Boosting Climate Analysis With Semantically Uplifted Knowledge Graphs

Jiantao Wu, Fabrizio Orlandi, Declan O'Sullivan, Enrico Pisoni, and Soumyabrata Dev, *Member, IEEE*

***Abstract*—Nowadays, the fast expansion of heterogeneous climate data resources accessible on the Internet has led to substantial data fragmentation on the web. For example, station-based sensor data from different sources are likely to be interrelated but may be stored in disparate formats, such as CSV, JSON, and XML. To address the data isolation problem, several semantically uplifted knowledge graphs are proposed for climate data exchange. While these knowledge graphs improve data interoperability, the advancement in multisource data interchange is limited to data included inside knowledge graphs. As a result, the exclusive interoperability of current climatic knowledge graphs hampers the flow of data into typical climate analysis workflows in contexts, where analytical models often need data in nonknowledge graph formats. This article addresses this issue by focusing on enhancing climate analysis workflows within the context of the Python machine learning environment, with a preference for tabular data. We propose an analysis workflow able to automatically integrate remote climate knowledge graph data with local tabular data so as to enhance the data usability with respect to certain climate analysis tasks. To underscore the importance of our study, we illustrate how the workflow streamlines the access to multisource climatic variables in the Python environment from a semantic perspective. The additional knowledge graph data have the potential to augment local datasets in the climate domain, as evidenced by an improvement in accuracy of up to 10% for machine learning geared on rainfall detection.**

***Index Terms*—Climate data, knowledge graphs (KGs), linked data, machine learning, semantic webs.**

Nomenclature

CA	Climate analysis.
CCTL	Climate change timeline.
KG	Knowledge graph.
KNN	K-nearest neighborhoods.
LOD	Linked open data.
LoG	Linked OntoGazetteer.
NOAA	National Oceanic and Atmospheric Administration.
PCC	Pearson correlation coefficient.
PM2.5	Fine particulate matter.
PRCP	Precipitation.
RDF	Resource description framework.
RF	Random forest.
SNWD	Snow depth.
SOSA	Sensor, observation, sample, and actuator.
SPARQL	SPARQL protocol and RDF query language.
SVC	Support vector classifier.
TAVG	Average temperature.
TMAX	Maximum temperature.
TMIN	Minimum temperature.

I. Introduction

THE RDF is a W3C-recommended standard paradigm for data exchange on the web. It enables the data interoperability by allowing them to be merged even when the underlying schemas vary and also allows schema development over time without needing all the data consumers to be updated. RDF datasets constructed based on the semantic model are called as KGs in the modern way. By integrating nodes and edges in a semantic model, the KGs conceive information, such as events and connections between things in an intelligent manner, for example, KG could be used to derive new information based on semantic rules. From this perspective, KGs have been constructed for a number of different uses. For example, by using KGs as its databases, Google[1] was able to enhance the intelligence of its search engine. Wikidata[2] and DBpedia[3] are encyclopedia databases that are built with the help of KGs. However, there are many sectors that produce enormous quantities of data but are underexplored in terms of using KG to develop intelligence on top of the data [1]. One such area is climate, which has made large quantities of sensor data publicly available. Numerous research works have utilized semantic technologies to mitigate the data isolation problem in the climate domain. For instance, Wu *et al.* [2] propose CA ontology for transforming CSV-formatted NOAA[4] climate sensor observations into RDF-formatted data that are

Manuscript received February 18, 2022; revised April 23, 2022; accepted May 6, 2022. Date of publication May 24, 2022; date of current version June 17, 2022. This work was supported in part by the Science Foundation Ireland under Grant 13/RC/2106_P2 at the ADAPT SFI Research Centre at University College Dublin. ADAPT, in part by the SFI Research Centre for AI-Driven Digital Content Technology, in part by the Science Foundation Ireland through the SFI Research Centre's Program, and in part by the EU H2020 Research and Innovation Program under the Marie Skłodowska-Curie under Grant 713567 at the ADAPT SFI Research Centre at Trinity College Dublin. *(Corresponding author: Soumyabrata Dev.)*

Jiantao Wu and Soumyabrata Dev are with the ADAPT SFI Research Centre, School of Computer Science, University College Dublin, 4 Dublin, Ireland (e-mail: jiantao.wu@ucdconnect.ie; soumyabrata.dev@ucd.ie).

Fabrizio Orlandi and Declan O'Sullivan are with the ADAPT SFI Research Centre, School of Computer Science and Statistics, Trinity College Dublin, 2 Dublin, Ireland (e-mail: fabrizio.orlandi@tcd.ie; declan.osullivan@tcd.ie).

Enrico Pisoni is with the European Commission Joint Research Centre, 21027 Ispra, Italy (e-mail: enrico.pisoni@ec.europa.eu).

Digital Object Identifier 10.1109/JSTARS.2022.3177463

[1] [Online]. Available: https://developers.google.com/knowledge-graph
[2] [Online]. Available: https://www.wikidata.org/
[3] [Online]. Available: https://www.dbpedia.org/
[4] [Online]. Available: https://www.ncdc.noaa.gov/

then stored in their climate KG—Link-Climate [3]—which are utilized to provide the different linked data sources of climate data to climate communities. Pileggi developed a knowledge base of climate-change-related facts organized chronologically using their CCTL ontology [4]. Surya *et al.* [5] incorporate sociogeological considerations into their climate change ontology model. Research in the semantic web community is more likely to develop ontology models for climate data based on specific use cases, which transforms the data schema into an ontology, potentially increasing data interoperability. They are, however, less focused on the intelligent use of climate KG data (e.g., allowing the semantic data augmentation on the tabular machine learning data feed).

Unfortunately, one major issue that accounts for the limited use of KGs for intelligent applications is that contemporary climate KGs, like ours, are the devoid of data exchange between KGs and machine learning pipelines frequently used in the PyData environment. Certain researchers still prefer to build machine learning models directly on a fixed dump of tabular data [6], which is often limited in terms of variety and amount. A significant disadvantage is that it may not be capable of developing highly reliable forecasting, since meteorological phenomena, such as rainfall, maybe highly reliant on a huge number of factors other than a local dataset. We offer a method for applying machine learning techniques with the easy combination of our climate KG by bridging our KG to PyData environment and, thus, simplify the collection of climate data within the PyData environment for machine learning workflows in this article. In contrast to other climate KGs, the additional data communication channel created for machine learning pipelines enables users to perform machine learning on climate-associated tasks automatically employing our KG climate dataset. Especially, the machine learning pipelines can advance the users' fixed climate tabular data (e.g., `CSV`) by combining our remote climate KG data. When applied to the NOAA original `CSV` dataset and enhanced with our climate KG atmospheric sectors, some common machine learning classification models offer many advantages over current weather prediction and classification tasks in terms of increased accuracy (up to 10% for rainfall detection), dependability, and reusability. A summary of the contributions of this article[5] to the climatic analysis field is listed as follows.

1) A well-documented open climatic KG is created by us to supply online multisource climate data for data consumers. The KG now covers weather, atmospheric, and air quality data and is continuously being enriched with more data in the climate areas.
2) Our climatic KG complies with linked data principles, which enables a wider data accessibility to other linked data. For example, we pair our weather stations with geographical context from Wikidata and DBpedia, which enables connections with other linked data through the geographical entities.
3) We bridge the interoperability advance of our climatic KG to PyData machine learning pipelines, allowing an easy obtainment of multisource climate data to boost climatic analysis with semantics that eliminates the need of handling data heterogeneity.
4) We perform a real machine learning case study w.r.t. the rainfall detection based on the NOAA tabular data to demonstrate the effectiveness of our article in terms of advantages in multisource data preprocessing and machine learning performance improvement (up to 10%) with the additional use of our climatic KG.

II. Related Work

Owing to the diversity of sensor data gathered globally, including air pollution, weather, and satellite reanalysis, these data are heterogeneous and supplied in a number of data formats (e.g., `CSV`, `JSON`, and `NetCDF`) [7]. It is challenging to enhance the data intelligence from these separate datasets [8], [9]. A widely accepted idea in recent research is to create a web of data by linking various disparate data sources. This requires that data adhere to four linked data principles [10]: 1) use uniform resource identifiers (URIs) as names for things; 2) use HTTP URIs so that people can look up those names; 3) when someone looks up a URI, provide useful information, using the RDF standard; and 4) include links to other URIs. Numerous studies have made strides in integrating various data sources into KGs complying with linked data principles in the climate domain. Typically, the aim is to provide more detailed information for specific applications. LoG [11] is an ontological gazetteer built from a collection of open KGs [e.g., GeoNames,[6] DBpedia] in order to provide additional KG-based underlying context for reference data used in textual geographical information retrieval. In addition, the KGs produced by researchers from diverse disciplines may be approved for publication on the LOD cloud,[7] to bolster the collection of linked datasets. Until May 2020, the LOD cloud has 1301 datasets with 16 283 links, and the number of datasets continues to increase rapidly. Despite the fact that the LOD cloud is an effort aimed at providing researchers with globally interoperable data, the data quality is debatable. Debattista *et al.* [12] presented an assessment measure to evaluate the LOD's data quality; however, their findings indicate that the average score is less than 60%, which represents the current state of LOD's data quality in any discipline (including the climate). On the other hand, since the schema (or ontology) for RDF triples is not standardized, documentation should be provided by KGs to assist users in either updating data or formulating SPARQL queries. This, however, is often overlooked by research.

To ensure a higher quality of climate data and sufficient context information to assist nonexperts in navigating the complex process of requesting climate KG data, our recent research developed an online portal[8] supplying a detailed documentation for our climate KG to assist researchers in comprehending the CA ontology and easily locating climate data that meet their

[5] In the spirit of reproducible research, all the source code is [Online]. Available: https://github.com/futaoo/kg-climate-analysis.

[6] [Online]. Available: http://www.geonames.org

[7] [Online]. Available: https://lod-cloud.net/

[8] [Online]. Available: http://jresearch.ucd.ie/linkclimate/

data requirements [13], [14]. In addition, we provide linked geographical context for our KG's spatial entities using only high-quality linked data from LOD sources, such as Wikidata and DBpedia, to guarantee that the data remain alive and useful over time. Despite the fact that our climate KG provides a broader range of data for climate studies, mining information by connecting various data sources to KGs and the requirement for automated KG data input into machine learning models (preferring tabular format), which are increasingly being explored for increased artificial intelligence [15], is hardly met by the climate KG. Currently, some relevant research works have concentrated on the integration of KGs with machine learning. Li *et al.* [16] use external KGs to augment machine learning prediction tasks for students' mental health condition. Lei *et al.* [18] propose to constrain optimization goals with KGs [17] to guide the model's learning for illness classification. Annervaz *et al.* [19] train deep learning models to obtain "world knowledge" from KGs and integrate them in sequent models for downstream tasks.

Unfortunately, to the best of our knowledge, the interoperable combination of machine learning and online climate KGs for climate-related tasks has been studied seldom if ever. Those studies, however, tend to download the whole dumps of KG data (i.e., with minimum interoperability). For instance, Annervaz *et al.* [19] used a couple of KG dumps, including Freebase 15K (FB15k),[9] WordNet18 (WN18)[10] and DBpedia ontology,[11] as the raw data feed. A critical disadvantage of the data dump is its inability to be processed dynamically to suit a variety of requirements, such as the dynamic acquisition of KG data [11]. In addition, consider a scenario that someone questions if a subset of the KG data is adequate to enable the models to perform promisingly in the Annervaz's experiment and tries to derive a number of subgraphs to make comparisons. All of the data dumps will, then, be loaded into memory and parsed as graphs in order to extract the subgraphs.

A current research trend in relation to this issue is to create a framework directly integrating machine learning pipelines with the SPARQL endpoints, which interfaces the online KGs by providing programmable query services. `RDFFrames` [20] is a framework written on the Python code to implement imperative programming in replace of SPARQL declarative programming to obtain the KG data. Using `RDFFrames`, users are able to formulate queries within a PyData ecosystem, and the queries will be translated into equivalent SPARQL queries to get the KG data. Another recent framework is `kgextension` [21], which focuses on integrating KG with the popular Python data mining pipelines—`Scikit − Learn` to enable tabular data readily to be linked to a remote KG so as to increase the data variety for local data. Typically, it contains a range of entity linking methods (e.g., DBpedia's Spotlight Entity Linking[12]) to facilitate the identification of entities to those equivalents in a remote KG. The common feature is that they all achieved the querying, manipulating, and selecting of KG data in a highly programmable manner, and the returned data are all formatted with the popular `Pandasdataframe`.

In this article, we combine Link-Climate preferably with `kgextension`, which can help us deal with the cases that users want to use Link-Climate graph data to be integrated with their local tabular data.

III. Methodology

In this section, we demonstrate how to utilize `kgextension`[13] and our climate KG to create a workflow that can help users easily benefit from the gain on climate machine learning tasks by integrating our climate KG with other datasets (such as `CSV` and other KGs). An overview of the workflow is shown in Fig. 1, where the area in gray color stands for the KG components of the workflow, and the blue area stands for components in relation to the PyData environment. The steps are ordered by the sequential numbers.

A. Leveraging Link-Climate KG as the Auxiliary Data Source

Link-Climate is a climate KG that we built as a derived work of many online climatic data services, such as NOAA online climate data[14] and PurpleAir's atmospheric data[15] in certain European cities. Link-Climate adopts the RDF triple store as the database implementation and has conformed to linked data principles. This enables that other RDF triple stores' facts linked to Link-Climate can be accessed conjointly through federated SPARQL queries [22]. We added geographical context for spatial entities in the Link-Climate from DBpedia and Wikidata, allowing users to utilize federated queries to query more information from these contextual KGs through the linked entities. Hence, Link-Climate not only offers the additional data for climate studies but also allows for the expansion of the existing meteorological data via federated queries to other usable linked datasets.

Various datasets transformed into Link-Climate are structured with the same set of ontologies. In contrast to schemas that are designed independently for multiple data sources, ontologies are generic semantics that are not tied to particular datasets. This enables the organization of multisource data at the general concept level in reference to only human knowledge in an interoperable manner. Considering the scenario when people want to find atmospheric data from NOAA and PurpleAir, they will unable to do so until they identify the exact meaning of the data from two distinct data sources. They may need to read a number of dataset-specific documentations since NOAA and PurpleAir do not organize their data in the same manner (e.g., different name conventions). However, if the datasets have been uplifted by shareable ontologies and transformed into Link-Climate KG, people can easily find and manipulate NOAA and PurpleAir data using SPARQL queries. In machine learning practice, the benefit of shareable ontologies can be further explored for feature selection. To illustrate this, we begin with

[9] [Online]. Available: https://paperswithcode.com/dataset/fb15\;k

[10] [Online]. Available: https://paperswithcode.com/dataset/wn18

[11] [Online]. Available: https://www.dbpedia.org/resources/ontology/

[12] [Online]. Available: https://www.dbpedia.org/resources/spotlight/

[13] [Online]. Available: https://github.com/om-hb/kgextension

[14] [Online]. Available: https://www.ncdc.noaa.gov/cdo-web/

[15] [Online]. Available: https://www.purpleair.com/

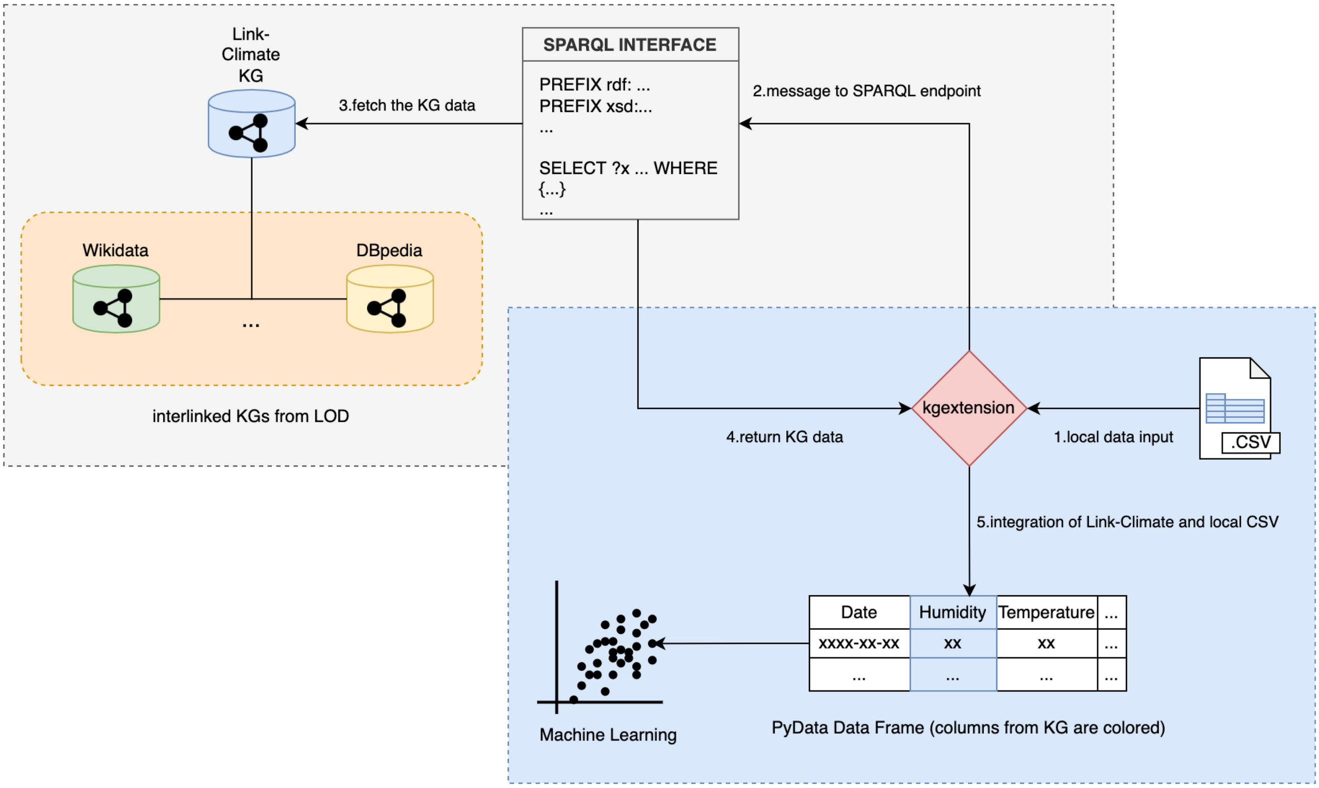


Fig. 1. Sketch of our proposed workflow.

a part of the ontology model for the Link-Climate atmospheric data being used for the following KG-augmented experiments. We use the data transformed from PurpleAir-hosted worldwide sensors as an example covering measurements of PM2.5, air humidity, air pressure, and temperature. The model's primary ontology is SOSA.[16] Along with SOSA, we developed climate domain-specific vocabulary in order to manage the resources associated with PurpleAir's sensor data. To keep things simple and focused, two components of the ontology model will be presented through graphical views[17]: the modeling of the PM2.5 index (see Fig. 2) and the modeling of the associated air sensor (see Fig. 3).

A short description of some of the main semantic classes and properties used in our model (and in the above figures) is given as follows:

Classes:

sosa:Observation—a measurement to the value of a property of a feature of interest;

sosa:Sample—a sample representative of a feature of interest;

sosa:ObservableProperty—an observable property of a feature of interest;

sosa:Sensor—a device, agent to conduct a procedure which determines how observations will be made;

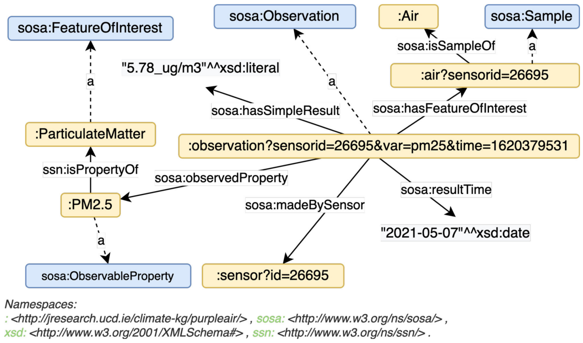


Fig. 2. Ontology modeled PurpleAir's PM2.5 observation (node ":?sensorid=26695&var=pm25&time=1620379531"), and uncolored nodes are literals of different data types.

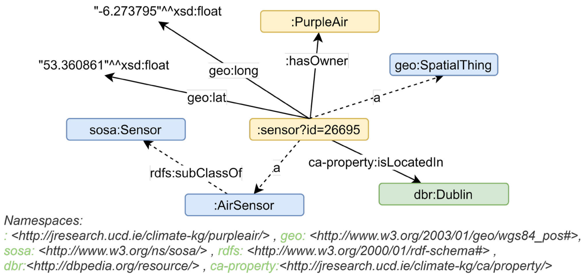

"-6.273795"^^xsd:float
:PurpleAir
:hasOwner
"53.360861"^^xsd:float
geo:long
geo:SpatialThing
geo:lat
a
sosa:Sensor
:sensor?id=26695
ca-property:isLocatedIn
rdfs:subClassOf
a
:AirSensor
dbr:Dublin
Namespaces:
: <http://jresearch.ucd.ie/climate-kg/purpleair/> , geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>,
sosa: <http://www.w3.org/ns/sosa/> , rdfs: <http://www.w3.org/2000/01/rdf-schema#> ,
dbr:<http://dbpedia.org/resource/> , ca-property:<http://jresearch.ucd.ie/climate-kg/ca/property/>

Fig. 3. Ontology model for a PurpleAir's sensor (node ":sensor?id=26695").

geo:SpatialThing—a class for representing anything with a spatial extent, i.e., size, shape, or position;

geo:lat—the latitude of a spatial thing;

[16] [Online]. Available: https://www.w3.org/TR/vocab-ssn/

[17] **Note**: Ontology vocabularies in the figures are already associated with web addresses (to meet the linked data principles) and comply with the form {prefix}:{literal term}, i.e., the name spaces are prefixed to the literal names of the node (the prefix starts with ":" means the name space defined by this work).

geo:long—the longitude of a spatial thing.

Properties:

sosa:hasSimpleResult—a simple value of an observation;

sosa:madeBySensor—linking an observation to the sensor that generates it;

sosa:resultTime—denoting the time when the observation is made;

: hasOwner—linking a sensor to its host organization;

ca-proerty:isLocatedIn—the property used in CA ontology to denote the administrative division where a thing is located in.

As shown in the ontology model for PurpleAir's air sensor data, the ontology model makes use of a variety of human-readable vocabulary (e.g., "hasSimpleResult" and "madeBySensor") to demonstrate the semantic connections between data. To use the ontology models for feature selection from datasets, data consumers must be acquainted with the ontology vocabulary in order to manipulate the Link-Climate KG. We direct visitors to the data portal[18] for a better understanding of the various ontology models utilized in Link-Climate. Given an expert of the Link-Climate ontology model, we assume that the expert is interested in only atmosphere variables of Link-Climate in Dublin for the downstream experiments. The following SPARQL query pattern can be formulated to precisely locate the data of interest. Furthermore, this pattern is not dataset specific prior to transformation into Link-Climate, allowing for the selection of semantically identical data across many datasets. This stage is critical for removing irrelevant data from machine learning tasks and speeding up the feature selection procedure. To further completing a full cycle of feature selection, e.g., to determine each feature importance in relation to a particular machine learning task, we propose tunneling Link-Climate into PyData ecosystem and conducted a real rainfall detection experiment, as illustrated in the following sections.

```
BASE
↪ <http://jresearch.ucd.ie/climate-kg/purpleair/>
PREFIX ca_prop:
↪ <http://jresearch.ucd.ie/climate-kg/ca/property/>
PREFIX sosa: <http://www.w3.org/ns/sosa/>
PREFIX ssn: <http://www.w3.org/ns/ssn/>
PREFIX dbr: <http://dbpedia.org/resource/>

SELECT DISTINCT ?sensor ?atm_var WHERE{
  ?atm_obs sosa:madeBySensor ?sensor;
           sosa:observedProperty ?atm_var .
  ?sensor ca_prop:isLocatedIn dbr:Dublin .
  ?atm_var ssn:isPropertyOf <Atmosphere> .

}
```

Listing 1. A SPARQL query that retrieves available atmospheric variables in Dublin.

B. Connecting Link-Climate in PyData Ecosystem

The PyData ecosystem includes a significant variety of vertically scalable and simple-to-use solutions such as Pandas, NumPy, and Scikit-Learn that are popularly used by climate studies for analytical workflows creation. The Link-Climate KG, on the other hand, provides a SPARQL interface for acquiring additional climate data. Typically, the solutions to "SELECT"-oriented data queries are in the tabular forms formatted with CSV or JSON. To allow potential climate studies to enhance their data by adding data from remote Link-Climate, a difficulty being worth addressing is to transform query solutions to be consistent with local data and then combine and feed it into the analytical workflows. As far as we know, numerous Python machine learning pipeline implementations require that the input data are in the Pandas Dataframe format. Given the growing popularity of Pandas Dataframe as a data feeding format for machine learning, we attempt to allow kgextension to submit SPARQL queries to the external Link-Climate KG rather than our private SPARQL endpoint service. The advantage is that kgextension automatically converts the results of SPARQL queries to Pandas Dataframe. It removes the need for data transformation when using machine learning models implemented by Python code. Listing 2 is a piece of trial code developed in the kgextension programming functions that connects the remote Link-Climate KG, submits the query, and converts the query solution to a Pandas Dataframe.

```
import kgextension
from kgextension.sparql_helper import
↪  endpoint_wrapper

link_climate = RemoteEndpoint(
url="http://jresearch.ucd.ie/kg/air-pollutants",
timeout=120, requests_per_min=100*60, retries=10,
↪  page_size=10000)

query = "SELECT ?subject ?predicate ?object WHERE
↪  {?subject ?predicate ?object .} LIMIT 25"

df_solution =
↪  endpoint_wrapper(endpoint=link_climate,
↪  query=query)
```

Listing 2. A piece of trial code to connect the remote Link-Climate KG.

IV. Case Study on Rainfall Detection

In this section, we demonstrate how to use Link-Climate KG within PyData environment to enhance a local NOAA daily summary dataset via the kgextension pipeline. The enhancement effect will be evaluated with respect to rainfall detection and see if the inclusion of Link-Climate data can improve NOAA daily summary data for daily rainfall detection tasks.

A. Data Preparation

As a starting point, we manually downloaded daily weather data for the Dublin Phoenix Park for a one-year period (from 2019.06.01 to 2020.06.01) from the NOAA, including daily maximum and minimum temperatures, daily precipitation amounts, daily average temperatures, and daily snow depth, which are compiled in a CSV file by the NOAA and used as

[18] [Online]. Available: http://jresearch.ucd.ie/linkclimate/

TABLE I
WEATHER INDEXES COVERED BY NOAA CLIMATE DATA; UNITS COMPLY WITH THE INTERNATIONAL SYSTEM OF UNITS

	STATION	NAME	DATE	PRCP	SNWD	TAVG	TMAX	TMIN
0	EI000003969	DUBLIN PHOENIX PARK, EI	2019-06-01	1.2	0.0	13.3	20.0	7.5
1	EI000003969	DUBLIN PHOENIX PARK, EI	2019-06-02	0.0	0.0	14.9	19.3	13.6
...	...	...	...	...	...	...	...	...
366	EI000003969	DUBLIN PHOENIX PARK, EI	2020-06-01	0.0	0.0	14.4	21.1	9.3

TABLE II
LOD CLOUD'S GEOGRAPHICAL INFORMATION INCORPORATED FOR THE STATION IN TABLE I

	STATION	NAME	place_at_DBpedia	in_city	latitude	longitude
0	EI000003969	DUBLIN PHOENIX PARK, EI	`http://dbpedia.org/resource/Phoenix_Park`	`http://dbpedia.org/resource/Dublin`	53.36	-6.33

TABLE III
`dataframe` CONTAINS ALL AIR SENSORS OF LINK-CLIMATE KG LOCATED IN DUBLIN

	sensor	in_city	lat	lon
0	`http://jresearch.ucd.ie/climate-kg/purpleair/sensor?id=26695`	`http://dbpedia.org/resource/Dublin`	53.360861	-6.273795
1	`http://jresearch.ucd.ie/climate-kg/purpleair/sensor?id=105508`	`http://dbpedia.org/resource/Dublin`	53.33499	-6.216279
2	`http://jresearch.ucd.ie/climate-kg/purpleair/sensor?id=59111`	`http://dbpedia.org/resource/Dublin`	53.301456	-6.258461
3	`http://jresearch.ucd.ie/climate-kg/purpleair/sensor?id=91889`	`http://dbpedia.org/resource/Dublin`	53.381096	-6.06105

the raw input for the machine learning models. The `CSV` file is described in full in Table I. Each variable's value is recorded in relation to the date; thus, each variable is a time series with a basic time step of one day.

B. Station's Geographical Information Augmentation With LOD Cloud

The station information in Table I only consists of the name (NOAA uses the location of the station as the name) and station code, which is less accessible in a KG as the literals are often not provided with URIs for identification purpose. Interlinking the literals to augment data is the least practical. Since the name is given in natural language by the NOAA, we use DBpedia Spotlight[19] through `kgextension` to find the equal places (with URIs) in LOD cloud, which has offered more information such as geographical information in regarding to the place by other one's efforts. We then filter the enriched geographical information provided by only DBpedia in the LOD cloud. The enriched station's geographical information is shown in Table II, which additionally includes the city location, latitude/longitude coordinate of the station, and the found entity in DBpedia.

C. Semantically Acquiring Atmospheric Features

As Table I indicates, only a limited number of meteorological variables can be employed as features for rainfall detection, despite the fact that some variables may be considered noisy features for the task. For a machine learning task, the predicting result may be highly dependent on the number of effective features fed into the models. Therefore, we conduct the experiment with the atmospheric variables data located in Link-Climate KG as external resources and premise that the some of the atmospheric variables can improve the rainfall detection task if they can accompany NOAA climate data as the data input for machine learning models. To accomplish this, we recap the power of the ontology as stated in Section III-A and formulate a SPARQL query as seen in Listing 3 to find available sensors in the same city of the NOAA station and their associated daily atmospheric data. During this step, the geographical information obtained from LOD cloud provides the necessary city location information (Dublin) to locate the air sensors Table III. To be short, we directly give the resulted Table IV, which has merged NOAA climate data and atmospheric data. In the following subsections, we demonstrate that the certain atmospheric variables have a significant positive impact on NOAA daily climate data w.r.t. the improvement of the rainfall detection task.

V. BOOSTING RAINFALL DETECTION WITH LINK-CLIMATE

We consider NOAA climate and Link-Climate atmospheric variables as multiple time variables that affect each other, and describe the rainfall detection as a binary multivariate time-series classification problem to which the solution should predict whether the rainfall occurs or not on a future day. The probability threshold is set to be 0.5 above which we claim the rainfall will happen on a future day. Let $\boldsymbol{x}^t$, $\boldsymbol{x} \in \mathcal{R}^N$ be an n-dimensional vector at time step t. $\boldsymbol{x}^t = (x_1^t, x_2^t, \ldots, x_N^t)$, where x_n^t denotes an NOAA climate variable or Link-Climate atmospheric variable at time step t. $y \in \{0, 1\}$ denotes the binary label and $y = 1$ (PRCP > 0) means that rainfall occurs. Hence, a formal definition of rainfall detection on time step t based on the observations of previous k time steps can be formulated as follows:

$$\boldsymbol{P}^t(y = 1 | (\boldsymbol{x}^{t-1}, \boldsymbol{x}^{t-2}, \ldots, \boldsymbol{x}^{t-k})).$$

In practice, we choose $k = 2$ as the number of preceding time step count. Given the aforementioned CA context and the merged datasets as Table IV of various Dublin's air sensors, we start by examining how the sensor distance from the NOAA

[19] [Online]. Available: https://www.dbpedia-spotlight.org/

TABLE IV
SAMPLE OF NOAA'S WEATHER DATA COMBINED WITH ATMOSPHERIC VARIABLES FROM LINK-CLIMATE

	created_at	Pressure	Humidity_%	PRCP	SNWD	TAVG	TMAX	TMIN
0	2019-06-01	1013.97	57.95	1.2	0.0	13.3	20.0	7.5
1	2019-06-02	1001.23	51.37	0.0	0.0	14.9	19.3	13.6
...	...	...	...	...	...	...	...	...
366	2020-06-01	1023.08	42.96	0.0	0.0	14.4	21.1	9.3

station contributes to the NOAA daily summary data for rainfall detection.

```
query_sensor = """
BASE
↪ <http://jresearch.ucd.ie/climate-kg/purpleair/>
PREFIX cap: <http://jresearch.ucd.ie/climate-kg/ca⌋
↪ /property/>
PREFIX sosa: <http://www.w3.org/ns/sosa/>
PREFIX ssn: <http://www.w3.org/ns/ssn/>
PREFIX dbr: <http://dbpedia.org/resource/>

SELECT ?sensor ?atm_var ?atm_obs ?val WHERE {
  ?atm_obs sosa:madeBySensor ?sensor ;
           sosa:hasSimpleResult ?val ;
           sosa:observedProperty ?atm_var .
  ?atm_var ssn:isPropertyOf <Atmosphere> .
  ?sensor cap:isLocatedIn""" + "<" +
↪ df_link_station['in_city'].astype(str).iloc[0]
↪ + "> .}"

df_sensors =
↪ endpoint_wrapper(endpoint=link_climate,
↪ query=query_sensor)
```

Listing 3. SPARQL query that finds sensors' atmospheric observations in Dublin from Link-Climate KG; `df_link_station` denotes Table II.

A. *Determining the Importance of Sensor Distance With the Boruta Algorithm*

The Boruta algorithm [23] is a highly successful way for selecting features in the machine learning field. Its approach is illustrated as: first, it adds scrambled duplicates of all features as unpredictability to the dataset. Then, it uses a feature importance measure such as mean decrease accuracy [24] to this enlarged dataset (original features + shadow feature) to train an RF classifier. The Boruta algorithm analyzes each cycle for a higher priority feature than the best of its shadow features and deletes elements that are considered highly irrelevant.

Given the algorithm, the proposed strategy to determine most important sensor can be summarized as a two-stage approach. In the first stage, we employ the Boruta algorithm independently on two groups of variables sorted from Table IV: 1) NOAA climate variables ("PRCP," "SNWD," "TMAX," "TMIN," and "TAVG") and 2) NOAA climate variables plus Link-Climate atmosphere variables, for each sensor. Three sensors at varying distances from the NOAA's Dublin Phoenix Park station are compared in group 2: the closest sensor is "sensor 26695, 3.7 km," the middle sensor is "sensor 59111, 8.0 km," and the furthest sensor is "sensor 91889, 18 km." This stage is to select the most important variables for the following preliminary rainfall detection experiments in the second stage. Once the "important features" are selected for NOAA station and each sensor, we then apply a range of frequently used classifiers to two groups of "important features" in order to perform compact classification tasks (without optimizing the models) for rainfall detection (see Table V for one classification result collection for sensor "sensor 59111"). Finally, we subtract the classification results for NOAA features from the classification results for each classifier for each air sensor in the second group in order to generate a statistic on the differences between each air sensor's data that improves rainfall detection performance. The statistic for each sensor's contribute to rainfall detection in addition to NOAA is shown in a box plot in Fig. 4. According to the box plot, only sensor "s26695" (i.e., the nearest) is able to enable most (over 75%) of the classifiers to achieve a positive impact on the rainfall detection for NOAA data. Moreover, the positive impact decreases as the sensor distance increases and eventually tends to converge on zero when the sensor is far from the NOAA station. Therefore, we conclude that the nearest sensor makes the most significant contribution to the NOAA data in terms of data usability for rainfall detection.

TABLE V
EXAMPLE OF RESULTS GIVEN BY THE CLASSIFIERS

	Accuracy	ROC AUC	F1 Score
Model			
SVC	0.69	0.71	0.70
Perceptron	0.69	0.70	0.69
LGBMClassifier	0.69	0.69	0.69
LabelSpreading	0.68	0.69	0.68
AdaBoostClassifier	0.67	0.68	0.67
LabelPropagation	0.66	0.68	0.66
BernoulliNB	0.65	0.67	0.65
LogisticRegression	0.69	0.67	0.68
PassiveAggressiveClassifier	0.63	0.66	0.62
XGBClassifier	0.64	0.66	0.65
KNeighborsClassifier	0.65	0.65	0.65
ExtraTreeClassifier	0.63	0.65	0.63
NuSVC	0.62	0.63	0.62
NearestCentroid	0.59	0.62	0.58
DecisionTreeClassifier	0.60	0.62	0.60
ExtraTreesClassifier	0.60	0.61	0.61
RandomForestClassifier	0.60	0.61	0.61
BaggingClassifier	0.60	0.61	0.61
GaussianNB	0.56	0.61	0.54
QuadraticDiscriminantAnalysis	0.56	0.61	0.54
DummyClassifier	0.57	0.56	0.57
RidgeClassifier	0.59	0.50	0.43
RidgeClassifierCV	0.59	0.50	0.43
SGDClassifier	0.59	0.50	0.43
CalibratedClassifierCV	0.59	0.50	0.43
LinearDiscriminantAnalysis	0.59	0.50	0.43
LinearSVC	0.59	0.50	0.43

TABLE VI
NOAA WEATHER DATASET ENHANCED WITH LINK-CLIMATE ATMOSPHERIC VARIABLES

	Humidity(t-2)	Pressure(t-2)	PRCP(t-2)	TAVG(t-2)	Humidity(t-1)	Pressure(t-1)	PRCP(t-1)	TAVG(t-1)	Rain(t)
2	57.95	1013.97	1.2	13.3	51.37	1001.23	0.0	14.9	yes
3	51.37	1001.23	0.0	14.9	51.49	1006.51	9.3	11.5	yes
...	...	...	...	...	...	...	...	...	...
366	37.19	1020.37	0.0	14.9	39.92	1020.93	0.0	14.6	no

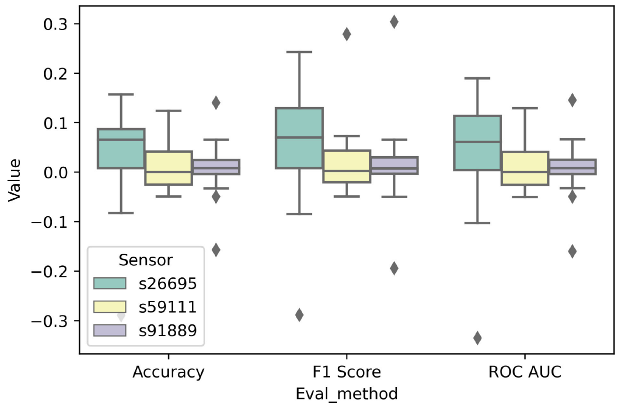


Fig. 4. Classification results statistics for sensors.

TABLE VII
PERFORMANCE EVALUATION OF RAINFALL DETECTION ON TWO DATASETS: 1) NOAA WEATHER DATASET AND 2) NOAA WEATHER DATASET ENHANCED WITH LINK-CLIMATE ATMOSPHERIC VARIABLES

Models	Datasets	Precision	Recall	F1-score	Accuracy
RF	NOAA	0.66	0.63	0.65	0.60
	NOAA + Link Climate KG	**0.76**	**0.80**	**0.78**	**0.74**
SVC	NOAA	0.65	0.73	0.69	0.61
	NOAA + Link Climate KG	**0.69**	**0.83**	**0.76**	**0.69**
KNN	NOAA	0.66	0.52	0.58	0.56
	NOAA + Link Climate KG	**0.77**	**0.62**	**0.69**	**0.67**

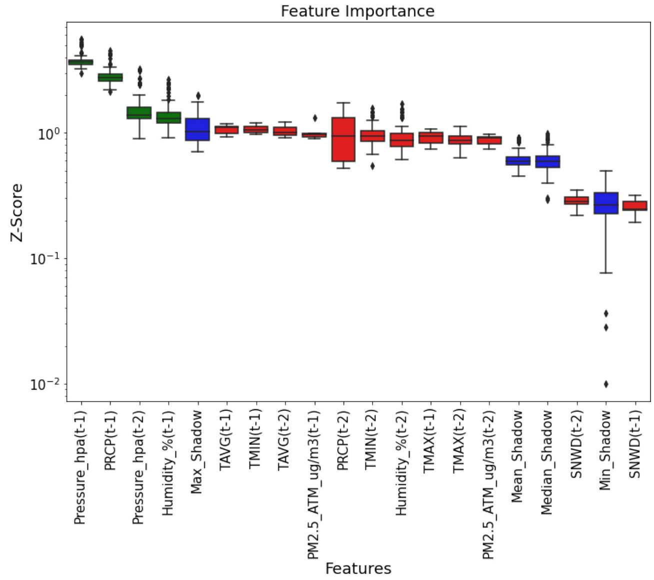


Fig. 5. Feature importance ranking; important features, unimportant features, and shadow feature statistics are shown in green, red, and blue, respectively.

B. Analysis of the Atmospheric Features on Rainfall Detection

This section is to demonstrate in depth the positive effects of Link-Climate atmospheric variables on enhancing NOAA climate data for rainfall detection. In Fig. 5, recapping the Boruta algorithm's approach in Section V-A, we directly provide the complete feature importance result of the combination of NOAA and Link-Climate atmospheric variables for rainfall detection using the sensor nearest to the NOAA station. The most critical features for rainfall detection, in order of importance, are "Pressure," "PRCP," "Humidity," and "TAVG" (closest to the "Max_Shadow" statistics). Following that, we compare the machine learning performance disparities in two situations based on the data's various sources: 1) case 1: NOAA climate variables, i.e., "PRCP" and "TAVG"; and 2) case 2: case 1 with additional Link-Climate atmospheric data—"Pressure" and "Humidity." The sorted training set for case 2 can be seen in Table VI. To construct the training and testing sets, we divided the data into two-thirds for training and one-third for testing, with the ratio of distinct labels being constant across the training and test sets. The training and testing sets are prepared in the format in line with Table VI.

C. Benchmarking With Machine Learning Approaches

The comparisons across different datasets are made on machine learning models, namely, RF, SVC, and KNNs, all of which are excellent at training with a minimal amount of data. These machine learning models are directly implemented using `sklearn` library of `Scikit-Learn` Python machine learning pipeline. In Fig. 6, we draw a group of confusion matrix pictures to show each model's learning performance across different datasets. The pictures lined in upper row are models applied on only NOAA data, and the lower row pictures are models applied on NOAA and Link-Climate KG data. As seen in Fig. 6, the recall rates for rainfall and nonrainfall predictions are significantly higher in the case of models applied on the NOAA and Link-climate data. To describe the performance more comprehensively, we direct readers to Table VII where more detailed criteria, such as precision, F1-score, and accuracy, are given for rainfall detection (label "yes"). It is clear to see that all of three models have higher performance in case 2, where Link-Climate KG data are used to enhance the original CSV-formatted NOAA data (highlighted with bold numbers). Especially, for F1-score, the increase is approximately as high as 10% on average for three machine learning models.

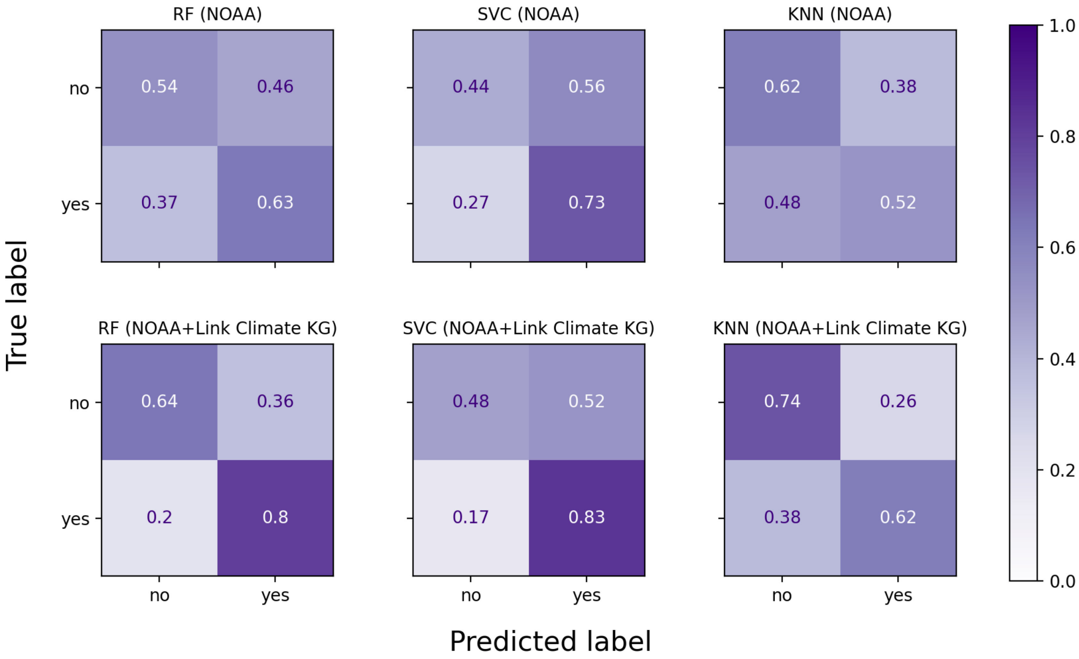


Fig. 6. Comparison of the performance of different machine learning models using NOAA data with and without Link-Climate KG data.

VI. Conclusion

In this article, we demonstrated a workflow that utilizes Link-Climate KG to improve a given tabular dataset in this study. The advantage of a KG that adheres to link data standards is that it can be accessible online through HTTP and readily obtained for on-demand and programmable data requests from clients. Because the majority of current machine learning tasks are performed on a fixed dataset, it is difficult to update the local data due to the bulked data associated with the fixed data scheme. However, with Link-Climate, the data structure is built using ontology, which allows for more flexibility in terms of acquiring multisource data. Despite the adaptability, worry over the increasing complexity and deteriorating quality of linked data has increased lately. The Link-Climate KG makes use of a KG portal to provide users with instructions to the KG manipulations. On the other hand, KG data are often downloaded beforehand as data dumps for machine learning researchers to do a range of tasks, which requires significant data preprocessing effort. The data dumps do not adhere to the linked data standards and, therefore, miss the benefits of linked data, which may provide them with on-demand and even live data. Thus, it should be a primary objective of linked data researchers to provide a simple method for bridging the divide between the KG and the contemporary machine learning environment to allow machine learning algorithms to be benefited with the linked data. To introduce linked data into popular machine learning pipelines, we need to address the major problem in regarding to the data exchange between graph data and tabular data for machine learning input. We propose using `kgextension` to bridge this gap in the climate domain using our Link-Climate KG. The proposed methodology allows users to convert KG data to PyData in such a way that popular machine learning pipelines, such as `Scikit-Learn`, may be used to analyze the KG data conjointly with local tabular data in a programmable manner. In addition, tabular data normally do not represent real-world things. It is sometimes advantageous to supplement the tabular data with additional data by using certain helpful entity link technologies to locate and connect the tabular-data-element-related real-world entities. A key milestone in our study that demonstrates this benefit is the DBpeida's Spotlight linker, which enables us to identify the geographical locations of NOAA climate observation stations through LOD cloud. The city information and longitude/altitude data may help simplify the SPARQL searches that are sent to our Link-Climate KG. Finally, we show that Link-Climate KG data complements NOAA data by significantly improving the performance of machine learning algorithms for rainfall detection. The data from Link-Climate KG's atmosphere monitoring system may also be utilized to investigate the potential of improving other climate-related activities.

In the future, we will work on developing an advanced pipeline with a graphical user interface to assist users in augmenting KG data for machine learning applications. We want to incorporate more data domains for climate research into the current Link-Climate KG, including remote sensing and transportation data. In addition, we place a high premium on sophisticated semantic web technologies, such as GeoSPARQL and temporal RDF, which together may significantly improve the performance of spatial and temporal computations simply by using high-level SPARQL queries.

References

[1] M. Rousi *et al.*, "Semantically enriched crop type classification and linked earth observation data to support the common agricultural policy monitoring," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 529–552, 2021.

[2] J. Wu, F. Orlandi, D. O'Sullivan, and S. Dev, "An ontology model for climatic data analysis," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 5739–5742.

[3] J. Wu, H. Chen, F. Orlandi, Y. H. Lee, D. O'Sullivan, and S. Dev, "An interoperable open data portal for climate analysis," in *Proc. IEEE USNC-URSI Radio Sci. Meeting/AP-S Symp.*, 2021, pp. 104–105.

[4] S. F. Pileggi and S. A. Lamia, "Climate Change TimeLine: An ontology to tell the story so far," *IEEE Access*, vol. 8, pp. 65294–65312, 2020.

[5] D. Surya, G. Deepak, and A. Santhanavijayan, "Ontology-based knowledge description model for climate change," in *Intelligent Systems Design and Applications*. Berlin, Germany: Springer, 2021, pp. 1124–1133.

[6] G. Cheng, X. Xie, J. Han, L. Guo, and G.-S. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 3735–3756, 2020.

[7] J. Wu, F. Orlandi, T. Alskaif, D. O'Sullivan, and S. Dev, "Ontology modeling for decentralized household energy systems," in *Proc. IEEE Int. Conf. Smart Energy Syst. Technol.*, 2021, pp. 1–6.

[8] P. Yue, C. Zhang, M. Zhang, X. Zhai, and L. Jiang, "An SDI approach for big data analytics: The case on sensor web event detection and geoprocessing workflow," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 10, pp. 4720–4728, Oct. 2015.

[9] Z. Miao *et al.*, "Integration of satellite images and open data for impervious surface classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 4, pp. 1120–1133, Apr. 2019.

[10] C. Bizer, T. Heath, and T. Berners-Lee, "Linked data: The story so far," in *Semantic Services, Interoperability and Web Applications: Emerging Concepts*. Hershey, PA, USA: IGI Global, 2011, pp. 205–227.

[11] T. H. V. M. Moura, C. A. Davis, Jr., and F. T. Fonseca, "Reference data enhancement for geographic information retrieval using linked data," *Trans. GIS*, vol. 21, no. 4, pp. 683–700, Aug. 2017.

[12] J. Debattista, C. Lange, S. Auer, and D. Cortis, "Evaluating the quality of the LOD cloud: An empirical investigation," *Semantic Web*, vol. 9, no. 6, pp. 859–901, Jan. 2018.

[13] J. Wu, H. Chen, F. Orlandi, Y. H. Lee, D. O'Sullivan, and S. Dev, "Automated climate analyses using knowledge graph," in *Proc. IEEE USNC-URSI Radio Sci. Meeting/AP-S Symp.*, 2021, pp. 106–107.

[14] J. Wu, F. Orlandi, D. O'Sullivan, and S. Dev, "Detecting rainfall events leveraging climate knowledge graphs," in *Proc. Photon. Electromagn. Res. Symp.*, 2021, pp. 2336–2341.

[15] S. Bhatt, A. Sheth, V. Shalin, and J. Zhao, "Knowledge graph semantic enhancement of input data for improving AI," *IEEE Internet Comput.*, vol. 24, no. 2, pp. 66–72, Mar./Apr. 2020.

[16] L. Li, H. Yang, Y. Jiao, and K.-Y. Lin, "Feature generation based on knowledge graph," *IFAC-PapersOnLine*, vol. 53, no. 5, pp. 774–779, Jan. 2020.

[17] Z. Hu, X. Ma, Z. Liu, E. Hovy, and E. Xing, "Harnessing deep neural networks with logic rules," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Aug. 2016, pp. 2410–2420, doi: 10.18653/v1/P16-1228.

[18] Z. Lei *et al.*, "A novel data-driven robust framework based on machine learning and knowledge graph for disease classification," *Future Gener. Comput. Syst.*, vol. 102, pp. 534–548, Jan. 2020.

[19] K. M. Annervaz, S. B. R. Chowdhury, and A. Dukkipati, "Learning beyond datasets: Knowledge graph augmented neural networks for natural language processing," in *Proc. Conf. North Amer. Chapter Assoc Comput. Linguistics: Hum. Lang. Technol.*, Jun. 2018, pp. 313–322.

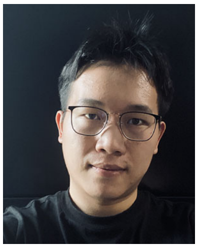
[20] A. Mohamed, G. Abuoda, A. Ghanem, Z. Kaoudi, and A. Aboulnaga, "RDFFrames: Knowledge graph access for machine learning tools," *VLDB J.*, vol. 31, pp. 321–346, 2022.

[21] T.-C. Bucher, X. Jiang, O. Meyer, S. Waitz, S. Hertling, and H. Paulheim, "scikit-learn pipelines meet knowledge graphs," in *The Semantic Web: ESWC 2021 Satellite Events*. Berlin, Germany: Springer, 2021, pp. 9–14.

[22] C. Buil-Aranda, M. Arenas, O. Corcho, and A. Polleres, "Federating queries in SPARQL 1.1: Syntax, semantics and evaluation," *J. Web Semantics*, vol. 18, no. 1, pp. 1–17, Jan. 2013.

[23] M. B. Kursa and W. R. Rudnicki, "Feature selection with the Boruta package," *J. Statist. Softw.*, vol. 36, pp. 1–13, Sep. 2010.

[24] Z. Najafi, H. R. Pourghasemi, G. Ghanbarian, and S. R. F. Shamsi, "Identification of land subsidence prone areas and their mapping using machine learning algorithms," in *Computers in Earth and Environmental Sciences*, H. R. Pourghasemi, Ed. Amsterdam, The Netherlands: Elsevier, 2022, ch. 39, pp. 535–545.



Jiantao Wu received the M.Sc. degree from University College London, London, U.K., in 2017. He is currently working toward the Ph.D. degree under the joint supervision of Prof. Soumyabrata Dev with the School of Computer Science, University College Dublin (UCD), Dublin, Ireland, and Dr. Fabrizio Orlandi and Prof. Declan O'Sullivan with Trinity College Dublin, Dublin.

Prior to being with UCD, he had been a Software Engineer for two years with China Electronics Technology Group Corporation, Beijing, China, until 2019. His research interests include knowledge graphs, machine learning, and sensor data processing.

Fabrizio Orlandi received the M.Eng. (with Hons.) degree in computer engineering from Universit´ di Modena e Reggio Emilia, Modena, Italy, in 2008, and the Ph.D. degree in computer science from the National University of Ireland Galway, Galway, Ireland, in 2014.

He is a Marie Skłodowska-Curie EDGE Fellow with Trinity College Dublin, Dublin, Ireland. He currently leads the project "DynamoKG"—exploring dynamic and uncertain facts in knowledge graphs. He is also involved in the "Beyond 2022" project as a Senior Researcher working on knowledge graphs for Irish History. Previously, since 2018, at the ADAPT SFI Research Centre, he has also led the research agenda of two industry projects with Wolters Kluwer and Huawei. Prior joining ADAPT, with Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany, he had the role of coordinator in the European Commission (EU) H2020 OpenBudgets.eu project and contributed to large European and industry research projects, such as BigDataOcean.eu and SLIPO.eu. In his research areas, he has experience on foundational and applied research on both EU-funded and industry projects. His research interests include knowledge graphs, linked open data, knowledge representation, semantic web, and the application of semantic technologies to different domains, such as social media, cultural heritage, and law and open data.

Declan O'Sullivan received the B.A. (Mod), M.Sc., and Ph.D. degrees in computer science from Trinity College Dublin, Dublin, Ireland, in 1985, 1988, and 2006, respectively.

He is currently a Professor of computer science with the School of Computer Science and Statistics, Trinity College Dublin (TCD), Dublin, where he is also a Co-Applicant Principal Investigator with the ADAPT SFI Research Centre. Since joining TCD from industry in 2001, he has established himself as an International Research Leader in his field authoring more than 260 scientific peer-reviewed papers and international journals. He is a member of three journal editorial boards. He has more than 12 chair roles in IEEE and IFIP conferences over the years. His research interests include semantic web, linked open data, knowledge graphs.

Mr. O'Sullivan has received competitive research funding as Principal Investigator and Co-Principal Investigator of approximately 7.8 million euros. He has received funding across a range of funding programs: European Commission (H2020 and Marie Curie); Science Foundation Ireland (FAME, CNGL, and ADAPT); HEA PRTLI (NEMBES and TGI); and from industry: Huawei, Accenture, Ericsson, Nokia Bell Labs, Ordnance Survey Ireland, and Central Statistics Office. He was elected a fellow in TCD in 2019 in recognition for the quality of his contributions.

Enrico Pisoni received the M.Sc. degree in environmental engineering from the Politecnico di Milano, Milano, Italy, in 2002, and the Ph.D. degree in information engineering from the University of Brescia, Brescia, Italy, in 2007.

He is currently a Scientific Officer with the European Commission Joint Research Centre, Ispra, Italy. He contributed to the Regional Integrated Assessment Tool project, developing a decision support system for air quality planning, and is at the moment mainly involved in the development, maintenance, and application of the Screening for High Emission Reduction Potential on Air integrated assessment model. His main research interests include integrated assessment modeling, surrogate modeling, and optimization techniques.

Soumyabrata Dev (Member, IEEE) received the B.Tech. degree (*summa cum laude*) from the National Institute of Technology, Silchar, India, in 2010, and the Ph.D. degree from Nanyang Technological University, Singapore, in 2017, both in electronics and communication engineering.

He is currently an Assistant Professor with the School of Computer Science, University College Dublin, Dublin, Ireland, where he is also an SFI Funded Investigator with the ADAPT SFI Research Centre. In 2015, he was a Visiting Student with the Audiovisual Communication Laboratory, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. He has authored more than 120 publications in leading journals and conferences. His research interests include remote sensing, statistical image processing, machine learning, and deep learning.