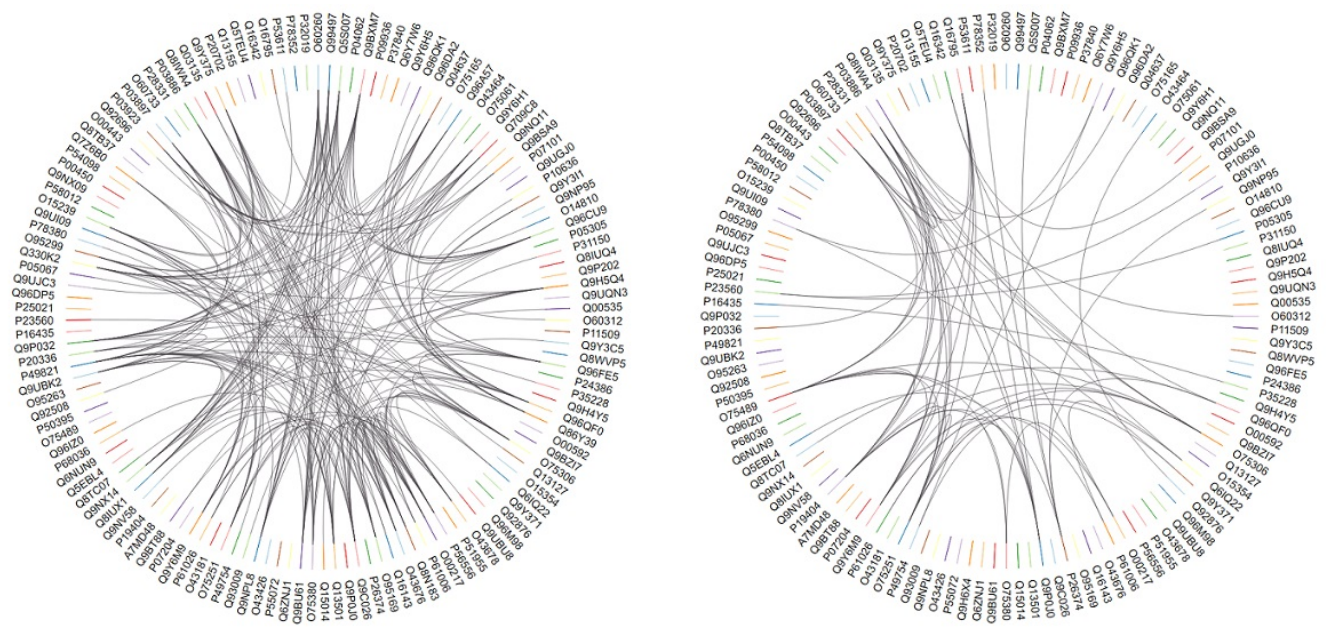


Figure 12: Similarity of BP (on left) and MF (on right) in PD.

[10] Henderson, D.W., 1963. Venn diagrams for more than four classes. *The American Mathematical Monthly* 70, 424–426.

[11] Jacomy, M., Venturini, T., Heymann, S., Bastian, M., 2014. ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software. *PLoS one* 9, e98679.

[12] Lin, D., 1998. Extracting collocations from text corpora, in: *Proceedings of the First Workshop on Computational Terminology*, pp. 57–63.

[13] MacQueen, J., et al., 1967. Some methods for classification and analysis of multivariate observations, in: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297.

[14] O’Donoghue, S.I., Gavin, A.C., Gehlenborg, N., Goodsell, D.S., Hériché, J.K., Nielsen, C.B., North, C., Olson, A.J., Procter, J.B., Shattuck, D.W., et al., 2010. Visualizing biological data—now and in the future. *Nature Methods* 7, S2–S4.

[15] Poewe, W., Seppi, K., Tanner, C.M., Halliday, G.M., Brundin, P., Volkman, J., Schrag, A.E., Lang, A.E., 2017. Parkinson disease. *Nature Reviews Disease Primers* 3, 1–21.

[16] Szklarczyk, D., Morris, J.H., Cook, H., Kuhn, M., Wyder, S., Simonovic, M., Santos, A., Doncheva, N.T., Roth, A., Bork, P., et al., 2016. The STRING database in 2017: Quality-controlled protein–protein association networks, made broadly accessible. *Nucleic Acids Research*, gkw937.

[17] UniProt Consortium, 2015. UniProt: A hub for protein information. *Nucleic Acids Research* 43, D204–D212.

[18] Vailati-Riboni, M., Palombo, V., Loor, J.J., 2017. What are omics sciences?, in: *Periparturient Diseases of Dairy Cows*. Springer, pp. 1–7.

[19] Veenstra, T.D., 2021. Omics in systems biology: Current progress and future outlook. *Proteomics* 21, 2000235.

[20] Wang, J.Z., Du, Z., Payattakool, R., Yu, P.S., Chen, C.F., 2007. A new method to measure the semantic similarity of GO terms. *Bioinformatics* 23, 1274–1281.

[21] Wei, Q., Khan, I.K., Ding, Z., Yerneni, S., Kihara, D., 2017. NaviGO: Interactive tool for visualization and functional similarity and coherence analysis with gene ontology. *Bmc Bioinformatics* 18, 1–13.

[22] Xie, A., Gao, J., Xu, L., Meng, D., 2014. Shared mechanisms of neurodegeneration in Alzheimer’s disease and Parkinson’s disease. *BioMed Research International*.

[23] Yan, J., Risacher, S.L., Shen, L., Saykin, A.J., 2018. Network approaches to systems biology analysis of complex disease: Integrative methods for multi-omics data. *Briefings in Bioinformatics* 19, 1370–1381.

[24] Yu, G., Li, F., Qin, Y., Bo, X., Wu, Y., Wang, S., 2010. GOSemSim: An R package for measuring semantic similarity among GO terms and gene products. *Bioinformatics* 26, 976–978.

[25] Zhu, J., Zhao, Q., Katsevich, E., Sabatti, C., 2019. Exploratory gene ontology analysis with interactive visualization. *Scientific Reports* 9, 1–9.

Use of Natural language inference in optimizing reviews and providing insights to end consumers

Chahat Tandon
Computer Science and Engineering
BIET
Davangere, India
chahat.7876@gmail.com

Pratiksha Jayesh Bongale
Computer Science and
Engineering BIET
Davangere, India
pratikshajb@gmail.com

Arpita T M
Computer Science and Engineering
BIET
Davangere, India
arpita.telkar@gmail.com

Sanjana R R
Computer Science and Engineering
BIET
Davangere, India
sanjanarrdv@gmail.com

Hemant Palivela
Digital Analytics,
eClerx LLC,
Austin, Texas, US
hemant.datascience@gmail.com

Nirmala C R
Computer Science and Engineering
BIET
Davangere, India
nirmala.cr@gmail.com

Article History: Submitted 7.1.2021 Revised 7.31.2021
Accepted 8.20.2021

Keywords: Natural Language Interface, Natural Language Processing, Deep Learning, Amazon Reviews

Abstract: Natural Language Interface or NLI has the potential to add syllogistic reasoning over the already existing facts and develop a new kind of knowledge dataset in itself. In this paper, we have demonstrated a Recognizing Textual Entailment wherein the task is to recognize whether a given hypothesis is true (Entailment), false (Contradiction) or unrelated (neutral) with respect to the sentence called premise. The task is performed by training MNLI corpus along with the manually collected dataset from Amazon Product Reviews each having hypothesis and premise pairs with corresponding labels. With this use case, we propose to bring sustainable development in the classification methods used by major E-commerce companies.

I. INTRODUCTION

In the era of Artificial Intelligence, Deep Learning and Big Data having immense improvements, there is a need to expand Natural Language Processing (NLP) beyond what it does. Hence, expansion of NLP to perform different tasks requires different kinds of datasets and this leads to different types of challenges. One function among those which has lately gained need and popularity is the Natural Language Inference (NLI), also called Recognizing Textual Entailment (RTE). It is the task of defining whether the given hypothesis h is true, false or undetermined with respect to the given premise p . Hypothesis h is also considered as the conclusion c in many of the explanations. A fortunate NLI system is the one that exactly determines whether the hypothesis is entailed, contradicted or neutral to the given premise p . As per the discussion held by Condoravdi et al. (2003)[1] and others, a successful NLI is a suitable computation measure for a real natural language understanding. Goldberg and Hirst (2017)[2] and Nangia et al. (2017)[3] in their discussion clearly noted that solving NLI problems perfectly means to attain human level understanding of language. Hence, a continuous effort is put into designing a high level performing NLI model that has faster learning rate along with massive understanding capabilities.

Every product or service on the internet has a review system to know the opinion of their services from the customers and to help boost the customers loyalty towards their company. Study says that around 86% of the customers consider them as an indispensable resource when selecting the product. Deciding to purchase a product after going through its hundreds of reviews would be a time-consuming task as customers need to go through reviews of different products to find the best one of their choice. How do we reduce time consumption in this review analysis process without human resources? This paper aims at designing a system built using the Natural Language Inference (NLI), a branch of NLP, that will reduce the number of reviews you have to go through for a particular product to find out what needs to be known. The model helps in classifying the reviews into three easy categories namely, entailment, contradiction and neutral. By grouping the common subjects, it becomes convenient for the customer to identify the required review information; thus making it easier for them to make a decision.

There are diverse benchmarks designed for the numerous usecase of Natural Language Inference in different fields of business. A system which analyses the customer reviews is very much required for better functioning of the company and this paper is the first attempt on this use case to the best of our knowledge. The mode is built solely on the NLI aiming at reducing the number of reviews for the customer on the product/services (P/S) in amazon. This paper proposes a review analyzing system with the sole usage of NLI. Given a set of reviews in the paired format, we developed the model in such a way that it determines whether the pair of reviews stands true to each other, false or totally not relatable to one another. e.g *The material of the shirt is super soft. I just love it.* and *I'm inspired by the soft touch of this cloth.* are a pair of reviews on the same product which means the same. *Satisfied with the stitch of the shirt is clean and perfect.* and *Disappointed with the stitching done.* *Threads are coming out.* are another pair of reviews which clearly means opposite to one another. Identifying such pairs of reviews is necessary as we can group the common reviews keeping the rare one.

II. BACKGROUND

Natural Language Processing (NLP) consists of two sub-tasks namely Natural Language Understanding (NLU) and Natural Language Generation (NLG). Together they deal with the understanding of human's languages, processing them, analyzing them in an attempt to understand the semantics of the natural language sentences, and ultimately, generating an output back in human's language as was sent as input so that it is interpretable by humans effortlessly. NLP bundles a broad variety of use-cases, wherein some are considered to be easy tasks and the others a complex one to deal with that include recognition of entities [4] such as names of places, persons, etc., within texts, Sentiment Analysis [5], Machine Translation [6] of languages, Machine Reading Comprehension (MRC) [7], etc. Natural Language Processing requires an important task within itself, called Natural Language Inference (NLI). It is the task of appropriately inferring one sentence from another and classifying any two sentences (Premise and Hypothesis) in three categories namely, *entailment*, *contradiction*, and *neutral*. In some cases, some known facts and prior knowledge about the topic is taken into account while classifying the text pair. For instance, most Hindi speakers would know how to acknowledge a 'Namaste' or 'Kya haal hai?', etc.

Multi-Genre Natural Language Inference (MNLI) [8], a larger corpus compared to SNLI roofs ten distinct genres of the English Language. It houses about 433k p-h (premise-hypothesis) pairs making it a good choice for NLI over English language. The test set has two categories namely, *matched set* - sentences with indistinguishable genres and *mismatched set* - sentences with distinguishable genres, thus, facilitating cross-genre language inference.

III. RELATED WORK

Natural Language Inference is best dealt by applying Deep Learning based methods. We know for a fact that Deep Learning based methods require a humongous amount of training data to be able to learn the language representation and produce desired results. To acknowledge the massive need, many researchers, crowd workers, and others, came forward and created numerous datasets for NLI purposes. Hossein Amirkhani et al. [9] created the largest NLI dataset called FarsTail, entirely dedicated to the Persian language. The dataset consists of 10,367 examples that are given in both Persian language and an indexed-format to be beneficial for non-Persian experimenters. The Premise-Hypothesis pairs were created utilizing about 3500 multiple-choice queries with no or a little involvement of annotators with annotating the pairs. They investigated several techniques ranging from the traditional ones to the state-of-the-art methods bundling word embedding methods including word2vec [10], fastText [11], ELMo [12], BERT [13], and LASER [14], various modeling approaches specifically dedicated to Natural Language Inference dealing, such as, DecompAtt [15], ESIM [16], HBMP [17], ULMFiT [18], and cross-lingual transfer approaches. Another research conducted by Mohammad Mosharaf

Hossain et al. [19] dealt with countering negation within English sentences as they are ubiquitous in common English sentences. Their study reveals that current datasets including the Stanford Natural Language Inference (SNLI) dataset, Recognizing Textual Entailment (RTE) dataset do not address negative words within sentences and which makes state-of-the-art transformers poor at handling words carrying negative meanings such as, no, nothing, never, not, isn't, haven't, hasn't, so on and so forth. The transformers tend to neglect the negative words or phrases and proceed with the inference task of classifying sentences into their respective classes.

In today's evolving world, code-mixing is prevalent. *Code-mixing* refers to the mixing of two or more languages while conserving with each other, for instance, "Hey buddy! Haven't seen you since the first week of December! Kya chal raha hai?". This phrase consists of two languages namely, English and Hindi; this is how most people talk these days on social media platforms like WhatsApp, Instagram, Facebook, etc. Simran Khanuja et al. [20] went on and created a whole new dataset that takes into account the mixing of different languages. It consists of 400 code-mixed (English-Hindi) premises taken from 18 Bollywood movie scripts for which 2240 hypotheses are in the same format as premises, i.e., code-mixed.

Currently, there are many datasets that aid NLI with not just English but many more languages such as Hindi, Turkish, Spanish, German, Thai, Chinese, etc. There are some datasets that include premises and hypotheses just in English, for instance, SNLI; and some that include other languages such as French, Spanish, Greek, German, Swahili, Urdu, Arabic, etc., for instance, XNLI. The English-dedicated datasets include, *SICK (Sentences Involving Compositional Knowledge)* [21] which is one of the initial trials towards building larger datasets to support NLI functions. It bundles around 10,000 premise-hypothesis pairs in English language. The dataset was annotated for dual purposes, one, to determine correlation between sentences and two, entailment. The initial dataset consists of irregularly picked from about 8,000 ImageFlickr dataset along with the SemEval 2012 STS MSR-Video Description dataset. A few rule-grounded lexical and syntactic transformations have been put into every sequence of words to generate appropriate classification (entailment, contradiction, or neutral). Another well known corpus down the line is the *Stanford Natural Language Inference (SNLI)* [22] that tackles the need of huge annotated data for the NLI task to be solved using Deep Learning architectures. The corpus consists of around 5,70,000 labeled premise-hypothesis pairs out of which the training set consists of 550k samples, validation set of 10k, and test set of 10k examples. The entire corpus was collected via the Amazon Mechanical Turk. Every premise was asked to be paired with three different hypotheses, one entailment, one contradiction, and a neutral one. *MedNLI* [23], as the name suggests, is dedicated to the medicinal domain. An addition to the English dominated corpora is *SciTail*. *SciTail* [24] consists of science questions and answers as hypotheses and relevant word sequences from the web were taken as premises. A total of 1,834 queries taken together with about 16k neutral examples and around 10k entailment examples

form the entire corpus. SciTail lacks the third label within NLI, the *contradiction* tag. *QA-NLI* [25], an automatically generated corpus built by leveraging the Question Answering datasets like the *SQuAD 2.0* [26]. The dataset followed the following pattern all over: correct answer - entailment, incorrect answer - contradiction, and unknown answer - neutral.

There are many Non-English corpora that have been designed to acknowledge NLI tasks in languages other than English including Evalita, ArbTEDS, German emails, etc. The Italian dataset, *Evalita* [27], consists of 800 short Italian text pairs constituted using the Wikipedia articles. An Arabic language corpus, *ArbTEDS* [28], containing 600 annotated text pairs involving both inferable and non-inferable. Candidate duplets are taken from the web leveraging a nearly automatic instrument, with news headlines in Arabic as hypotheses and a passage rendered by Google's API for the just taken headline as the corresponding premise, annotated by eight annotators. *German emails* [29], a corpora built by emails sent by customers of a multimedia software company to its support hub as premises and the different class description as hypotheses. The matching classes relate to the entailment category (around 600) and non-matching classes map to the non-entailment category (around 21k). *ASSIN* [30], mixture of 10k couples, having both European Portuguese and Brazilian Portuguese pertaining to two different classes, namely, entailment and non-entailment. The massive *cross-lingual* dataset, XNLI which stands for *Cross-lingual Natural Language Inference*, the MultiNLI text pairs that were collected in a crowd-sourced manner, these pairs were then translated into 14 distinct languages by experts.

IV. DATA COLLECTION

A. Dataset and Task to be performed:

We handpicked 500 reviews, including one star to five-star reviews in order to give our data some uniformity. For instance, the review dataset for a particular shirt contained reviews ranging from its color, size, fabric quality to the fitting and the thread count. The dataset contained both positive as well as negative reviews. We then bifurcated the obtained reviews in three categories; entailment (0), neutral (1) and contradiction (2). Four columns were obtained containing the column id, the hypothesis, premise and the label. Consider the following review examples used for labelling the sentences:

For a book review: "The book cover is beautiful!" and "Beauty lies in the cover as well as the content of the author." can be labelled as **entailment (0)**.

For a phone review: "I bought this phone for my father, and he liked it a lot, fast charge, good battery life" and the "screen width is small" can be labelled as **neutral (1)**.

For a laptop review: "Bluetooth connection problem" and "Bluetooth connectivity is awesome but the battery drained fast" can be labelled as a **contradiction (2)**.

Id	premise	hypothesis	label
1	Good product but price is too high for tht item all over a good item	Bad product. After washing all the prints are gone .. don't buy .. even sweat patches are removing the print on shirt	2
2	Superb, awesome, color also not faded	The color is the most appropriate one i wanted	1
3	Good product	Worst product in amazon	2
4	The product which was provided was good and it was 100 percent cotton	Cloth is not worth the money	1
5	Good fabric	Best cloth material	0
6	Great quality for this greatprice	Excellent product quality. Go ahead and purchase	0

Fig. 1 - A snippet of training dataset

B. Input and Output:

Each dataset contains two sentences (a premise and a hypothesis) and a labeling class that indicates if the sentences describe the same thing (entailment), disagree with one another (contradiction) or talk about different things (neutral). So the model needs to take in two inputs (the two sentences) and return one of the three classes. For the output part, we feed in a submission xls file that saves the predictions made. A snippet of submission file has been shown below:

Id	Predictions	Actual Predictions
1	2	2
2	1	1
3	2	2
4	2	1
5	0	0
6	0	0
7	1	1
8	0	0
9	0	1
10	2	2

Fig 2- A snippet of Submission.xls

V. METHOD

A. XLM-RoBERTa:

Model used for this use case is XLM-RoBERTa. XLM-RoBERTa is based on RoBERTa which was proposed in 2019. This multilingual model is trained on filtered Common Crawl data across 100 different languages. The crawl data corpus is a collection of data in petabytes collected over 8 years of web crawling. In specific, we used xlm-roberta-large configuration of XML-RoBERTa. There are 24 hidden layers, 16 attention heads and 1024 hidden units.

We used TPUs in training. TPU short for *Tensor Processing Unit*, are application specific integrated circuits used to accelerate the workloads in machine learning. Being different from GPUs, a TPU needs to be initiated and set up to carry out work with the specified model in the notebook. Explicitly, a "strategy" needs to be demarcated regarding the working of the model and how it can be replicated across the eight GPU chips on the TPU board. The later part of these replica models being merged back together also needs to be taken care of once training has completed.

B. Process Flow

After setting the max length to 80; the batch size needed to be multiplied by the number of replicas (8).

This made sure that each of the eight GPU chips in the TPU was made to use the specified batch size and not one eighth of that number. The learning rate was set to $1e-5$. The train and test set; each containing four columns namely, Id, hypothesis, premise and label was loaded. Augmentation of dataset helps increase diversity in the dataset all the while increasing accuracy. For this purpose, we used the Multi-Genre NLI Corpus dataset. The dataset was found to include $392702 \text{ rows} \times 3 \text{ columns}$. This proved to have added an advantage while classifying our dataset; thus improving the model's overall performance. The proposed system can be described by a process flow shown below:

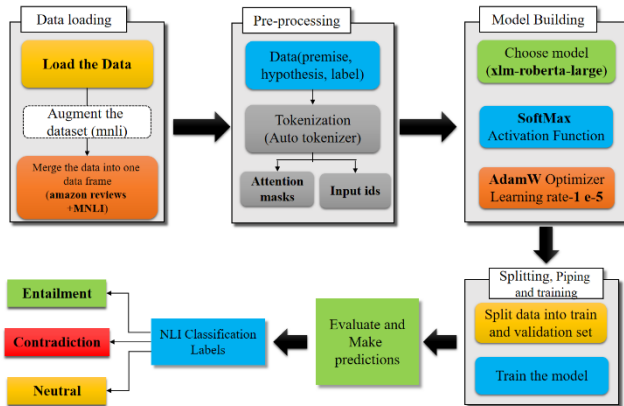


Fig 3- Process flow of the proposed model

C. Training Phase

In order to start with the training phase, we first **concatenated** our collected dataset with that of MNLi. In order for a machine to understand human-language, words need to be encoded and fed as integer inputs. This is known as **encoding**. In order to prepare sentences to be fed as input for training, the text is tokenized i.e. assigned numbers(tokens). Each model has its own unique set of tokens. Every sentence from the dataset is then converted from strings to arrays of tokens using Auto Tokenizer class from Transformers. The tokenizer even separates the words themselves into sub words. For instance, a word “bookmark” will be split as “book” and “mark” subwords. In our case, the premise and hypothesis both act as input and so the tokens from both will be merged into one array.

D. Split and Pipeline

The training set was split into 80% training and 20% validation set with a random state of 2020. Data Pipeline is used to exact every ounce of performance from the model at hand. We used Tensorflow data API to create a data pipeline in order to increase the performance while training. Commands like shuffling, slicing, prefetching the next batch, etc can be performed easily because of this pipeline. Lastly, the RoBERTa was added as the backbone of our model along with a softmax function layer in order to apply the correct class (entailment, neutral, contradiction or 0, 1, 2).

E. Evaluations and Predictions

The model is trained for 5 epochs with varying datasets each time and values for Training loss/ Accuracy as well as Validation loss/ Validation accuracy are noted. The new obtained predictions are saved in submissions.xls as output and can be later compared with the actual dataset. The below graphs showcase the evaluations obtained for the shirt dataset.

After the evaluation of individual datasets, a mixed dataset was created manually to check how the model would perform. It consisted partly of reviews of all the four products hand picked randomly. The model was deemed to have been trained properly as the validation accuracy obtained was 86%. Thus, we can conclude that the model seems to be ready to be used for **real-life business cases**.

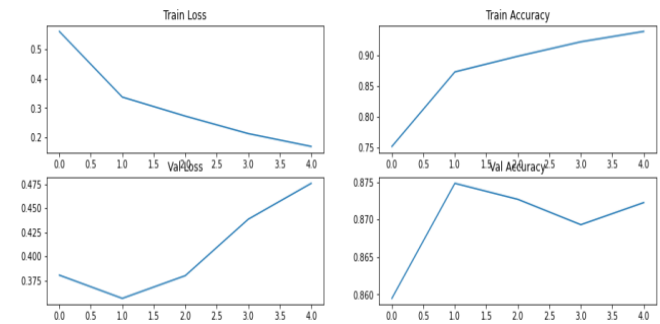


Fig 4- Evaluations obtained on shirt dataset.

VI. COMPARISON OF PREDICTIONS

The reviews collected for our dataset were of different products. 4 different products were chosen from amazon and all its reviews were collected. The first product we chose was the men's yellow shirt. Reviews for the same spoke about the material quality, buttons hook up, stitching done and various other features of the shirt. Regardless of whether the customer liked or disliked, all the reviews were selected to prepare our dataset. The NLI model minimized the number of similar reviews which would consume huge time in reading all of them. This minimization will not only help the customers in choosing their best product based on reviews easily and more sprightly, but will also help the sellers to look for the negative reviews for their product easily. This will assist the seller in improvising their shirt considering the negative reviews of the customer.

Similarly, the other products chosen were the crime and punishment book, Samsung M11 smartphone, Dell and HP laptops from amazon e-commerce website. All the possible reviews notwithstanding to positive and negative ones were collected for the same. Once the number of reviews were minimized by the NLI model by eliminating the reviews holding the same meaning, there remains fewer reviews making it easy for the customer to decide on purchasing their product quicker saving a huge amount of time for them. The creation of our model does not restrict to helping only the customers of the e-commerce website, it also makes things easier for the sellers online to understand customer needs and act faster and accordingly on it as the same in the above described shirt product. Any product chosen and review collected on the same, our NLI model designed for the reviews minimization eliminates one review from the one's, those are entailing. This system is found useful to all

customers, sellers and manufacturers. Due to this magnificent feature of our model, this is considered to be a very good usecase in business. Such a type of model design is the first attempt in the field of NLP to the best of our knowledge.

A table showing Training/Validation and Testing accuracy on the aforementioned product review datasets has been showcased below. It is evident that the model was trained on a diverse variety of data and so was able to achieve great accuracy scores even when tested on different products. Thus this experiment can be deemed to be successful and useful for today's business cases.

Table 1– Training/Validation and Testing Accuracy

Reviews	Sets	Accuracy		
		Train	Validation	Test
Shirts	1-100	93.98	87.23	70
Book	101-200	94.18	87.39	60
Mobile	201-300	93.71	87.22	70
Laptop	301-400	94.04	87.1	79
	401-500	93.77	87.11	61

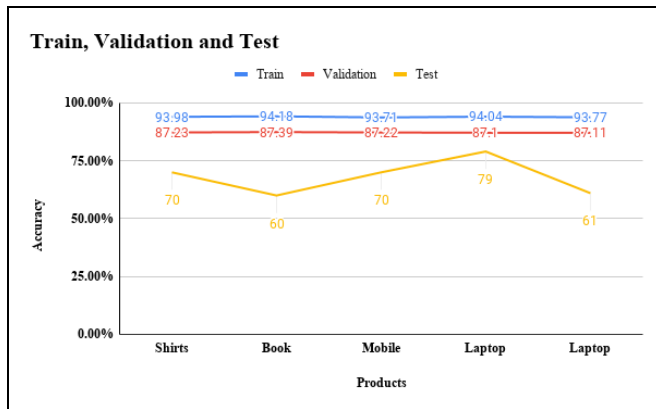


Fig 5 - Train, Validation and Test Accuracy on all the Products

VII. CONCLUSION

In this paper, we proposed the use of Natural Language Interface with respect to real life major E-commerce companies. We presented experiments on MNLI corpus along with manually obtained Amazon review dataset designed with 500 samples each having hypothesis and premise pairs with corresponding labels. The task of classifying hypotheses and premises in labels, i.e. true (Entailment), false (Contradiction) or unrelated (neutral) was done with 87% accuracy overall. The proposed model captures the need and possibilities with the use of NLI on a huge scale. This model caters to the need of classification strategy required in today's world of review driven sales. We hope that this model could be a stepping stone for future advancements in retail as well as general use cases.

REFERENCES

[1] Cleo Condoravdi, Dick Crouch, Valeria De Paiva, Reinhard Stolle, and Daniel G Bobrow. 2003. Entailment, intensionality and text understanding. In Proceedings of the HLT-NAACL 2003 workshop on Text meaning-Volume 9, pages 38–45. Association for Computational Linguistics.

[2] Yoav Goldberg and Graeme Hirst. 2017. Neural Network Methods in Natural Language Processing. Morgan & Claypool Publishers.

[3] Nikita Nangia, Adina Williams, Angeliki Lazaridou, and Samuel Bowman. 2017. The RepEval 2017 Shared Task: Multi-Genre Natural Language Inference with Sentence Representations. In Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP, pages 1–10, Copenhagen, Denmark. Association for Computational Linguistics

[4] V. Yadav, S. Bethard, A survey on recent advances in named entity recognition from deep learning models, in: Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 2145–2158.

[5] A. Keramatfar, H. Amirkhani, Bibliometrics of sentiment analysis literature, Journal of Information Science 45 (1) (2019) 3–15.

[6] S. Yang, Y. Wang, X. Chu, A survey of deep learning techniques for neural machine translation (2020). arXiv:2002.07526.

[7] R. Baradaran, R. Ghiasi, H. Amirkhani, A survey on machine reading comprehension systems (2020). arXiv:2001.01582.

[8] A. Williams, N. Nangia, S. R. Bowman, A broad-coverage challenge corpus for sentence understanding through inference, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018, pp. 1112–1122.

[9] Hossein Amirkhani, Mohammad Azari Jafari, Azadeh Amirak, Zohreh Pourjafari, Soroush Faridan Jahromi, Zeinab Kouhkan, FarsTail: A Persian Natural Language Inference Dataset, arXiv:2009.08820v1 [cs.CL] 18 Sep 2020.

[10] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in neural information processing systems, 2013, pp. 3111–3119.

[11] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics 5 (2017) 135–146.

[12] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018, pp. 2227–2237.

[13] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, 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, Volume 1 (Long and Short Papers), 2019, pp. 4171–4186.

[14] M. Artetxe, H. Schwenk, Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond, Transactions of the Association for Computational Linguistics 7 (2019) 597–610.

[15] A. P. Parikh, O. Tackstrom, D. Das, J. Uszkoreit, A decomposable attention model for natural language inference, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 2249–2255.

[16] Q. Chen, X. Zhu, Z. Ling, S. Wei, H. Jiang, D. Inkpen, Enhanced LSTM for natural language inference, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2017, pp. 1657–1668.

[17] A. Talman, A. Yli-Jyra, J. Tiedemann, Sentence embeddings in NLI with iterative refinement encoders, Natural Language Engineering 25 (4) (2019) 467–482.

[18] J. Howard, S. Ruder, Universal language model fine-tuning for text classification, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2018, pp. 328–339.

[19] Mohammad Mosharaf Hossain, Venelin Kovatchev, Pranoy Dutta, Tiffany Kao, Elizabeth Wei, Eduardo Blanco, An Analysis of Natural Language Inference Benchmarks through the Lens of Negation.

[20] Simran Khanuja, Sandipan dandapat, Sunayana Sitaram, Monojit Choudhury, A New Dataset for Natural Language Inference from Code-mixed Conversations, arXiv:2004.05051v2 [cs.CL] 13 Apr 2020.

[21] M. Marelli, L. Bentivogli, M. Baroni, R. Bernardi, S. Menini, R. Zamparelli, SemEval-2014 Task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment, in: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 2014, pp. 1–8.

[22] S. R. Bowman, G. Angeli, C. Potts, C. D. Manning, A large annotated corpus for learning natural language inference, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 632–642.

- [23] A. Romanov, C. Shivade, Lessons from natural language inference in the clinical domain, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 1586–1596.
- [24] T. Khot, A. Sabharwal, P. Clark, SciTail: A textual entailment dataset from science question answering, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018, pp. 5189–5197.
- [25] D. Demszky, K. Guu, P. Liang, Transforming question answering datasets into natural language inference datasets (2018). arXiv:1809.02922.
- [26] P. Rajpurkar, R. Jia, P. Liang, Know what you don't know: Unanswerable questions for SQuAD, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2018, pp. 784–789.
- [27] J. Bos, F. M. Zanzotto, M. Pennacchiotti, Textual entailment at evalita 2009, Proceedings of EVALITA 2009 2 (6.4) (2009) 1–7.
- [28] M. Alabbas, A dataset for Arabic textual entailment, in: Proceedings of the Student Research Workshop associated with RANLP 2013, 2013, pp. 7–13.
- [29] K. Eichler, A. Gabryszak, G. Neumann, An analysis of textual inference in German customer emails, in: Proceedings of the Third Joint Conference on Lexical and Computational Semantics (*SEM 2014), 2014, pp. 69–74.
- [30] E. R. Fonseca, L. Borges dos Santos, M. Criscuolo, S. M. Alu'isio, Overview of the evaluation of semantic similarity and textual inference, Linguamatica 8 (2) (2016) 3–13.
- [31] A. Conneau, G. Lample, R. Rinott, A. Williams, S. R. Bowman, H. Schwenk, V. Stoyanov, XNLI: Evaluating cross-lingual sentence representations, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 2475–2485.

Journal of Visual Language and Computing

journal homepage: www.ksiresearch.org/jvlc/

Long-Term Predictions of Bike-Sharing Stations' Bikes Availability

Enrico Collini^a, Paolo Nesi^{a,*} and Gianni Pantaleo^a

^a*Distributed Systems and Internet Technologies Lab, Department of Information Engineering, University of Florence, Florence, Italy, <https://www.disit.org>, <https://www.snap4city.org>*

ARTICLE INFO

Article History:

Submitted 3.1.2021

Revised 6.1.2021

Second Revision 8.1.2021

Accepted 8.15.2021

Keywords:

Available bikes prediction

Bike-sharing

Machine learning

Prediction models

Smart city

ABSTRACT

Bike-sharing systems are present in many cities as a valid alternative to fuel-based public transports since they are eco-friendly, prevent traffic congestions, reduce the probability of social contacts. On the other hand, bike-sharing present some problems such as the irregular distribution of bikes on the stations/racks/areas (still very used for e-bikes) and for the final users the difficulty of knowing in advance their status with a certain degree of confidence, whether there will be available bikes at a specific bike-station at a certain time of the day, or a free slot for leaving the rented bike. Therefore, providing predictions can be useful for improving the quality of e-bike services. This paper presents a technique to predict the number of available bikes and free bike slots in bike-sharing stations (the best solution for e-bikes). To this end, a set of features and predictive models have been developed and compared to identify the best prediction model for long-term predictions (24 hours in advance). The solution and its validation have been performed by using data collected in bike stations in the cities of Siena and Pisa, in the context of Sii-Mobility National Research Project on Mobility and Transport and Snap4City Smart City IoT infrastructure. The Random Forest (RF) and Gradient Boosting Machine (GBM) offer a robust approach for the implementation of reliable and fast predictions of available bikes in terms of flexibility and robustness to critical cases, producing long-term predictions in critical conditions (i.e., when there are only few remaining available bikes on the rack).

© 2021 KSI Research

1. Introduction

Today, about 55% of the world's population lives in urban areas, and this figure is expected to reach 68% in 2050, according to the "World Urbanization Prospects 2018", published by the United Nations Department of Economics and Social Affairs [17]. Fuel-based transportations are one of the most important causes of certain gas emissions and thus of air pollution. Bike-sharing systems may represent a part of the solution. Therefore, their use is increasing in many cities, being a more sustainable alternative to public transportation reducing congestion and pollution. The bike-sharing solution adopting bike rack stations are capable to

detect the presence of the bike, to assess their status, to recharge e-bikes, and release/manage the bike-sharing system. In this case, the bikes can be very simple even when they are e-bikes. The alternative solution could be floating bike-sharing systems in which the users can take the bikes from the road and leave them in any place, in some cases with specific rules and areas. The bikes have to be more intelligent, and capable to communicate with the central management servers their position, etc., such as Mobike solution. Floating solutions are still not very effective in the case of e-bike since the recharging can be easier on racks. The recharging of floating bikes has to be performed by collecting bikes and/or by recovering them to put a charged battery. All these activities are very expensive to be performed for a large number of bikes.

In the context of this article, the solution with simple bikes (even e-bike) and smarter stations is addressed. The bikes can be typically released at any station providing that a free slot is available, this may create

*Corresponding author

enrico.collini@unifi.it (E. Collini); paolo.nesi@unifi.it (P. Nesi);

gianni.pantaleo@unifi.it (G. Pantaleo)

ORCID(s): 0000-0002-1304-5545 (E. Collini); 0000-0003-1044-3107 (P. Nesi); 0000-0002-9235-437X (G. Pantaleo)

discomfort to the users when the station is full, and the user has to search for an empty slot in near bike racks to leave the bike, and then return by walk. One of the problems of bike-sharing is related to the irregular distribution of bikes among the various stations and the impossibility to know with a certain confidence where to find at least a bike at the desired station in a precise time slot of the day, or just few minutes in advance. The same for the possibility to find a free slot to leave the bike. Therefore, predicting the availability of bikes (as well as to predict the presence of free slots) per station over time can be very useful for managing the demands for bikes per station and to plan/schedule a bike redistribution [2].

1.1 Related Works

In recent years, many researchers have studied urban bike-sharing systems, mainly on four main areas of interest.

The first area is the *design of Bike-Sharing Systems*. In [3], a mathematical model has been proposed to determine the number of docking stations needed, their location and the possible structure of the cycle path network, as well as models to make predictions about possible routes taken by users between stations of origin and destination.

The second area is related to the *analysis of the behavior and dynamics of a Bike-Sharing system*. In [4] and [5], clustering and forecasting techniques are used on the network of Bike-Sharing stations in Barcelona to obtain useful information to describe the city's mobility. In [6], the authors studied the Vélo' system. They interpreted the system as a dynamic network by analysing how bicycle flows distribute spatially along the network. In [8], clustering techniques are used to analyse the Vienna docking station network. In [7], different Bike-Sharing services are analysed highlighting the differences in bike flows and routes.

The third area is referring to the *redistribution of bicycles among stations of the city* that is necessary to compensate for the imbalance created during their use. For example, in [18], [19], [20], the authors studied the optimization of the routes taken by vehicles with the aim of balancing the number of bicycles in each station.

The last area concerns the *prediction of bikes availability*.

In [4], four different predictive models for estimating the availability of bikes in stations have been compared. The authors use a Bayesian network to predict the status of a bike station (full, almost empty or empty) using bicycle parking information only 2 hours in advance. They achieve a forecast accuracy of about 80%. In [5], ARMA models have been used to predict the number of vacancies one hour in advance, while in [1], the authors present a system for predicting bike traffic of a bike-sharing network in Lyon. In [21], data mining and cluster techniques based on historical data series are used to estimate pickup and return activity patterns at bike stations in Vienna; while in [22], the authors presented an ARIMA model that takes into account

both spatial and temporal factors to predict the number of available seats in each docker station.

In most cases, the prediction algorithm aimed at understanding the total number of used bikes in the whole network over time, which is a topic of interest for the operator. This is also much simpler than the prediction of the bike or slot on single rack.

In [23], the authors presented a predictive model of the state of the public bike-sharing stations in Barcelona 2 days in advance. Thus, the Random Forest has been applied to predict the status of a station (i.e., when a station is full, almost full, if there are slots and bikes available, almost empty or empty, two days in advance) with a maximum accuracy of about 75%. The authors also consider in the model some external factors as holidays, and weather information observing that the inclusion of these external factors was not relevant. In [24] and [25], a probabilistic approach based on dynamics modelling of a single bicycle parking using Markov chains in continuous time has been proposed. In [24], the authors predict the number of available bikes per bike station in Paris with an error measure of about 3.5 in terms of RMSE (which is very high for small size racks), for a prediction horizon of one hour in slots of 10 minutes. In [25], The authors use statistical methods to model the spatio-temporal shifts of bikes between stations, and then estimate bike check-in results based on the model and online check-out records. A random forest-based prediction mechanism is further proposed to model and forecast the users' check-out behaviours. In [26], an approach based on Graph Convolutional Neural Network has been used to analyse the dynamics between the different bike-sharing stations in the city. In [27], a deep learning model for short-term prediction of the number of available bikes is presented. In [27], the authors adopt LSTM and GRU models to predict the number of bikes per docker station 1, 5 and 10 minutes in advance with one-month historical data, and they apply a Random Forest as a benchmark. In [28], the authors present an approach based on the application of machine learning models as Random Forest and LSBoost algorithms to create univariate models to predict the number of available bikes at each of the 70 stations of the Bay Area Bike Share network. RF with a MAE of 0.37 outperformed LSBoost with a MAE of 0.58 bikes/station with a prediction horizon of 15 minutes. The authors also apply a Partial Least-Squares Regression to model available bikes at the spatially correlated stations of each region obtained from the trip's adjacency matrix. Results show that the MAE was approximately 0.6 bikes. Finally, in [29], the authors propose a framework based on recurrent neural networks to predict bike demand for each station in a bike-sharing system one hour in advance. Table 1 shows a comparative overview of the most relevant related works.

Table 1: Related Work implementation overview

Paper	Models Type	Features	Pred. Horizon	Accuracy/Error Measures
[4]	Bayesian network (station status: full; almost empty; empty)	Holiday, Weather and Historical data	2 hours	Accuracy about 80%
[23]	Random Forest (RF) (station status: full; almost full; bikes available; almost empty; empty)	Weather and Historical data	2 days	Accuracy about 70%
[24]	Markov chains (#available bikes)	Historical data	1 hour	RMSE about 3.5
[27]	- GRU - LSTM - RF (#available bikes)	Historical data	3-time intervals: 1, 5, 10 min	-
[28]	- RF - LSBoost - PLSR (#available bikes)	Historical data	From 15 min to 120 min	MAE (RF) about 0.37 for 15 min pred. horizon
[29]	RNNs (#check-in/check-out)	Weather and Historical data	1 hour	MAE about 1.2

1.2 Article Overview

The **main contribution of this paper** consists in presenting a solution for long-term prediction of available bikes on bike-sharing stations, and thus of the number of free slots by knowing the size of the station and the number of broken bikes. To this aim, a model has been identified to predict the availability of bikes 24 hours in advance (long-term predictions) with a resolution of 15 minutes, and thus also the free slots in the stations. The prediction of available bikes is a non-linear process whose dynamic changes involve multiple kinds of factors, coming from the context. To this end, the solution has been obtained by taking into account different cities and locations, and despite the differences characterizing the two cities (namely Siena and Pisa), in both cases the identified features and model have been the same, thus demonstrating the validity of the derived results. The precision obtained for long terms prediction has been much better than those provided in the literature.

The solutions have been implemented in the context of Sii-Mobility project (national mobility and transport smart city project of Italian Ministry of Research for terrestrial mobility and transport, <http://www.sii-mobility.org>) and Snap4City infrastructure (<https://www.km4city.org>) [9], [10], [11], which in turn is based on Km4City model. Sii-Mobility aimed at defining solutions for sustainable mobility, engaging

city users, providing predictions on parking, suggesting bikes availability status to users at least 15 minutes/1 hour in advance to allow them to make a conscious decision, and maybe change their own plan. As a result, the solution has been capable to produce reliable prediction even 24 hours in advance.

The paper is structured as follows. Section 2 provides a description of the bike-sharing data and their characterization in terms of clustering in groups. In addition, the identification of several features at the basis of the predictive models is reported. In Section 3, the machine learning approaches adopted to identify and validate the predictive models and framework are presented. Section 3.1 presents the metrics for the assessment, Section 3.2 the ARIMA model. In Section 3.3, the machine learning approaches are presented. A computational cost analysis on the proposed solutions is presented in Section 3.4. The feature relevance of the predictive model is discussed in Section 3.5 and Section 3.6 reports the results based on a feature reduction analysis. Conclusions are drawn in Section 4.

2. Data Description And Feature Identification

As mentioned in the introduction, the main goal was to find a solution to predict the number of bikes available in each bike station 24 hours in advance. Thus, by knowing the size of the bike station and the number of broken bikes on the rack, we can derive the number of free slots to leave the rented bike. Typically, the status of each bike station is checked and registered on the central server every 15 minutes. The data we adopted refer to 15 stations located in the municipality of Siena and 24 stations located in Pisa. In order to understand the typical time trend H24 (multiple seasonality may be present, i.e.: daily, weekly and seasons over the year) of bikes availability per station. Since the service acceptance is evolving quite rapidly over time, the seasonal trends taken into account are the daily and weekly ones. This means that the learning and predictions have to be continuously updated. We took into account data from June 2019 to March 2020 for Siena and Pisa stations. A clustering approach has been applied in order to classify together Pisa and Siena's stations based on their time trend of bikes availability over the day, which is also correlated to the typical services in the neighbourhoods. In detail, the K-means clustering method has been applied to identify clusters. In K-means clustering, there is an ideal center point that represents a cluster. The clustering has been performed on the basis of the H24 time trend, considering the normalized trend of bikes availability measure. The optimal number of clusters resulted to be equal to 3, and it has been identified by using the Elbow criteria [12]. In particular, each cluster represents a group of bike-sharing stations. For each cluster, we selected the representative bike rack as the one closer to the center of the considered cluster. Figure 1 reports the typical

trends during the day of the representative bike rack for each cluster.

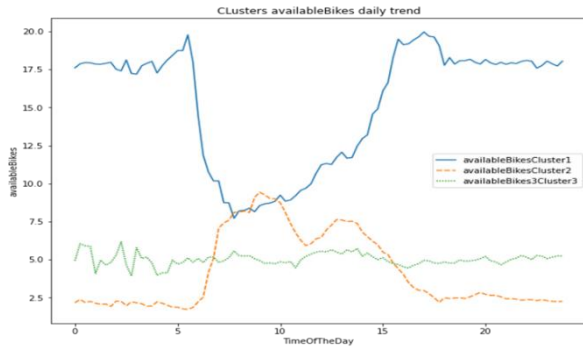


Figure 1: Typical day trend of available bikes of bike rack clusters. Cluster 1 is represented by “Stazione F.S.” in Pisa, Cluster 2 by “PoloMarzotto” in Pisa, and Cluster 3 by “Due Ponti” rack in Siena.

The bike stations/racks belonging to **Cluster 1** are typically characterized by a decrement of bike availability at lunchtime, and they are mainly located close to the railway stations, airport, mobility hubs, etc. Bike racks belonging to **Cluster 2** are typically positioned in the central area of the cities, and they are characterized by an increment of the bikes availability in the central part of the day (lunch hours, since most of the people are parking their bikes to get lunch). **Cluster 3** presents an almost uniform trend in the bike availability and bike racks are mainly positioned in the peripheral areas of the city.

Moreover, we have also detected some changes in the typical time trends from working days and weekends as shown in Figure 2. Figure 2a reports the comparison between the trend for working days and weekends for “Curtatone” station in Siena, while Figure 2b shows the trends of working days/weekends for the bike rack called “Stazione F.S” in Pisa.

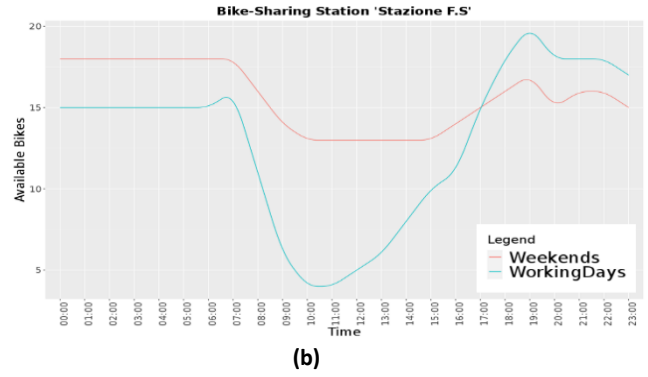
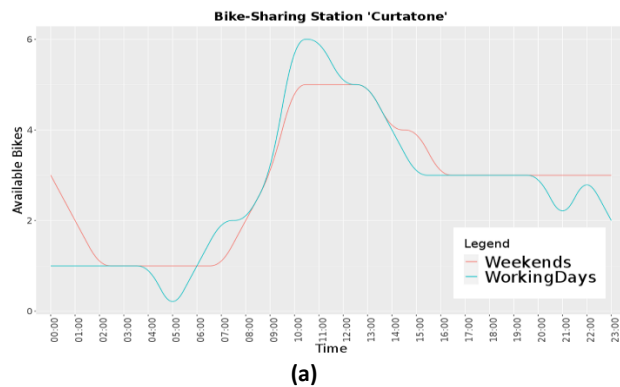


Figure 2: Working days/weekend trends of the (a) “Curtatone” bike-sharing stations in Siena and (b) “Stazione F.S” stations in Pisa municipality.

2.1 Feature Identification

With the aim of designing a prediction model, a set of features have been proposed, identified, and tested. Typically, the values are recorded every 15 minutes. Please note that the temporal window for the training is not based only on 15 minutes, but the measures over months are taken every 15 minutes.

Features belonging to the **Baseline (time series)** category refer to aspects related to the direct observation of bike status over time as in [13]. Date and time when measures are taken, the number of bikes on racks, information on weather the observation day was a weekend etc., belong to this category.

We considered also features describing the **differences over time**. Usually, the trend of the number of bikes is similar from one week to another for the same day (e.g., Monday to prev/next Monday), in the same month for example. Therefore, the following features have been included and refer to the number of available bikes at the observation time t in the day d , with respect to the previous week ($d-7$) (PwB) and the previous day ($d-1$) (PdB), as:

$$PwB = availableBikes_{d-7,t}$$

$$PdB = availableBikes_{d-1,t}$$

And thus, other features have been included in the model for capturing the difference between the number of bikes captured at the observation time (time slot t and day d) and the available bikes in the:

- previous time slot ($t-1$) of previous week ($d-7$)
dPw:

$$dPw = availableBikes_{d,t} - availableBikes_{d-7,t-1}$$

- successive time slot ($t+1$) of previous week ($d-7$)
dSw:

$$dSw = availableBikes_{d,t} - availableBikes_{d-7,t+1}$$

- previous time slot ($t-1$) of the previous day ($d-1$)
dPd:

$$dPd = availableBikes_{d,t} - availableBikes_{d-1,t-1}$$

- successive time slot ($t+1$) of the previous day ($d-1$)
dSd:

$$dSd = availableBikes_{d,t} - availableBikes_{d-1,t+1}$$

Other features have been included in the model for capturing the difference between the number of bikes captured at the observation time

$$dP2w = availableBikes_{d-7,t} - availableBikes_{d-14,t}$$

- day (d-1) and the one of two days prior (d-2) dP2d: $dP2d = availableBikes_{d-1,t} - availableBikes_{d-2,t}$

Features belonging to the **real-time weather and weather forecast** are also collected every 15 minutes (i.e., temperature, humidity and rainfall). Please note that, according to our analysis, the significant values for the weather are those related to the current time and the hour just before the measured bike availability time. For example, in order to predict the number of available bikes at the rack at 3 pm, the weather features at 2 pm and at the current time are relevant. Thus, the weather conditions influence the decisions on using the bike or other transportation means. Similarly, the weather forecast influences the plan to get the bike.

The data collected from historical values of each bike rack are in practice all the data in the learning window (several weeks or months) of the past, as described in Section 2. For each time sample, the features of Table 2 are collected and when needed estimated and stored.

Table 2: Overview of the feature used in the prediction models

Category	Feature
Baseline-Historical	Available Bikes in the past
	Time, month, day
	Day of the week
	Weekend, Holiday
	Previous week (PwB)
	Previous day (PdB)
Diff. from actual values and prev. observations	Previous observation's difference of the previous week (dPw)
	Subsequent observation's diff. of the previous week (dSw)
	Previous observation's difference of the previous day (dPd)
	Subsequent observation's difference of the previous day (dSd)
	Previous observation's difference between the previous week and two weeks earlier (dP2w)
	Previous observation's difference between the previous day and two days earlier (dP2d)
Real-time weather and weather forecast	Max Temperature Forecasted
	Min Temperature Forecasted
	Temperature
	Humidity
	Pressure
	Wind Speed
	Cloud Cover Percentage

When the long-term prediction is performed 24 hours in advance, the training/learning is performed once a day for each bike rack. Please note that performing the training more often may not produce significantly better results, and it is very computational expensive since the prediction should be performed for each bike rack.

3. Prediction Models

In the study of the model, we have tested several machine learning solutions to predict the number of available bikes at bike-sharing stations/racks. Several techniques have been discharged since they did not produce satisfactory results for long-term prediction, among them: Bayesian Regularized Neural Network that achieves an R2 (defined in the sequel) of about 0.4 for each bike-sharing station.

In this section, the results of the two best solutions are considered and compared to predict the number of available bikes at bike racks and to identify the features that could be the best predictors for the purpose. Thus, the techniques compared and reported in this paper are those that resulted to be the most effective. And in particular: **Random Forest (RF)** [14], **Gradient Boosting Machine (GBM)** [15] and the **Auto-Regressive Integrated Moving Average** (e.g., ARIMA) as a representative of the traditional statistical approaches [16]. Those solutions have been applied on the features presented in Table 1.

3.1 Assessment metrics

The accuracy of the resulting models has been evaluated against different metrics. Thus, before presenting the results, the assessment metrics are presented in this subsection. The R-squared which is defined as:

$$R^2 = 1 - \left(\frac{\sum_{i=1}^n (obs_i - pred_i)^2}{\sum_{i=1}^n (obs_i - \bar{y})^2} \right)$$

Where

$$\bar{y} = \frac{1}{n} \sum_{i=1}^n obs_i$$

The MASE (Mean Absolute Scaled Error) which is calculated as:

$$MASE = mean(|q_t|), \quad t = 1, \dots, n$$

where

$$q_t = \frac{obs_t - pred_t}{\frac{1}{n-1} \sum_{i=2}^n |obs_i - obs_{i-1}|}$$

And $obs_t = observation at time t$, $pred_t = prediction at time t$, n is the number of the values predicted over all test sets (96 daily observations per 7 days). The RMSE (Root Mean Square Error) calculated as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (obs_i - pred_i)^2}{n}}$$

The MAE (Mean Absolute Error):

$$MAE = \frac{\sum_{i=1}^n |obs_i - pred_i|}{n}$$

Among them, the MASE is clearly independent from the scale of the data. When MASE is used to compare predictive models, the best model is the one presenting the smaller MASE.

3.2 ARIMA model

The ARIMA model has been executed as multi-step forward with updated iteration technique: the forecast was computed one hour in advance. The best ARIMA model has been identical for all the clusters and resulted to be a so called (1,1,2), respectively for p, d, q order in AutoArima. ARIMA model cannot be used for medium-long term forecasts due to the relevant errors produced. An approach to cope with this problem could be to apply the forecasting ARIMA technique as a multi-step forward to make 24-hour predictions (96 time slots). In other words, to compute 24 forecasts (i.e., 1 hour in advance per 24 times): the real observations recorded in that hour (four slots of 15 minutes) are inserted into the training set, and the prediction for the next hour is computed with the new information. Therefore, the model needs to be trained every hour, so that 24 times per day per 15/20 bike-sharing stations per city, which is computationally more expensive than the others. Moreover, this approach cannot be claimed as long-term prediction. Then, the training set is updated with the observations recorded in the predicted hour and a new forecast is executed for the next hour. Table 3 shows the results for the ARIMA model for the main bike-sharing stations in the different clusters for short-term prediction.

Table 3: ARIMA multi-step forward (short term online predictions) with updated iteration results in terms of MASE and RMSE per station in Siena.

ARIMA Model Results			
MASE	RMSE	Cluster	city
0.10	2.22	1	Pisa
1.23	1.58	2	Siena
0.52	1.15	3	Siena

For this reason, the solution has been discharged, despite the fact that for the ARIMA, the obtained accuracy in terms of MASE on the short-term is better than those obtained by machine learning techniques for long terms, as presented in Table 5. Please remind that, the goal was to find a computationally viable solution to make satisfactory long-term predictions in terms of precision for several different cases.

The comparison of the needed processing time per each bike-sharing station, among the models considered above, is also relevant and it is reported in Table 6.

3.3 Experimental Results via machine learning

In detail, for **GBM** a regression tree with a maximum depth of 9 was used as a basic learner and the total number of trees was increased to 500 while the

minimum number of observations in each leaf was increased to 5. The learning rate has been set to 0.1. Note that, determining the optimal (hyperparameter) settings for the model is crucial for the bias-reduced assessment of a model’s predictive power. The choice of GBM parameters has been obtained by a hyperparameter tuning implementation. Different combinations of parameter values have been tried on the dataset (see Table 4).

Table 4: Hyperparameter ranges and types for GBM model

Hyperparameter	Type	Start	End	Default
n.tree	Integer	100	10000	100
shrinkage	Numeric	0.01	0.3	0.1
interaction.depth	Integer	3	10	1
bag.fraction	Numeric	0.1	1	0.5

The **RF** has been set with number of trees composing the forest equal to 500 and the candidate feature set equal to 1/3 of the number of the data set variables.

The result of RF and GBM machine learning solutions are compared in Table 5 with respect to the clusters, exploiting all the features presented in Table 2. The predictive models have been estimated on a training period of 7 months. MAE, MASE, RMSE and R2 measures have been estimated on a testing period of 1 week after the 7th January 2020. This comparison has highlighted that both the approaches produce similar results. On the other hand, RF is more precise in most cases obtaining a better R2. The GBM approach achieved better results only in cluster 3, which presents almost stable trends (see Figure 1) and thus less critical cases since the risk to find the rack empty is low. Moreover, the values are not very far from those obtained by RF in the same cluster.

Table 5: Machine Learning Models results and comparison for different clusters. In bold the best results for the comparison

“Stazione FS” (cluster 1)	RF	GBM
MAE	3.467	3.481
MASE	0,600	0,603
RMSE	4.136	4.296
R2	0.989	0.820
“Polo Marozotto” (cluster 2)	RF	GBM
MAE	3.108	3.214
MASE	1.209	1.250
RMSE	3.605	3.764
R2	0.985	0.763
“due Ponti” (cluster 3)	RF	GBM
MAE	1.632	1.529
MASE	0,999	0,936
RMSE	2.148	1,991
R2	0,966	0,655

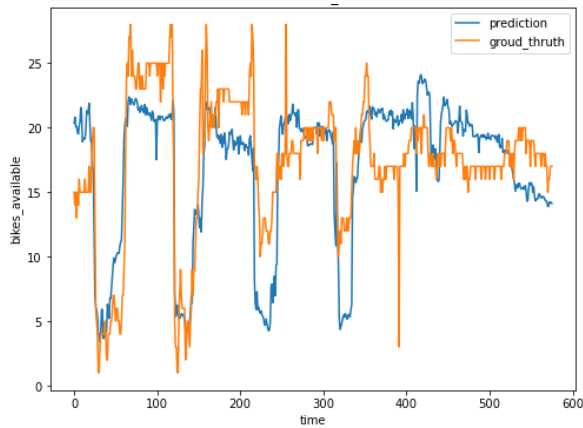


Figure 3: RF predicted values vs real in testing period for Cluster 1 reference bike rack.

3.4 Computational Costs

Table 6 shows that almost all the approaches may produce predictions every hour for the next hour in a reasonable estimation time. On one hand, in order to produce satisfactory predictions, the ARIMA approach needs to re-compute the training every hour (even if the online training can be seen as an alternative it is also a computational cost). This is a quite expensive cost of about 30s for each bike-sharing station, due to the fact that the charging stations can be hundreds. On the other hand, machine learning models (i.e., GBM and RF) provide predictive models with 96 values in advance with quite satisfactory results, they produce better results with less effort with respect to ARIMA. GBM processing time is quite low and results in terms of error measure are better with respect to the RF. GBM model can be considered the best solution for a real-time application.

Table 6: Forecasting Models comparison in terms of processing time

Processing Time	ARIMA	RF	GBM
Average training time	30.9 sec	410.3 sec	21.8 sec
Training frequency	1 time per hour	1 time per day	1 time per day
Training period	1 months	7 months	7 months
Forecast window	1 hour	1 day	1 day

3.5 Feature Relevance

In Figure 4, the feature’s relevance [15] for the three clusters has been reported by considering RF and GBM. From the comparison it should be noted that both techniques present almost the same features in the first 5 most relevant features.

The most important features are those related to the past values of the time series (available bikes), to *Time*, *Day of the Week*, *weekend (yes or no)*, *Day*. The

information regarding weather such as Air pressure, humidity and temperature are less relevant.

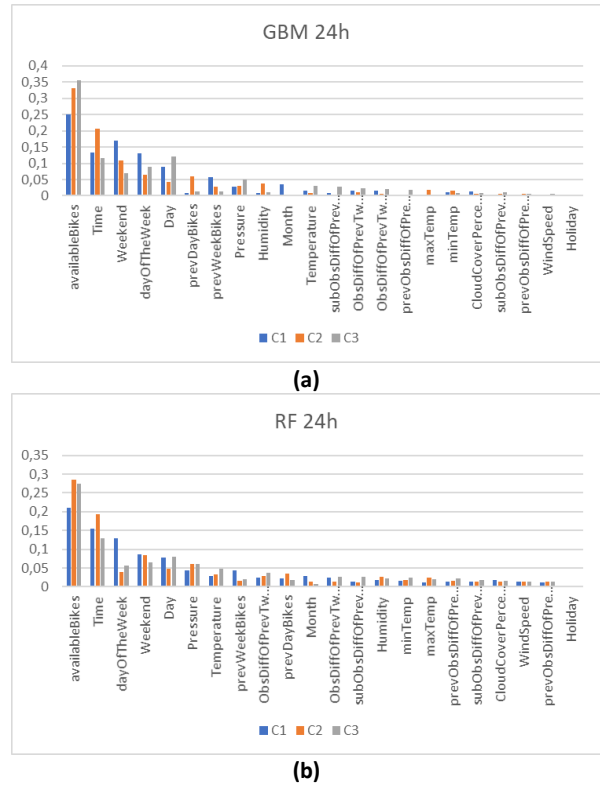


Figure 4: Feature relevance for the RF and GBM with respect to the clusters of bike racks.

3.6 Feature Reduction

According to Table 2, the features are classified into three main groups: temporal, weather, and differential. In addition, it can be observed that the top 5 features are those belonging to the temporal category. In Table 7, the impact of reducing the feature space is reported, the case in which all features are considered has been already reported in Table 5.

Table 7: Impact of feature reduction to precision of predictions in the different clusters: C1, C2 and C3.

		RF			GBM		
		c1	c2	c3	c1	c2	c3
Temporal	MAE	4.36	3.85	4.22	3.93	4.27	1.85
	MASE	0.75	1.33	0.73	0.68	1.47	1.13
	RMSE	5.71	4.61	5.03	4.89	4.88	2.41
	R2	0.98	0.98	0.98	0.78	0.72	0.63
Temporal + Weather	MAE	4.22	3.12	1.68	3.69	3.26	1.52
	MASE	0.73	1.18	1.03	0.64	1.22	0.93
	RMSE	5.03	3.6	2.19	4.38	3.84	1.96
	R2	0.98	0.98	0.96	0.81	0.76	0.65
Temporal + differential	MAE	3.19	3.89	1.72	3.32	4.21	1.58
	MASE	0.55	1.35	1.05	0.57	1.47	0.97
	RMSE	3.87	4.50	2.31	4.19	4.88	2.08
	R2	0.98	0.98	0.96	0.79	0.73	0.65

It can be noted that the RF results to be the best ranked in terms of R2 with respect to GBM in all cases.

In addition, it can be observed a general improvement of performance with the increment of features, as usual in RF and GBM. The weather, as well as the differential features, may lead to gain about 1% in terms of MAE (the average MAE for RF is about 4.14 for Temporal only, and 3.01 for Temporal + Weather, and 2.93 for Temporal + Differential). This analysis is providing some evidence that to compute all the features may increase the precision of a small amount at the expense of much higher computational costs.

4. Conclusions

In this paper, we proposed machine learning methods to predict the number of available bikes 24 hours in advance in any station of bike sharing systems. The proposed methods use a model which takes high dimensional time-series data from the smart bike station and uses real-time and forecast weather information as input to perform the long-term prediction. ARIMA model cannot be used for long term forecasts (24 hours in advance) because the iterative forecasting model should be trained at least 24 times per day per several bike-sharing stations per city. To this aim, RF and GBM algorithms have been considered as alternative finding a satisfactory computationally viable solutions to make long-term predictions that produce satisfactory results in terms of precision.

In the models, we have considered several features, such as the *Baseline-Historical data*, the *difference among actual values and previous observations*, the *Real-time weather and weather forecast*. In almost all predictive models, the top 5 features are those belonging to the *Baseline-Historical* category according to the feature relevance analysis performed. Please note that, despite the different trends of the clusters, in all cases the identified features and model have been the same, thus demonstrating the validity of the derived results. Using all the features may increase the precision of the models of a small amount compared to reducing the feature space to the top 5 or including also the weather or the differential metrics.

The entire approach resulted to be very flexible and robust with respect of the sporadic lack of data samples. The predictive models can produce predictions 24 hours in advance via mobile Apps. The solution has been deployed as a feature of Smart City Mobile Apps in the Tuscany area to encourage sustainable mobility. <https://play.google.com/store/apps/details?id=org.disit.toscana>

Acknowledgment

The authors would like to thank the MIUR, the University of Florence and companies involved for co-founding Sii-Mobility national project on smart city mobility and transport. Km4City and Snap4City (<https://www.snap4city.org>) are open technologies and research of DISIT Lab. Sii-Mobility is grounded and has contributed to Km4City open solution. In addition, the authors would like to thank to Gabriele Bruni for his

support in the early experiments of this research.

References

- [1] Flandrin P. Robardet C. Rouquier J. Borgnat P., Abry P. and Fleury E. "Shared Bicycles in a City: a Signal Processing and Data Analysis Perspective," *Advances in Complex Systems*, vol.14, n.3, 2011, pp.415-438.
- [2] Hulot, Pierre, Daniel Aloise, and Sanjay Dominik Jena. "Towards station-level demand prediction for effective rebalancing in bike-sharing systems." Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018
- [3] Lin, Jenn-Rong, and Ta-Hui Yang. "Strategic design of public bicycle sharing systems with service level constraints," *Transportation research part E: logistics and transportation review*, vol.47, n.2, 2011, pp.284-294.
- [4] Froehlich, Jon Edward, Joachim Neumann, and Nuria Oliver. "Sensing and predicting the pulse of the city through shared bicycling," *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.
- [5] Kaltenbrunner, Andreas, et al. "Urban cycles and mobility patterns: Exploring and predicting trends in a bicycle-based public transport system," *Pervasive and Mobile Computing*, vol.6, n.4, 2010, pp.455-466.
- [6] Flandrin P. Robardet C. Rouquier J. Borgnat P., Abry P. and Fleury E. "A dynamical network view of Lyon's Velo'v shared bicycle system," *Dynamics On and Of Complex Networks*, Volume 2. Birkhäuser, New York, NY, 2013, pp.267-284.
- [7] Gupta S. Ma D. Bargar A., Gupta A. "Interactive visual analytics for multicity bikeshare data analysis," *The 3rd International Workshop on Urban Computing*, New York, USA, Vol. 45, 2014.
- [8] Colace, Francesco, et al. "A multilevel graph approach for predicting bicycle usage in London area." Fourth International Congress on Information and Communication Technology. Springer, Singapore, 2020.
- [9] C. Badii, P. Nesi, I. Paoli. "Predicting available parking slots on critical and regular services exploiting a range of open data," *IEEE Access*, 2018, <https://ieeexplore.ieee.org/abstract/document/8430514/>
- [10] C. Badii, E. G. Belay, P. Bellini, D. Cenni, M. Marazzini, M. Mesiti, P. Nesi, G. Pantaleo, M. Paolucci, S. Valtolina, M. Soderi, I. Zaza. "Snap4City: A Scalable IOT/IOE Platform for Developing Smart City Applications," *Int. Conf. IEEE Smart City Innovation*, China 2018, IEEE Press. DOI: <https://ieeexplore.ieee.org/document/8560331/>
- [11] C. Badii, P. Bellini, A. Difino, P. Nesi. "Smart City IoT Platform Respecting GDPR Privacy and Security Aspects," *IEEE Access*, 8 (2020): pp.23601-23623.
- [12] Kodinariya, T. M., & Makwana, P. R. "Review on determining number of Cluster in K-Means Clustering," *International Journal*, vol.1, n.6, pp.90-95, 2013.
- [13] Kim, Kyoungok. "Investigation on the effects of weather and calendar events on bike-sharing according to the trip patterns of bike rentals of stations." *Journal of transport geography* 66 (2018): 309-320.
- [14] Breiman, Leo. "Random forests," *Machine learning*, vol.45, n.1, 2001, pp.5-32.
- [15] J. H. Friedman. "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol.29, n.5, pp.1189-1232, 2001.
- [16] Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. "Time series analysis: forecasting and control," John Wiley & Sons, 2015.
- [17] United Nations, Department of Economic and Social Affairs, Population Division (2018). *World Urbanization Prospects: The 2018 Revision*, Online Edition
- [18] Chappert B. Taille A. D. L. Laroche F. Meunier F. Benchimol M., Benchimol P. and Robinet L. "Balancing the stations of a self-service "bike hire" system," *RAIRO-Operations Research*, vol.45, n.1, 2011, pp.37-61.

- [19] Meunier F. Chemla D. and Wolfler-Calvo R. "Balancing a bike-sharing system with multiple vehicles", Proceedings of Congress annual de la société Française de recherche opérationnelle et d'aide la décision, ROADEF2011, Saint-Etienne, France, 2011.
- [20] Morency C. Contardo C. and Rousseau L. "Balancing a dynamic public bike-sharing system," Vol. 4. Montreal, Canada: Cirrelt, 2012.
- [21] Mattefeld D. C. Vogel P., Greiser T. Understanding bike-sharing systems using data mining: "Exploring activity patterns," *Procedia-Social and Behavioral Sciences*, vol.20, pp.514-523, 2011.
- [22] Calabrese F. YoonJ. W., Pinelli F. "Cityride: a predictive bike sharing journey advisor", 13th International Conference on Mobile Data Management, IEEE, 2012.
- [23] Boris Bellalta Gabriel Martins Dias and Simon Oechsner. "Predicting occupancy trends in barcelona's bicycle service stations using open data," *sai intelligent systems conference (intellisys)*, IEEE, 2015.
- [24] Daniël Reijtsbergen Mirco Tribastone n Nicolas Gast, Guillaume Massonnet. "Probabilistic forecasts of bike-sharing system for journey planning," *Proceedings of the 24th ACM international on conference on information and knowledge management*, 2015.
- [25] Yuanchao Shu Peng Cheng Jiming Chen Thomas Moscibroda Zidong Yang, Ji Hu. "Mobility modeling and prediction in bike-sharing systems," In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, pp.165-178.
- [26] Lin, Lei, Zhengbing He, and Srinivas Peeta. "Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach," *Transportation Research Part C: Emerging Technologies*, vol.97, 2018, pp.258-276.
- [27] Wang, Bo, and Inhi Kim. "Short-term prediction for bike-sharing service using machine learning," *Transportation research procedia*, vol.34, 2018, pp.171-178.
- [28] Ashqar, Huthaifa I., et al. "Modeling bike availability in a bike-sharing system using machine learning," 5th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS), IEEE, 2017.
- [29] Chen, Po-Chuan, et al. "Predicting station level demand in a bike-sharing system using recurrent neural networks," *IET Intelligent Transport Systems*, 2020.

Journal of Visual Language and Computing

Volume 2021, Number 1