



Towards Increased Understanding of Open Data Use for Software Development

Maciej Grzenda¹ · Jaroslaw Legierski²

Published online: 22 November 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

The way open data resources of varied type and volume are used by software applications remains only partly known. In this study, following Cross-Industry Standard Process for Data Mining, we propose a methodology for collecting and analyzing access data describing the use of open data resources by individual software applications. The methodology includes novel categorization of the data collected at an exposition portal providing access to underlying open data portals and third-party services. Furthermore, it enables research into the use of both different open data resources and resource groups such as Big Data resources for software development. We apply the methodology to analyze the re-use of open urban data during reference software development events. The identification of open data use by individual applications is largely improved compared to baseline scenario, as shown by numerical indicators including F_1 measure. Insight into re-use of data streams and actual development time is obtained.

Keywords Open data · Application development · Application programming interface · Big data

1 Introduction

The growth of open data movement results in a major increase in the availability of open data resources. Some national-level endeavors reach the level of thousands of open data sets, as in the case of the Canadian federal government and <http://open.canada.ca> open data portal. The efforts of central governments are accompanied by local governments (Chatfield and Reddick 2017) and individual cities (Pereira et al. 2017; Thorsby et al. 2017) gradually opening their data sets and increasing the value of their offerings.

These efforts are partly driven by indices describing the state of open government data publication such as Global Open Data Index (Open Knowledge International 2015). Still, one of the key issues remaining is to ensure that all data that should be open is made open and current (Cowan et al. 2014). In particular, the question arises as to whether the data needed for application development are made available and whether the way it is done satisfies the needs of the software development community. Among other organizations, the European Commission is focusing on generating value through the re-use of public sector information, and emphasizes the significant potential for public data re-use in new products and services (European Commission 2016).

Unfortunately, which data sets are used for application development remains largely an open issue. This question, similarly to the question of what kind of applications are developed has recently been proposed as a part of a new research agenda (Thorsby et al. 2017). In spite of major expectations regarding the transformative potential of open data, the understanding of and the empirical data describing open data use for application development remain limited.

At least two factors seem to contribute to this phenomenon. First of all, many governments adopt a ‘data over

✉ Maciej Grzenda
M.Grzenda@mini.pw.edu.pl

Jaroslaw Legierski
jaroslaw.legierski@orange.com

¹ Faculty of Mathematics and Information Science,
Warsaw University of Technology, ul. Koszykowa 75,
00–662 Warszawa, Poland

² Orange Polska S.A., IoT and Advanced Technologies,
ul. Wolumen 11, 01-912 Warszawa, Poland

the wall' approach to open data provisioning (Sieber and Johnson 2015) i.e. concentrating on making data available rather than proactive engagement in their use, including the analysis of the way the data is used. Equally importantly, the principle of not tracking open data access performed by individual users is fundamental for open data. At the same time, government data includes not only tabular data, but also maps, data from variety of sensors, such as traffic sensors, location of public transport vehicles, smart meter readings, and pollution data to mention a few examples. Some of these resources are Big Data and stream data resources and require novel methods of data provisioning making high performance and high availability platforms mandatory.

The understanding of how data resources of varied categories are likely to be used by novel applications is vital to plan provisioning of not only tabular open data, but also new open data categories such as map data, location data and text data offered as both data files and online data streams including Big Data. Ideally, the ability to record and analyse the use of different open data resources by applications, while preserving the privacy of their creators could be sought. Furthermore, there is a need for empirical research identifying resources of which categories are used most often and the frequency of requests for resources and resource groups. Such data are needed to plan the development of open data portals matching their expected load and could be used for data mining projects identifying complex patterns in open data usage. Moreover, there remains a need to identify software development community needs and answer the question whether novel open data indices capturing the state of open government data publication from software development perspective are worth considering.

The purpose of this study is to propose and validate the methodology enabling a) the collection of data on open data use for software development projects, b) research into how these projects access individual open data resources and resource groups and c) the use of this research to guide the development of open data portals and initiatives such as hackathons. More precisely, to address the aforementioned need for increased understanding of actual open data use by software developers and foster the development of data portals addressing the needs of software development, we propose a research methodology of documenting and analyzing open data use, which is focused on these needs and preserves the privacy of open data users. Our methodology is gradually refined following a structured Cross-Industry Standard Process for Data Mining (CRISP-DM) (Chapman et al. 2000; Provost and Fawcett 2013). It is inspired by the experience arising from open data competitions organized under the name of BIHAPI in Poland. The methodology we propose combines

preparing an open data portfolio, classifying individual open data resources, collecting data on their use and calculating open data use indicators. Furthermore, we analyze the data from two developer competitions fulfilling these conditions, which attracted over 100 software development teams. Finally, we refine recommendations for open data provisioning and open data content classification.

The novelty of our approach is threefold. First of all, by proposing the novel methodology and the Unified Open Data Provisioning Model it includes we show how ground truth data on open data use by numerous applications can be collected. In particular, we propose and validate the use of exposition portal making available open data resources of varied categories and sharing resources from a number of underlying open data portals to collect data on open data use. We show with numerical indicators of False Positive and F_1 measure that the data collected with our methodology makes it possible to eliminate errors in open data use identification made otherwise i.e. when assuming uniform use of open data by different applications. Secondly, we propose a multi-dimensional standard of open data categorisation enabling the analysis of open data use performed at the level of open data resource groups rather than individual resources only. Last but not least, we enable empirical research into software development process involving both open data and third party services, which we illustrate with the estimation of the actual development time during reference software development competitions.

The remainder of this paper is organized as follows. The second section provides the theoretical foundation by discussing current recommendations and practices in the field of open data provisioning from an application development perspective. This is followed by the proposal of the methodology of collecting open data use indicators through developer contests, which is laid out in Section 3. The software development competitions and the data they provided are described in Section 4. Section 5 includes results of the quantitative analysis of the application development performed in these competitions and information gains arising from our methodology. This is followed by discussion contained in Section 6. Finally, conclusions and future work outlined in Section 7 complete this study.

2 Open Data Use for Software Development

2.1 Enabling Innovative Applications Through Open Data Provisioning

Open data is considered a valuable asset enabling the development of novel software applications and services based on them. As shown in the study of Swedish

IT-entrepreneurs (Lakomaa and Kallberg 2013), public open data is considered very important for many IT-startups. 82% of entrepreneurs surveyed in this study considered open data to be valuable in supporting and strengthening their business plans. Moreover, companies build their offering on the basis of open data sets. Among others, Lindman et al. (2014) analyze varied business models of pioneer entrepreneurs in the open data area. Immonen et al. (2014) show that interest in open data business is high. Zeleti and Ojo (2017) in their study on open data value capability architecture, provide an in-depth analysis of the competencies needed in Xpreso - an open data-driven private organisation in Ireland, which plays the role of both data consumer and producer and relies on open data. These studies confirm the potential economic gains arising from open data. Hence the need for analysing both open data re-use by novel software applications and increased understanding of the needs of software developers.

When analyzing the use of open data for application development, a possible solution is to perform opinion surveys on the experiences of individual actors in an open data ecosystem. This approach has been adopted *inter alia* in open data studies performed in Sweden (Lakomaa and Kallberg 2013), Finland (Lindman et al. 2014) or Brazil (Pereira et al. 2017). However, it remains impossible to analyze low level features such as the frequency of requests made to individual open data resources in this way. Such features are of major importance for understanding the similarities and differences in the way individual open data sources are used.

Government at different levels can also benefit directly from open data re-use. As an example, the winning applications of competitions organized by New York City are reported to be both innovative and cost less than applications built by the government (Masip-Bruin et al. 2013). Some other examples involving the use of open data include the development of a hybrid navigation system including indoor navigation (Zaragozı et al. 2015) or web portals analyzing real estate prices (Chen et al. 2015). To support even more active development, Charalabidis et al. (2016) have proposed a system supporting software development with open data resources.

Successful open data-based applications confirm that work on making open data more accessible for the software development community is of crucial importance. In particular, Immonen et al. (2014), while studying an open data based business ecosystem, observe that the permanency of both data sources and Application Programming Interfaces (APIs) is expected by both data providers and application developers. The ability to use and design APIs is also one of key competencies identified in a study on open data value capability architecture (Zeleti and Ojo 2017). Currently, existing data portal indices focus

on the needs of the software development community to some extent only. In particular, Global Open Data Index (GODI) (Open Knowledge International 2015), does not explicitly promote the needs of software developers by assigning points to the portals making APIs available. GODI is focused on scoring government data in categories such as open licensing, data format, up-to-date, public availability, and open access, i.e. categories crucial for all data consumers. However, a growing awareness of the varied needs of different actors in open data-based ecosystems can be observed. As an example, the Open Government Data Portal Index (OGDPI), proposed recently by Thorsby et al. (2017), even though based mostly on evaluation of open data portal features such as the number of datasets per 100 000 citizens or data visualization and data manipulation capabilities, already rewards API exposition. OGDPI raises the portal rating by 5 points (out of 100 possible) when the portal exposes APIs.

2.2 Difficulties in Identifying Open Data and APIs Use for Application Development

Studies confirm the need for open data policies which take more account of the needs of open data users and open data reuse (Zuiderwijk and Janssen 2014). Hence, in parallel to the growth of open data availability, the widespread growth of open data initiatives aiming to facilitate citizen-government interactions can be observed (Sieber and Johnson 2015). An important purpose of such initiatives is to take measures to accept and promote citizen feedback. On the other hand, many open data initiatives are still frequently oriented on the data provision process rather than data use (Zuiderwijk et al. 2015). This is in spite of the fact that open data is expected to promote innovation. As a consequence, even though empirical evidence showing the use of open data by innovative enterprises exists, the underlying process of open data use for innovation requires further research. Only some aspects of the use of open data for innovation, such as the development of a business model framework for managing open data applications (Chien-Chih 2016), have recently been addressed.

In spite of the benefits of open data creation and re-use, as observed in Zuiderwijk et al. (2014) only a limited application of theory and development of theory aimed at the understanding of the open data phenomenon is present in extant literature. This is consistent with the fact that open data initiatives are largely oriented on data provisioning, which limits the interest in the way the data is actually used. In particular, the model of open data provision as a unidirectional provisioning of the data (Sieber and Johnson 2015) is followed by many institutions. Such unidirectional data provisioning can fulfill the rules proposed by organizations such as the Open

Knowledge Foundation. Unfortunately, the authors of the summary of open data assessment frameworks (Donker and van Loenen 2017) observe that misalignment between the providers and users and between supplied and demanded data sets may follow from limited attention paid to the user perspective.

Sieber and Johnson (2015) have analyzed different levels of government engagement in open data promotion and value creation. The first of the levels, described as the ‘data over the wall’ model, relies on the assumption that the exposition of the data itself is sufficient to enable public benefits. However, one of the consequences of this model is that not much is known about the actual use of the data and the reasons for which individual data resources are consumed. As a consequence, gaining knowledge describing the process of open data use becomes difficult, if not impossible. To promote the use of open data and gain more insight into the way open data is used, some government institutions adopt the ‘open data activist’ model, proposed as another level within the same classification (Sieber and Johnson 2015). This model relies on active government participation in fostering open data use and promoting more active citizen involvement.

Apart from the ‘data over the wall’ approach, another aspect contributes to difficulties in uncovering patterns of open data use for application development. One of the fundamental assumptions for open data exposition is that data should be made available to anyone, with no requirement of registration (OpenGovDataorg 2007). This is mandatory to preserve the privacy of the citizens using the data. The no-registration requirement is commonly accepted in the open data community. In particular, in the Global Open Data Index, the number of points assigned to an open data portal is reduced if the data is available online only after registering (Open Knowledge International 2015). In line with these expectations, Thorsby et al. (2017) identified among 37 American cities only one which required an account and password to enter. Thus, this portal was not considered ‘open’.

While lack of registration and authorization simplifies access to data and preserves privacy, it makes the separation of open data access performed by data interpreters from application-based traffic on open data portals difficult or even impossible. Under the no-registration access policy, requests to use individual open data sets cannot be linked to the users or applications consuming the data. This simplifies the access to the data and protects the privacy of the users, which is fundamental for open data exposition. On the other hand, access to open data can occur due to a variety of reasons, such as the use of open data by Non-Governmental Organizations (NGOs) to monitor the actions of the authorities or by universities to use open data in research and education. Therefore, it remains

virtually impossible to identify the use of data sources by applications building novel services with open data. It is also impossible to identify the groups of data sources used by individual applications. The question arises whether the two conflicting needs of privacy protection and analyzing empirical data on open data use for application development can be addressed at the same time.

Whether to make registration a mandatory step preceding the access to open data is related to the access to open data exposed via the APIs. APIs expose open data to software applications, by providing services which can be called by the applications to query and retrieve relevant subsets of data. API-based access becomes the only choice when near-real time data streams are exposed. In particular, high volume, variety and velocity data, known as big data (Gartner 2017a), requires API-based exposition rather than file download. This is because time-consuming downloading of large files would result in major delays in the processing of open data streams such as public transport location streams, making near real-time data analysis impossible.

As far as open commercial APIs are concerned, they are frequently exposed in the RESTful model proposed by Fielding (2000). Understanding of REST APIs is also considered one of key competencies by some open-data driven organisations (Zeleti and Ojo 2017). Importantly, commercial REST APIs frequently rely on the a priori registration of API users. The registration process provides API user with an API key, which has to be presented whenever APIs of interest are accessed. This model is used by many open API providers such as Amazon (2019a), Twitter (2019), and Google (2019). One of the benefits of the use of API keys is that it allows open API platforms to tackle excessive volumes of requests generating too large a load for open data platforms. This benefit is also observed by client applications calling APIs. Should the number of requests to the API of interest be too large, one client application could slow down the entire API exposition platform and negatively affect the performance of the remaining applications in turn. An API key mechanism has been used to address this issue, as it can be used also to apply per-client throttling limits. Such limits are applied to clients using API keys in the services of Amazon (2019b), Twitter (2019) or Google (2019). The use of an API key is also important from the point of view of the security of the programmer, especially the security of messages sent from the provider to the developer, which then are better protected against being replaced by malicious software. Currently, in most cases, obtaining the API key requires providing personal information about the developer. However, obtaining the API key can also be anonymous and can be used only to distinguish between the consumers of services exposed by the API.

2.3 Categorization of Open Data Resources

Before the use of open data for application development can be analyzed, decisions on the possible components of an analytical framework have to be made. Different applications developed with open data resources are likely to use different subsets of the available data. Hence, the question arises of whether and how open data resources can be grouped into subsets sharing similar features. Thus, what is fundamental for the analysis of open data use is the categorization of open data resources. We use the term open data resource here to go beyond the traditional understanding of open data as data offered in the form of a downloadable data file. This is because open data is gradually being extended to include other forms such as data streams. Examples of data streams include (but are not limited to) news feeds from Twitter or data streams arriving from operational monitoring of physical systems (Ellis 2014).

As far as open data categorization is concerned, one of the key aspects is content area, defined as the domain the data resource comes from. The Global Open Data Index (Open Knowledge International 2015) relies on a methodology which lists several data categories such as budget and spending. However, due to the fact that the Global Open Data Index is used to assess country-level open data achievements, data categories also include categories defined at a national level such as national law. Therefore, to analyze the content of open data portals in American cities, Thorsby et al. (2017) propose a Dataset Content Index (DCI) dedicated to content scoring, which includes its own set of content areas. In this methodology, the portal is scored based on the number of data sets in each of 25 content areas defined based on the domain the data comes from, such as animals, education, real estate, or traffic.

Another feature of a data resource is whether it can be classified as a traditional data source or a Big Data resource. Big data by its definition requires innovative forms of data processing (Gartner 2017a). Despite the possibilities they offer, big data resources may not be sufficiently reused due to the technical complexity they require. It has already been observed that one of the barriers to open data adoption is lack of technical abilities (Jetzek et al. 2014). Importantly, Marz and Warren (2015) notice that traditional systems have failed to scale to Big Data. Moreover, there is a growing concern regarding the intersection of open data and big data resources. Big and Open Linked Data (BOLD) integrates open data with linked data and big data (Dwivedi et al. 2017; Janssen et al. 2017). As pointed out in Janssen et al. (2017) BOLD not only results in new opportunities, but also has a significant impact on privacy and transparency and can have a huge impact on the functioning of society. This confirms that big data resources have a particular role among other

open data resources. Furthermore, open data principles such as those expressed in Open Data Charter (2017) recommend that as far as possible data should be released in its original, unmodified form. This further promotes the provisioning of big data resources in their raw form rather than the aggregated form suitable for traditional data processing systems.

One more aspect of open data resource is the level of openness. Tim Berners-Lee proposed in 2010 a 5-star deployment scheme for Open Data (Berners-Lee 2010). One star data means that data is available under open license, but locked up in a document. Further levels add the fact that the data is available in a structured i.e. machine-readable way (2 star level), and can be processed using non-proprietary software since it is provisioned in a non-proprietary format (3 star level). The four star level adds the requirement that the (most important) data items have a Uniform Resource Identifier (URI), and can be shared on the web. Finally, the five star level means that not only are all the previous conditions met, but also the data is linked to other data. Needless to say, the level a data resource is published with has a major impact on its usability for software development. In particular, proprietary formats or data locked in scanned documents largely hinder its automated processing. Jetzek et al. (2014) identified the list of barriers to open data adoption, some of which include closed datasets, lack of data policies, incomplete data sets and their metadata. Five star level data clearly helps overcome many of these barriers.

Furthermore, a growing emphasis on real-time stream data processing has been observed in recent years. Importantly, implementation of a real-time system is within the reach of nearly any organization (Ellis 2014). In such cases, the Lambda architecture (Marz and Warren 2015) necessitates making the data available for processing in parallel with storing it. The pattern of (1) accumulating enough data, (2) publishing it, and (3) consuming it with software applications becomes inappropriate when processing latency has to be minimized. Oliver et al. (2012) observe that dedicated architecture solutions are required to deal with massive volumes of continuously flowing information.

It is important to note here that the inclusion of big data and data stream provisioning makes the planning of open data provisioning more difficult. The interest of software developers in individual open data sources including batch download and online streams becomes unclear. Moreover, the number of data resource categories requiring their own data provisioning solutions is growing. Modern open data portals offer traditional flat files in formats such as CSV, maps in raster and vector formats exposed in Web Map Service and Web Feature Services standards, or data streams exposed e.g. through RESTful APIs. This is accompanied

with text news made available via Twitter and RSS and many others.

3 Research Method

3.1 The Data Mining Process and the Role of Data Sources

Cross-Industry Standard Process for Data Mining (CRISP-DM) defined in Chapman et al. (2000) provides the codification of a data mining process based on well defined stages. Schuff et al. (2018) observe that CRISP-DM addresses business understanding and data understanding, which is the advantage of CRISP-DM. These aspects are of also particular importance for our study.

The ultimate objective of CRISP-DM is to support data analytics projects by providing a reference model specifying how to clarify business needs, understand available data, prepare it and use it for modelling purposes. In the analysed case, we use it to structure our works on identifying data of possibly high value for the modelling of open data use by software developers. Below we focus on the first three phases of CRISP-DM process, namely *business understanding*, *data understanding* and *data preparation* i.e. the phases aiming to provide data needed for data mining purposes.

Business Understanding Government data includes variety of data categories such as tabular data, but also maps offered through web services, data from sensors such as traffic sensors located at strategic country network and city streets or location data of public transport vehicles. As some of these data resources are Big Data resources they demand new platforms to store and make data available. Moreover, provisioning of on-line data streams at open data portals requires constant integration of open data portal with underlying data sources unlike in the case of the data files uploaded on weekly or monthly basis. Hence, observing whether and how data resources of different categories are used by third-party applications is a major step to understand which data should be open in online manner and whether high load of open data portal caused by it can be expected. Before large scale data provisioning this interest could be estimated during pilot activities such as hackathons.

Furthermore, it is important to identify possibly through such pilot activities which combinations of resources are used by individual applications. Otherwise, not including just one of relevant data resources in constant open data offering may make regular use of some novel applications and services impossible. In this case, the data mining techniques of frequent itemset mining can be used.

Different applications may request new data rarely (e.g. new budget data) or on regular basis aiming to get near-real-time updates (e.g. the location of public transport vehicles). In general, the more Big Data and stream data resources are open, the higher load at open data portals may occur. Hence, the need to predict the frequency of requests made to individual resources by individual applications. This can be done with machine learning techniques including regression techniques.

All the aforementioned needs require ground truth data showing which open data resources and when i.e. also how regularly and frequently are used by individual applications. Importantly, identifying the use of individual resources and groups of resources, and predicting the frequency of requests for individual resources should be possible both at the level of individual resources and resource groups. As an example, the question can be how frequently map data together with stream data is requested, as these requires map server and streaming subsystem as vital subsystems of the open data portal.

Data Understanding In line with the objective of this phase the data that could be used to address the needs identified in the previous phase has to be sought. Provost and Fawcett (2013) in their discussion of this phase of CRISP-DM observe that in some cases ancillary projects may be required to arrange data collection. Such a project is required to collect the data on open data use by individual applications. How to collect such data is a key part of our results of *Data understanding* phase in the analysed case. Due to the complexity of the data collection project, we discuss it below in Section 3.2.

Data Preparation The objective of this stage is to provide the data ready to be used for modelling purposes. While in Section 3.2 we propose how relevant data can be collected, the preparation stage aims at providing the final data on open data use. In the analysed case, in line with CRISP-DM recommendations for *Construct Data* task of this stage, we produce derived attributes i.e. the attributes developed based on open data resource, describing which groups every resource belongs to. In this way, the ability to analyse open data use not only at the level of a resource, but also at the level of content area or content type is ensured. The details of the features we propose to be derived are discussed below in Section 3.3.

3.2 Collection of Data on Open Data Use

Software development competitions, including hackathons, are a major form of promoting open data use and fostering innovative data use and reuse and clearly belong to the ‘open data activist’ model (Sieber and Johnson 2015).

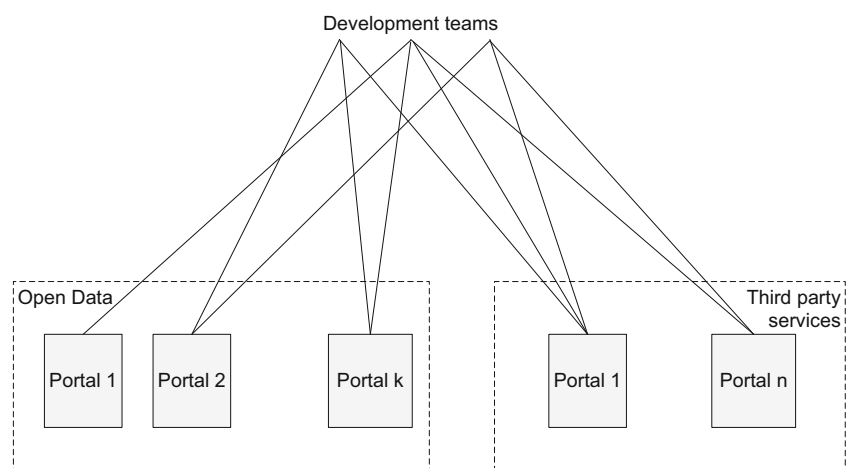
They can also provide important insights on open data use and innovation potential. This creates unique opportunities to collect opinions of open data users, in this case software developers, on the open data provisioning process. While the opinions of individual data users are frequently analyzed, the big data philosophy also suggests the investigation of *dark data* (Gartner 2017b) i.e. data typically used for immediate temporary and technical purposes only. In the analyzed case of open data use, we believe that contest logs collected by data provisioning platforms and showing the use of individual open data resources by software development teams are of particular value. Typically, development teams are able to use data sets and APIs from many portals and internet repositories to develop applications combining data and services from different sources. In some cases, the development process may rely on mashups i.e. the development of web applications that are created by combining different resources available on the web (Ordonez et al. 2015). In such cases, even if collected, the information about data usage in this model is distributed among many different open data portals and therefore difficult to obtain and analyze. As shown in Fig. 1, it is not clear for a portal administrator investigating the logs collected at the portal of interest, whether the data used from this portal has been used in combination with other data resources and third party services to develop a single application or not.

Hence, for the logs registering open data use to enable quantitative analysis of open data use for application development, extra conditions have to be fulfilled. First of all, as previously observed, the majority of open data portals rely on a no-registration policy and may even have no mechanisms to set up user accounts or manage API keys. Secondly, the data captured at a single open data portal will not show the use of open data in combination with other services. However, we believe these issues can be successfully resolved.

First and foremost, we propose to preserve the privacy of the users of open data portals and address the need of analyzing software development activities at the same time. More precisely, we propose to use software development competitions such as hackathons as a way to collect reference data on open data use. Through the knowledge they provide, open data provisioning could be adapted to the needs of software developers to unlock the potential of open data re-use. Importantly, the model we propose does not change the fundamental assumption that open data portals should offer their data under a no registration policy. We believe that the API key mechanism, which has already been accepted by the software development community, can also be applied for open data access by this community, which is what we will validate with empirical research in this study. The activities of anonymous software development teams can be monitored with the use of API keys i.e. unique identifiers related to the applications being created. This requires distribution of API keys playing the role of unique identifiers of individual applications. Importantly, such identifiers can be randomly provided to individual teams. While presenting any kind of identifiers or credentials when accessing the data is considered an obstacle by the majority of open data initiatives, it has become a de facto standard when software development involves the use of third-party services. The aforementioned examples of APIs exposed by Google or Facebook clearly confirm this observation.

Furthermore, let us observe that collection of logs is a standard procedure of software systems and takes place for a variety of reasons, resource use, performance monitoring and problem resolution being among the most important of them. Hence, both API key use and log collection are de facto standards in the IT community. Therefore, we believe data access policy could be differentiated based on the actors. In particular, application developers could be offered the use of API keys when accessing open data. To

Fig. 1 Typical mashup-based application development model



make this process fully preserve the fundamental privacy needs, let us propose the use of API keys to be considered only for the purpose of research aimed at analyzing open data use and to be implemented via a dedicated exposition portal. Such selective use of API keys we believe can be applied during software development competitions attracting multiple stakeholders or on a laboratory scale of single or multiple teams. In terms of the additional overhead the use of API keys creates, let us observe that the time needed to get an API key, which is typically in the range of a few minutes, is negligible compared to the software development time. The development time is at least several hours, which directly follows from the duration of typical hackathons. Hence, unlike in the case of data download by data interpreters, the use of API key does not contribute significantly to the overall time needed to re-use the data.

Secondly, let us propose using an *exposition portal* in front of individual open data portals. Such a portal, taking the form of a dedicated software system, makes it possible to provide the registration services for individual software development teams, but also provides a unique chance of exposing APIs served by multiple open data portals through a unified exposition portal. This simplifies the development of applications exploiting the services and data sets made available by multiple organizations. Moreover, no changes to underlying open data portals aimed at the inclusion of authorization services are required. Not surprisingly, the major role of the exposition portal is to collect logs on the use of individual open data resources by individual development teams, irrespective of whether such functionality can be directly offered by the underlying open data portals or not. The overall architecture of the model proposed in this study is provided in Fig. 2. This architecture will be referred to as the Unified Open Data Provisioning Model (UODPM) in the remainder of this study. The key objective of the exposition portal is to make available underlying services to software development

teams. This is combined with the authorization process and the collection of access logs showing the use of open data resources. Importantly, let us emphasize that we propose API key values to be used to identify the open data needs of different applications, whereas the identity of the teams using them does not have to be collected. More precisely, pre-defined API keys can be distributed among the teams attending research-oriented software development activities grouping multiple development teams.

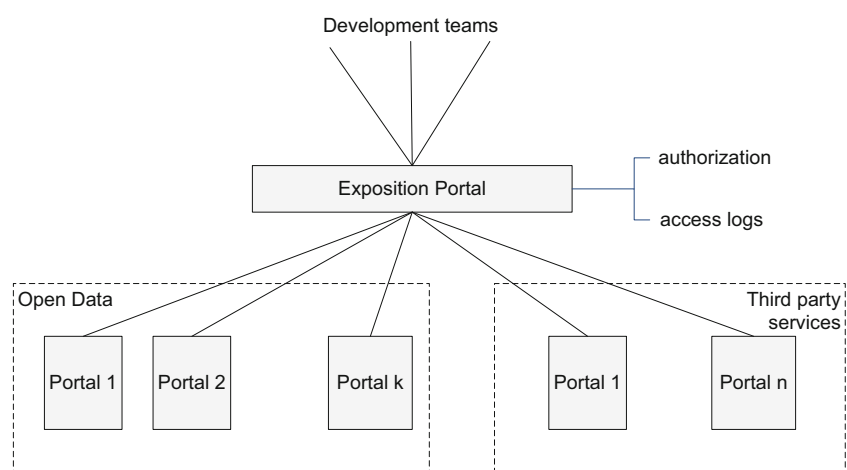
Moreover, let us propose that the unified service provisioning model could also include third-party services, including services other than open data-related such as telecommunication services. Among other benefits, this will enable frequent pattern mining i.e. identifying frequently used combinations of open data resources and generic services such as locating an application user on a map. Hence, we will use the term service to refer both to an API exposing open data resources and to a third-party API.

Finally, we believe that to provide empirical evidence on open data use, some of the software development competitions can be extended to last longer than the typical weekend time frame of a hackathon, as reported *inter alia* by Chatfield and Reddick (2017). This will help to enable not only preliminary qualitative analysis of developer opinions, but also the collection of access logs showing the actual use of individual open data resources, possibly in combination with the use of other services. The time frame of a competition can be set based on the investigation of implementation time, which can be revealed by log investigation and may vary depending on factors such as the composition of the target group of the competitions.

3.3 The Classification of Open Data Resources

Once data on the use of individual open data resources becomes available, the question of aggregating the use of similar services arises. As observed above, this requires

Fig. 2 The architecture of Unified Open Data Provisioning Model



classification of individual resources. This necessitates a categorization system matching the content of open data resources and possible third-party services under consideration. We propose that the classification system in Table 1 be used for the purpose of analyzing open data use for application development. The features we propose here are intended to complement other features laid out to describe open data resources and already proposed in open data assessment frameworks aimed at all open data users, such as the features proposed in Donker and van Loenen (2017). Our list is focused on the unique needs of software development and is driven by the differences in the way data of different categories are published and processed. It includes the features already used for general purpose description of open data resources i.e. *level* of openness defined in Berners-Lee (2010) and *content area*, but also new features going beyond the perspective of actors in the open data ecosystems other than developers.

Furthermore, we propose including in the set of key features a *category* stating whether a resource can be considered Big Data or not. It follows directly from the definition of Big Data that big data resources require innovative forms of data processing, which can influence the adoption of such resources by software developers.

What is also of primary importance for software development is the *content type* of the resource. Processing text data or geo data requires dedicated techniques matching each of these categories such as natural language processing, or spatial queries, which due to their complexity go far beyond filtering and aggregation of tabular data. Hence, we can expect varied levels of interest from the software development community in using such resources,

irrespective of whether the resources under consideration belong to the same content area such as public transport or not. For this reason, we propose that content type should take the following key values: tabular data - collects data made available in the form of tables of feature values, frequently provided in the form of CSV files; and text data - the resources, composed mostly of text such as citizen suggestions for public transport improvement, expressed in natural language. Two other categories are related to maps and are divided into vector and raster (i.e. bitmap-based) maps. Other structured data is reserved for complex structured content involving hierarchies of features such as variable-length lists of office departments, including the data on the number of citizens waiting to be served. The latter content type denotes the data which due to its complex structure is not included in tables; it is instead expressed in formats such as JSON and requires IT skills to process, unlike CSV data. Finally, non-open data service is used to represent APIs not related to open data such as third-party services of social media portals. The content type values we propose above are partly related to RFC4288 standard (Freed and Klensin 2005) i.e. current best practice for media types, which initially were defined for registering media types for use in Internet mail and now serve to describe Internet content including responses generated by internet services. However, we focus on the differences in data provisioning and processing rather than in data transmission, which is why we suggest to differentiate between some of text types such as plain text and XML data. In particular, we consider XML an example of structured data even though plain text and XML can both be treated as text under RFC4288 standard.

Table 1 Resource categorization for software development analytics

Feature	Number of values	Values	Comments
Level	5	1,2,3,4,5	The levels proposed in Berners-Lee (2010)
Content area	Data dependent	Data dependent	In the case of country-wide portals Global Open Data Index (Open Knowledge International 2015) can be used. For local and/or city portals recommendations of (Thorsby et al. 2017) can be used. These taxonomies can be extended with the taxonomy of non-open data services.
Category	2	Standard, Big Data resource	(Gartner 2017a) definition of big data can be used to distinguish the two forms of data
Content type	10	tabular data, text data, vector maps, raster maps, other structured data, audio, video, image, mixed, non-open data service	Apart from tabular data offered in the form of CSV files, other content types can be present, which require their own data processing techniques.
Time dependency	3	last value, historical, mixed	Type mixed consists both of last value and historical data types
Service category	Data dependent	Open data, and possibly other service-dependent	Apart from open data other categories of services such as telco services can be used
Serving platform	Data dependent	Data dependent	One value per one underlying open data portal e.g. identifying the city portal the data comes from

Furthermore, *time dependency* feature identifies whether an open data resource makes it available to poll for the most recent known value e.g. bus location (denoted with last value), just download historical batches of data (historical) or both techniques of accessing the data are supported (denoted with both). Moreover, what directly follows from our model is the inclusion of *service category* i.e. whether the API of interest provides access to open data or third-party service. Similarly, it is natural to be able to distinguish individual underlying open data and non-open data platforms, which is the reason for including *servicing platform* feature. Finally, let us use the term *access data* to refer to the logs of the use of individual open data resources and third-party services collected at exposition portal and extended with the features listed in Table 1.

4 The Application of the Methodology to Software Development Competitions in Poland

4.1 The Reference Software Development Competitions

In line with the open data activist model, a series of application development competitions was organized in the period 2013–2016 in Poland. The primary objective of these competitions was to promote open data use in a combined community involving software developers, business and academia partners in order to not only foster software development, but also strengthen the commercial potential of the applications. All the competitions were organized by the Research and Development Center of

Orange Polska telecom in cooperation with academic and business partners, including venture capital firms willing to provide funding for the best applications developed within the competitions. The first edition started in 2013 under the name Open API Hackathon. Within this edition, only a few open data services were made available. In 2014 the contests were renamed BIHAPI and the open data offering was considerably extended.

Each competition consisted of two phases, namely the idea phase (phase 1) and the development phase (phase 2), details of which are provided in Fig. 3. During the first phase, software development teams composed of one or more developers submitted their ideas for applications based on open data and services. Every person could participate in many teams, one team was linked to one unique application. After the initial evaluation by the jury, some teams qualified for the software development phase. Since the juries were composed of academia and business members including venture capital firms, the applications combining innovative aspects with business or social potential were promoted in the evaluation process. For the same reasons, representatives of all major stakeholders of the software development community decided to participate in the competitions. In particular, the software development teams that took part in the competitions included independent software developers, startups and teams from other private sector companies interested in open data re-use. Since performing well was a way to start collaboration with the venture capital firms, a high level of engagement from the teams forming startups or interested in forming a startup with the help of the venture capital firms was observed. During the software development phase, the teams were given access to a platform exposing open data and

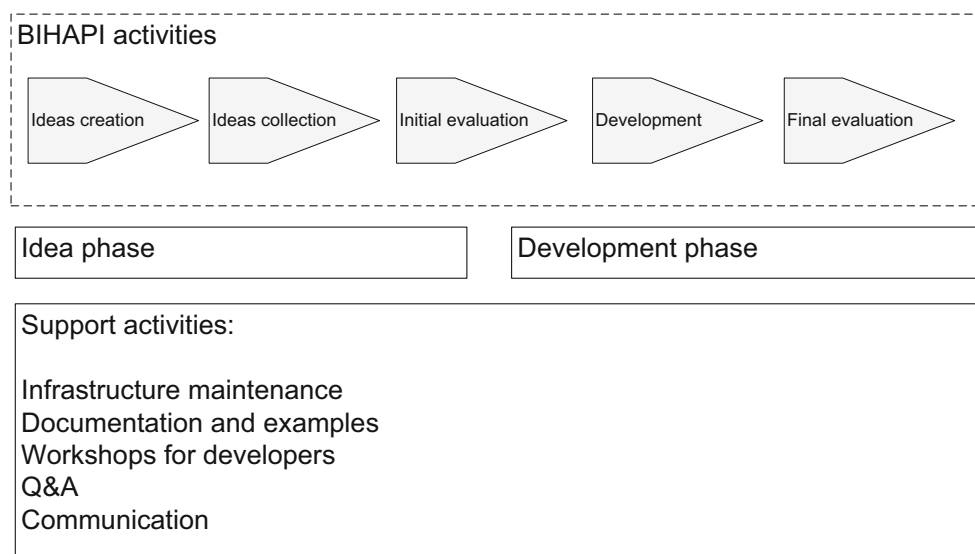
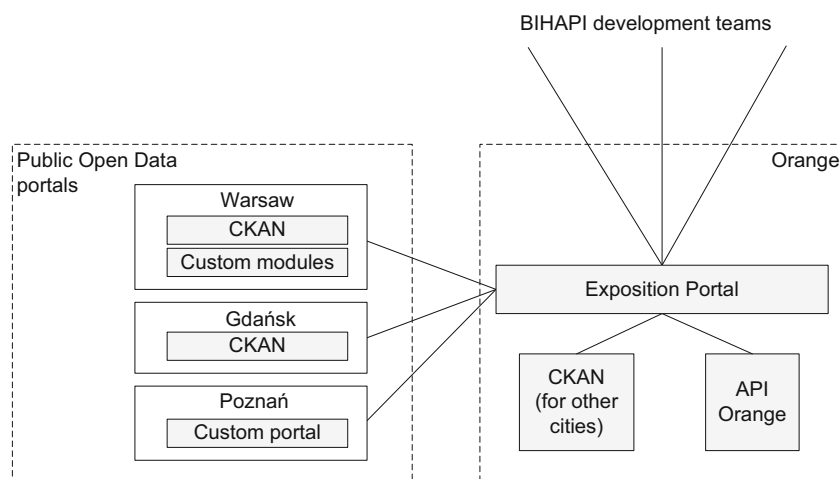


Fig. 3 BIHAPI contest timeline

Fig. 4 BIHAPI software platform



third-party APIs and developed their applications. The platform matched what we refine in this study as the Unified Open Data Provisioning Model and was based on Oracle Communication Services Gatekeeper (OCSG). In BIHAPI 2016, the platform provided software developers with a unified access to open data services from three open urban data portals and telecom services. The composition of the BIHAPI 2016 platform is shown in Fig. 4.

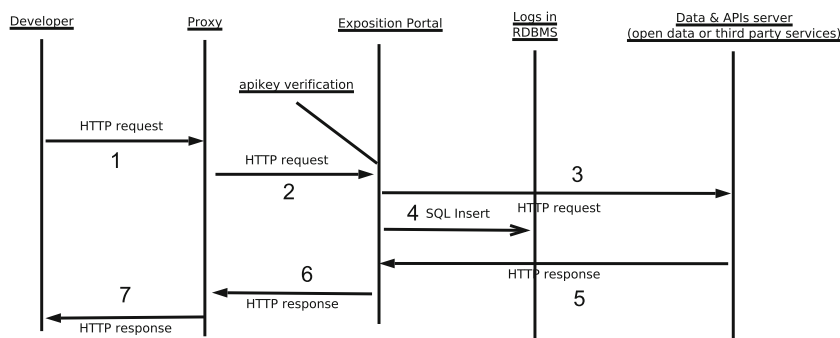
All developers taking part in the contest were informed of the possibility of using the exposition portal to access open data and open services made available for the competitions through appropriate APIs. The exposition portal was also responsible for the verification of API keys and the collection of logs. The details of this process are shown in Fig. 5. In the approach applied in BIHAPI, all resources and third party services are made available as services accessible with Hypertext Transfer Protocol (HTTP). Hence, accessing every resource, such as retrieving open data of interest, requires sending an HTTP request in the first step. Next, as shown in Fig. 5 this request is routed through a proxy server (2) to the exposition portal. The proxy server can implement security policies such as blocking requests from some network domains due to security reasons. Next, the exposition portal verifies the API key provided in the request and for API keys already registered and known

to the portal (3) directs the call to the appropriate open data platform of a participating city or service platform exposing Orange API. This step is followed by (4) creating new log entry in a database hosted by Relational Database Management System (RDBMS). The entry includes the identifier of the resource that has been requested. The response i.e. open data content or response from telecom service (5) is routed through the proxy server (6) to the client application or developer calling the service. The pattern of using exposition portal to direct requests to individual open data portals and third party services, applied in BIHAPI software system is also the basis of our general recommendations made in Unified Open Data Provisioning Model.

Importantly, the teams were also allowed to use other resources and to access the provisioned resources via direct access to underlying open data portals e.g. the portal of the City of Warsaw. In this way, all developers interested in open data re-use could participate in the competitions irrespective of which open data resources they wanted to use and how they preferred to access them.

As far as the services made available via the exposition portal were concerned, the first group included open data portals. The portals of the cities of Warsaw and Gdańsk were based on the CKAN data server (CKAN 2015).

Fig. 5 A sequence diagram for requests made to BIHAPI software platform



A dedicated CKAN server was also providing the data repository for BIHAPI partners without their own open data infrastructure at the time of the contest, including the City of Kraków and Malopolska Region. Furthermore, the unified exposition portal provided access to third-party services, namely telecom services offered by internal backend servers of Orange Polska telecom. As a consequence of applying the system documented in Fig. 4, all application developers accessing open data sources via the system could take advantage of unified access to the urban data from multiple cities. Importantly, exposed APIs included data sets from multiple cities and were provided for developer contest purposes only. In this way, the data on open data use was related to its use for innovative software development and did not include the use of the data for other purposes such as data analysis by NGOs. Furthermore, the development phase took five weeks in the 2015 edition and seven weeks in the 2016 edition of the competitions. Hence, development could be performed over substantially longer periods than the time of a typical weekend long hackathon. For this reason, the term *software development competition* rather than hackathon is used to refer to BIHAPI.

4.2 API Exposition and Data Collection

One API made available through the BIHAPI platform provided access to one open data resource or one third-party service, with the majority of APIs being of open data type. The growing number of APIs used in individual editions reflects the growth of the open data movement in Poland. It is worth noting here that the overall number of APIs exposed in these competitions is similar to or larger than 94 i.e. the median number of data sets opened by American cities as documented by Thorsby et al. (2017).

Table 2 includes the number of exposed APIs, the number of teams in both phases, and the winners in the aforementioned editions of the competitions. Moreover, the number of teams that finished the second phase successfully by submitting the application for evaluation and the number of these teams that used the exposition portal are provided in the table. It can be observed that even though participants were allowed to decide whether to use the exposition portal or not, the majority of them registered in the exposition portal, obtained API keys and used them, which has been reflected in the logs collected by the service. More precisely,

in BIHAPI 2015, 18 out of 28 teams that finished their applications, and in BIHAPI 2016, 35 out of 49 teams successfully submitting application after the second phase, used the exposition portal. This yields 64.3% and 71.4% of teams in 2015 and 2016, respectively. This confirms the hypothesis that the use of API keys is not considered an obstacle by the majority of the software development community.

As stated above, the selection of teams for the second contest phase and the selection of winners was made by the jury. The complete listing of co-organizers and jury members can be found at <http://bihapi.pl/>. For further details on the first competitions, including open data resources and telecom services provided during these editions, see our previous study (Grabowski et al. 2015). In every edition, the authors of the applications of both highest quality and deployment potential were awarded prizes. The best applications used open data from different content areas and were focused on the data of different content types (maps, text feeds, geo time series). Furthermore, the exposition of open data from multiple open data portals attracted developers interested in the re-use of open data made available by different cities. Further details on the best applications developed during BIHAPI 2016 and the data they used can be found in Table 3.

As stated above, the requests for open data resources made by the applications developed during the competitions were recorded in OCSG platform logs. The logs included API key identifiers and provided data for the investigation of open data use by individual applications.

4.3 The Categories of Open Data Resources Provided

Since a decision to use only the logs from BIHAPI2015 and BIHAPI 2016 editions was made, the question arises of what open data resources were made available in these two editions of the competitions. It is important to note that all open data resources made available for these competitions can be classified as resources of the same level on the scale laid out by Tim Berners-Lee in a 5-star deployment scheme for Open Data (Berners-Lee 2010). As no RDFs were developed for these data sets, but all of them were accessible via URIs, these resources were classified to be fourth level resources. Furthermore, due to the fact that the majority of the data was city data, the content areas,

Table 2 BIHAPI contest statistics

BIHAPI edition	total	open data	other	Phase I	Qualified for phase II	Finished phase II	Finished phase II and used exposition portal	Winners
2015	81	44	37	101	67	28	18	5
2016	14	112	37	145	58	49	35	8

Table 3 BIHAPI winners

Application	Description
Veturilo Finder	A mobile application, which help users rent city bikes in Warsaw
WhatHappened	A mobile application dedicated for Warsaw residents, which informs users about extraordinary situations such as damaged roads or pollution.
StandingInQueue	A mobile application for a queuing system installed in Warsaw administration offices
GreenWarsaw	An application allowing users to locate individual trees and shrubs and their clusters. The application can also be helpful for allergy sufferers, allowing users to check if in the immediate area is the vegetation to which they are allergic.
WillIBeInTime	A smartphone application, which displays information about the arrival of the tram, bus or train to stop selected by the user. Information is based on the actual vehicle location (GPS).
PSI - Poznań Stat Info	An application that visualizes the statistical data of the city of Poznań, grouped by category of information
BikinginKraków	An application, which enables the assessment of cycling routes for the city of Krakow. Besides navigation function, it has the ability to add guidance on the use of the routes.

proposed by (Thorsby et al. 2017) to classify the content of American city portals, were used to describe the data. The data from numerous content areas was made available for the competitions.

Moreover, telecom services exposed for the competitions included getting geolocation of a mobile phone expressed through longitude and latitude, or sending and receiving messages over the mobile phone networks including sending and receiving SMS messages by applications. Some of these services exposed low-level data on mobile telecom network such as signal strength of the mobile network observed at a mobile phone and required expert telecom knowledge to be used. Finally, all log records were described with the features listed in Table 1, which provided access data used in the remainder of this study.

5 The Analysis of Open Data Use Registered at Exposition Portal

Let us analyze the way access data can be used to investigate the way open data resources are used by application developers. One of the key aspects of the evaluation of open data portals is whether they contain resources of different content types. The question arises as to whether this opens the way to developing many applications, each consuming the data from one or two content areas, or whether application developers tend to combine many resources from multiple areas when developing a single application. It follows from the analysis that only 37.7%

of the teams relied on data from one content area only, while the same proportion of teams used the data from three content areas or more, the most extreme case using data from nine areas. This confirms the importance of making available data from as many content areas as possible so as not to hinder open data re-use for application development. Such a need for multi-category data provisioning is also promoted by the Global Open Data Index, which provides assessment of open government data publication on a per data category basis (Open Knowledge International 2015).

What is equally important when planning an open data provisioning platform is the identification of the content types that are used by software developers. Interestingly, Fig. 6 shows that it is map data rather than tabular data that was used in the largest number of projects. This suggests that the re-use of map data enables more innovative applications than the processing of the tables of records, in spite of the simplicity of the tabular format. The fact that resources of varied content types are used by application developers suggests that open data provisioning platforms to foster data re-use in innovative applications should go beyond the frequent standard of making tabular data available for download and should include other formats and services, such as map services. Moreover, many applications combined multiple content types, but never all of them. In fact, none of the content types was used in every application. This confirms that dealing with too many varied content types can be difficult or not justified. Furthermore, real time communication services exposed by telco platforms gained major interest

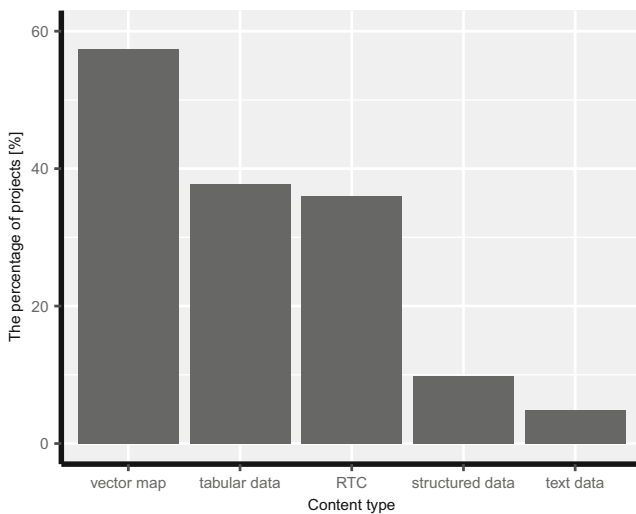


Fig. 6 The percentage of projects using resources of different content types. Based on the combined data from the 2015 and 2016 editions on the use of both open data resources and third party services

from software developers. This shows that integration of third-party services within an exposition portal provides an insight on the actual use of open data in combination with other generic services such as sending messages or displaying user location on a map.

Let us calculate numerical indicators summarising the insight into the use of data of different content types and content areas provided by the approach we propose. To do this, let us compare access data showing actual resource use with the baseline scenario. In the absence of access data, as no data showing variation in resource use among applications exist, the only assumption that can be made in baseline scenario is that every application uses the same set of resources. The question which we answer below is how large mistakes are made when uniform use of all resources by every application is predicted compared to true resource use shown by access data. We will use false positive and F_1 performance measures i.e. measures used to describe performance in data mining domain (Flach 2012; Japkowicz and Shah 2011), which are frequently used to estimate the performance of classifiers used for binary prediction. These indicators will be used to quantify errors in estimating resource use i.e. errors made when not using access data. It is important to note that these errors can be eliminated by using access data collected with the methodology proposed in this study. Formally, we treat the prediction of every resource type use as a separate classification problem and calculate averaged indicators to summarise all predictions.

Let $\Omega = \{1, \dots, R\}$ denote the set of R resources used by at least one of G teams. Furthermore, let $f_T : \Omega \rightarrow \{1, \dots, T\}$ and $f_A : \Omega \rightarrow \{1, \dots, A\}$ denote functions used in data preparation phase to assign the index of content type and content area to every resource, respectively. Let

$\mathbf{v}_g : g \in \{1, \dots, G\}$ be an R -dimensional vector describing resources used by team g such that $\mathbf{v}_{g,r} = 1$ if a team g used resource r , and 0 otherwise. Such a vector can be developed with access data. Let $\hat{\mathbf{v}}_g : \hat{\mathbf{v}}_{g,r} = 1, g \in \{1, \dots, G\} \wedge r \in \Omega$ be an R -dimensional vector denoting predicted resource use as defined by baseline scenario. Let $d(\mathbf{y}_g, x, f) = 1$ when exists $r \in \Omega : \mathbf{y}_{g,r} = 1 \wedge f(r) = x$ and 0 otherwise. In other words, d function takes the value of 1 if and only if the vector \mathbf{y}_g reports the use of at least one resource of type x e.g. at least one resource of map content type.

Next, by false positive count for content area x we define the number of teams, which contrary to baseline scenario assumptions have not used resource of content area x . The false positive count for content area x we formally define as the number of teams which have not used resource from area x i.e. $FP_A(x) = \frac{|\{d(\hat{\mathbf{v}}_g, x, f_A) : d(\hat{\mathbf{v}}_g, x, f_A) = 0, g = 1, \dots, G\}|}{G}$. Similarly, $FP_T(x)$ denotes the number of teams, which have not used resource of content type x . Under baseline scenario, $TP_A(x) = A - FP_A(x)$ and $TP_T(x) = T - FP_T(x)$. Furthermore, false negative (FN) measures are equal to 0, as for all resources we predict they will be used i.e. the situation that $\hat{\mathbf{v}}_{g,r} = 0 \wedge \mathbf{v}_{g,r} = 1$ will not happen. Finally,

let $FP_A = \frac{\sum_{x=1}^A FP_A(x)}{A} \times 100\%$ i.e. be the average false positive rate over different content areas. Similarly FP_T will be an average over different content types. Furthermore, $F_1(x) = \frac{2Prec(x) \times Rec(x)}{Prec(x) + Rec(x)}$, whereas $Prec(x) = \frac{TP(x)}{TP(x) + FP(x)}$ and $Rec(x) = \frac{TP(x)}{TP(x) + FN(x)}$ as defined *inter alia* in Flach (2012) and Japkowicz and Shah (2011). It is enough to use in the formulas above $TP_T(x)$ and $FP_T(x)$, or $TP_A(x)$ and $FP_A(x)$ to get $F_{1T}(x)$ and $F_{1A}(x)$ i.e. F_1 measure for content type and content area x , respectively. Similarly to FP_A , F_{1T} is calculated as an average of $F_{1T}(x)$ over all content types and F_{1A} as an average value of $F_{1A}(x)$ over all content areas x , respectively.

Table 4 shows both FP rates and F_1 measures for baseline scenario i.e. shows the errors made when uniform open data use is assumed rather than ground truth data, collected at exposition portal proposed in this study, is used. Particularly large false positive rates of 85.92% can be observed for content area i.e. in majority of cases the prediction that at least one resource of every content area will be used is wrong. This is also confirmed by $F_1=0.224$ i.e. very far from the optimum of $F_1 = 1$ denoting perfect prediction of

Table 4 Accuracy indicators of baseline scenario

Feature	False positive indicator	FP value	F_1 measure	F_1 value
Content type	FP_T	70.82%	F_{1T}	0.416
Content area	FP_A	85.92%	F_{1A}	0.224

open data use. The performance indicators for content type are better. Still, $FP_T = 70.82\%$ is a very high error rate showing that baseline scenario fails to predict which content types will be used by individual applications. The errors in open data prediction quantified in Table 4 are eliminated by the use of access data collected with UODPM. Hence, these errors show how significant gains arising from collecting access data from the teams using UODPM can be.

One more aspect of software development is whether software developers are interested in the use of big data resources requiring by definition innovative forms of data processing (Gartner 2017a) or data streams, which also require their own data processing techniques. Table 5 answers these questions. First of all, we may observe that the use of big open data resources is relatively limited. However, many software teams decided to use fast changing data made available as the most recent values and taking the form of data streams. Hence, the limited use of big data resources may be attributed to the fact that many data streams processed in the competitions could not be categorized as big data resources due to their limited volume, although they exhibited the features of high velocity data. Indeed, the competitions were limited by the restricted availability of open big data resources in the offerings of the contributing cities. In addition, we checked the number of projects exploiting data from more than one city. It turns out that such attempts were limited; a phenomenon which we shall investigate in the future. Among possible explanations, lack of cross-city standards making the same open data resources not only available, but also provisioned in the same format, is not the least important. Interestingly, the high proportion of projects using Real Time Communication (RTC) services reveals that many developers were interested in going beyond traditional open data services and performed mashups of services involving third party services. This suggests that the development community is willing and able to experiment with a variety of resources, but this potential is hindered by the limited availability of resources of some categories.

Last but not least, the question arises of how access data can be used to provide additional insights into the development process. One of the key questions is the time that should be allocated for software development. This can be measured by the number of distinct days during which

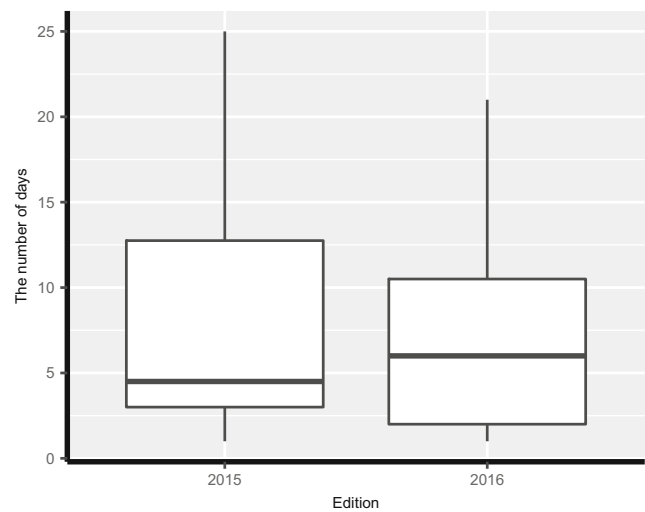


Fig. 7 The time the activity of individual teams has been registered in the logs

the activity of individual software development teams is observed in the access data, an indicator which we will refer to as *access days*. It is important to note here that this provides the lower bound for development time. In particular, some initial development efforts may not use services provided through UODPM. However, development of services consuming open data or third-party services inevitably requires accessing them. Similarly, tests of the applications, as an integral part of the development process, necessitate the use of the same services. Hence, let us estimate the time spent on application development by observing the number of access days. Figure 7 shows a box-and-whisker diagram of the number of access days. It follows from the figure that the majority of development teams worked over periods longer than two days, and performed development over up to ten days or even more. We can assume that this provided the basis for gradual refinement of not only the software code, but also the idea of the application. Furthermore, the indicators such as access days show how detailed the analysis of the software development process based on access data can be.

6 Discussion

Open data is expected to attract and engage citizens and portal users in exploring data and exploiting open data

Table 5 Frequency of using resources of different types in the applications

Feature	% of applications
Do resources used in the project include big data resources?	14.75%
Do resources used in the project include resources made available as data streams?	45.90%
Do resources used in the project include resources from more than one city?	9.83%

reuse for open services innovation (Chatfield and Reddick 2017). It is important to analyze to what extent, if any, the needs of the software development community differ from the needs of citizens or NGOs interpreting and analyzing the data. First of all, our analysis of empirical access data showing open data use for application development reveals that 45.9% of software projects developed within the competitions we analyzed used data stream resources in their projects i.e. the most recent data on the phenomenon of interest such as the most recent location of public transport vehicles. This form of accessing data is available only through APIs, which can be polled for the most recent data. In particular, making the data available in the form of files containing batches of data for individual hours or days, while convenient for citizens and persons interpreting larger portions of data, generates additional latency in data availability. This latency, while not important for off-line analysis, is not suitable or even acceptable when the objective of the software developer is to show the most recent data e.g. current bus location on the map. The analysis confirms that the needs of software developers as open data actors are largely distinct from the needs of other portal users.

Empirical data collected in the two editions of the open data competition we analyzed confirms that the registration of development teams preceding the use of the provisioning system was accepted by the majority of developers and unlocked research opportunities in this important area. In this way, data on the re-use patterns of open data becomes available, which can complement the analyses based on surveys typically used to collect the opinions and reflect the experience of open data entrepreneurs. The unified model makes it possible to capture and analyze open data use in varied software development groups. In particular, the model proposed in this study made it possible to collect data on software development activities performed by teams representing all major stakeholder groups including independent developers, startup teams and other teams from the private sector.

Furthermore, access data collected from the exposition platform can be used to identify which open data resources are used together, how frequently and for how long. These findings, while crucial in providing increased understanding of the software development process, also provide a basis for the planning of open data provisioning in the Big Data era. In particular, the interest of development communities in the processing of various content types suggests the need for analysis of open data portals in terms of content types they serve. Otherwise the potential of applications performing the fusion of varied content types may not be matched by the portals, even though their offerings in terms

of the content areas they offer may be complete. However, provisioning multiple categories of data requires dedicated solutions such as map-oriented web services or systems capable of provisioning high volume data in near real-time.

To guide the development of scalable eGovernment infrastructures addressing software development needs going beyond the download of moderately sized files, an exposition platform can be considered. By using the platform proposed in this study during events such as hackathons, long term competitions or during academic courses, the interest in individual open data resources can be assessed. This enables research of open data re-use patterns among different software development groups and in different countries. Importantly, we propose that the exposition platform should be an optional extension of open data portals, preserving their no-registration policies. Hence, any data users, including but not limited to software developers, are free to refer directly to open data portals and access the data without registration, but also without fostering research on open data re-use. Furthermore, only access data is used to analyze open data re-use patterns. In particular, startups and other participating teams are not expected to provide the source code even for research purposes.

Among key related issues, the development of a classification system used to categorize open data resources is also of major importance. The content areas and the maturity of open data defined based on a 5-star Deployment Scheme for Open Data (Berners-Lee 2010) are of clear value for software development. However, our analysis also reveals major variation in the use of open data resources of different content types. Similarly, the aforementioned significant interest in data streams show that describing open data resources with additional features is justified. The features proposed in this study reflect the complexity of data processing, which is largely different e.g. for tabular data, maps or text data. Figure 6 shows that resources of different content types are used by a largely different number of applications. Furthermore, numerical indicators, including F_1 measure also show that the assumption that all applications use open data resources of the same categories is false. High false positive rate for content type $FP_T = 70.82\%$ shows that the actual open data use is by far different from baseline uniform open data use assumption. The use of access data makes it possible to eliminate such errors by providing ground truth access data.

Finally, let us observe that the tradeoff between ease of development and the potential for the innovative data processing techniques that some data categories such as big data require are just examples showing that software developers are a distinct group of actors in an open data ecosystem.

7 Conclusions and Future Work

We have outlined a unified open data provisioning model, making it possible to register the use of individual open data resources and third party services for application development. In this way, additional insight into these software-related aspects of open data re-use has been provided. We validate the hypothesis that the open data provisioning system can be successfully combined with an authorization system, known from other API exposition platforms. In this way, access data showing which resources and services were used by individual applications can be collected.

The extraction and analysis of reference access data reveals that many developers re-use not only tabular data, but also data of other content types such as maps, data streams, and big data resources. While some aspects of open data use can be different depending on the open data offering or the composition of developer groups, the access data enabled increased understanding of software development needs.

The approach that we propose in this study contributes also to the need of aggregating the use of different resources by defining new features of major influence on software development, including content type and data latency. By introducing these features, we suggest distinguishing resources that though possibly belonging to the same content area and level, can demand largely different provisioning and processing complexity.

The processing of access data enables further analysis of open data use for software development, going beyond the scope of this study. The needs of the software development community can be analyzed using both access data and other techniques such as semi-structured reviews. One of the topics for future research could be analyzing the variation in the frequency of accessing different resources and comparing planned vs. observed use of open data resources. Furthermore, prediction of future open data use based on current access data can be attempted.

Finally, we propose that an increased awareness of the needs of software developers be built through open data policies explicitly oriented towards this target group. Furthermore, we believe that the development of novel open data indices used to measure the maturity of open data portals from the perspective of software developers could be a part of a future research agenda. A possible option is to develop a *Software Agent-Oriented Open Data Index* focused on the needs of software applications consuming and processing data rather than human users. By including a number of novel criteria, such an index could be used to guide the development of open data portals in terms of aspects such as the extent to which they provision near real-time data streams as well as periodic dumps

of data. Furthermore, it could document portal maturity on a per content type basis, which could complement existing content area-focused assessment. Refinement of best practices in this domain would help guide the efforts of open data portal administrators. This would help strengthen the potential for open data re-use for innovative application development.

Acknowledgments The authors would like to thank Sebastian Grabowski from Orange Polska for initiating, planning and supervising numerous open data initiatives in Poland, including the BIHAPI series of competitions and the Apps4Warsaw (MUNDO) project (Apps4Warsaw, 2014), which resulted in both the open data software and the data used in this study.

During the first stage of this research, Maciej Grzenda was both with Warsaw University of Technology and Orange Polska.

Funding This work was supported by the European Union's Horizon 2020 research and innovation program under grant agreement No. 688380 VaVeL: Variety, Veracity, VaLue: Handling the Multiplicity of Urban Sensors.

Compliance with Ethical Standards

Conflict of interests The authors declare that they have no conflict of interest.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Amazon (2019a). Developer's guide to the amazon appstore. http://m.amazonappservices.com/developers-guide?cmp=US_2017-00_ACH-Developer-Guide&ch=web&chlast=web&pub=web&publast=web&type=org&typelast=org.
- Amazon (2019b). Throttle api requests for better throughput. <https://docs.aws.amazon.com/apigateway/latest/developerguide/api-gateway-request-throttling.html>.
- Berners-Lee, T. (2010). Is your linked open data star? <http://www.w3.org/DesignIssues/LinkedData.html>.
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R. (2000). Crisp-dm 1.0, step-by-step data mining guide. <https://the-modeling-agency.com/crisp-dm.pdf>.
- Charalabidis, Y., Alexopoulos, C., Diamantopoulou, V., Androutopoulou, A. (2016). An open data and open services repository for supporting citizen-driven application development for governance. In *2016 49th Hawaii International Conference on System Sciences (HICSS)* (pp. 2596–2604). <https://doi.org/10.1109/HICSS.2016.325>.
- Chatfield, A.T., & Reddick, C.G. (2017). A longitudinal cross-sector analysis of open data portal service capability: The case of australian local governments. *Government Information Quarterly*, 34(2), 231–243. <https://doi.org/10.1016/j.giq.2017.02.004>. <http://www.sciencedirect.com/science/article/pii/S0740624X16302283>.

- Chen, Z.H., Tsai, C.T., Yuan, S.M., Chou, S.H., Chern, J. (2015). Big data: Open data and realty website analysis. In: 2015 8th International Conference on Ubi-Media Computing (UMEDIA), pp 84–88. <https://doi.org/10.1109/UMEDIA.2015.7297433>.
- Chien-Chih, Y. (2016). A value-centric business model framework for managing open data applications. *Journal of Organizational Computing and Electronic Commerce*, 26(1-2), 80–115. <https://doi.org/10.1080/10919392.2015.1125175>.
- CKAN (2015). Ckan the comprehensive knowledge archive network. <https://ckan.org/>.
- Cowan, D., Alencar, P., McGarry, F. (2014). Perspectives on open data: Issues and opportunities. In *2014 IEEE International Conference on Software Science, Technology and Engineering* (pp. 24–33). <https://doi.org/10.1109/SWSTE.2014.18>.
- Donker, F.W., & van Loenen, B. (2017). How to assess the success of the open data ecosystem? *International Journal of Digital Earth*, 10(3), 284–306. <https://doi.org/10.1080/17538947.2016.1224938>.
- Dwivedi, Y.K., Janssen, M., Slade, E.L., Rana, N.P., Weerakkody, V., Millard, J., Hidders, J., Snijders, D. (2017). Driving innovation through big open linked data (BOLD): Exploring antecedents using interpretive structural modelling. *Information Systems Frontiers*, 19(2), 197–212. <https://doi.org/10.1007/s10796-016-9675-5>.
- Ellis, B. (2014). *Real-Time Analytics: Techniques to Analyze and Visualize Streaming Data*, 1st edn. New York: Wiley Publishing.
- European Commission (2016). Open data policy. <https://ec.europa.eu/digital-single-market/en/open-data>.
- Fielding, R.T. (2000). REST: architectural styles and the design of network-based software architectures. Doctoral dissertation, University of California, Irvine, <http://www.ics.uci.edu/fielding/pubs/dissertation/top.htm>.
- Flach, P. (2012). *Machine Learning: the Art and Science of Algorithms that Make Sense of Data*. New York: Cambridge University Press.
- Freed, N., & Klensin, J. (2005). Media type specifications and registration procedures. <https://tools.ietf.org/html/rfc4288>.
- Gartner (2017a). Big Data definition. Gartner. <http://www.gartner.com/it-glossary/big-data>.
- Gartner (2017b). Dark data definition. Gartner. <http://www.gartner.com/it-glossary/dark-data>.
- Google (2019). Google developers website. <https://developers.google.com/>.
- Grabowski, S., Grzenda, M., Legierski, J. (2015). The adoption of open data and open api telecommunication functions by software developers. In Abramowicz, W. (Ed.) *Business Information Systems* (pp. 337-347). Cham: Springer International Publishing.
- Immonen, A., Palviainen, M., Ovaska, E. (2014). Requirements of an open data based business ecosystem. *IEEE Access*, 2, 88–103. <https://doi.org/10.1109/ACCESS.2014.2302872>.
- Janssen, M., Konopnicki, D., Snowdon, J.L., Ojo, A. (2017). Driving public sector innovation using big and open linked data (BOLD). *Information Systems Frontiers*, 19(2), 189–195. <https://doi.org/10.1007/s10796-017-9746-2>.
- Japkowicz, N., & Shah, M. (2011). *Evaluating Learning Algorithms: a Classification Perspective*. New York: Cambridge University Press.
- Jetzek, T., Avital, M., Bjorn-Andersen, N. (2014). Data-driven innovation through open government data. *Journal of Theoretical and Applied Electronic Commerce Research*, 9(2), 100–120. <https://doi.org/10.4067/S0718-18762014000200008>.
- Lakomaa, E., & Kallberg, J. (2013). Open data as a foundation for innovation: the enabling effect of free public sector information for entrepreneurs. *IEEE Access*, 1, 558–563. <https://doi.org/10.1109/ACCESS.2013.2279164>.
- Lindman, J., Kinnari, T., Rossi, M. (2014). Industrial open data: Case studies of early open data entrepreneurs. In *2014 47th Hawaii International Conference on System Sciences* (pp. 739–748). <https://doi.org/10.1109/HICSS.2014.99>.
- Marz, N., & Warren, J. (2015). *Big data: principles and best practices of scalable realtime data systems*. Manning.
- Masip-Bruin, X., Ren, G.J., Serral-Gracià, R., Yannuzzi, M. (2013). Unlocking the value of open data with a process-based information platform. In *2013 IEEE 15th Conference on Business Informatics* (pp. 331–337). <https://doi.org/10.1109/CBI.2013.54>.
- Oliver, M., Palacín, M., Domingo, A., Valls, V. (2012). Sensor information fueling open data. In *2012 IEEE 36th Annual Computer Software and Applications Conference Workshops* (pp. 116–121). <https://doi.org/10.1109/COMPSACW.2012.31>.
- Open Data Charter (2017). Open data charter. <https://opendatacharter.net/>.
- Open Knowledge International (2015). Global open data index. <https://index.okfn.org>.
- OpenGovData.org (2007). The 8 principles of open government data (opengovdata.org). <https://opengovdata.org/>.
- Ordóñez, A., Alcazar, V., Rendon, O.M.C., Falcarin, P., Corrales, J.C., Granville, L.Z. (2015). Towards automated composition of convergent services: a survey. *Computer Communications*, 69, 1–21. <https://doi.org/10.1016/j.comcom.2015.07.025>. <http://www.sciencedirect.com/science/article/pii/S0140366415002637>.
- Pereira, G.V., Macadar, M.A., Luciano, E.M., Testa, M.G. (2017). Delivering public value through open government data initiatives in a smart city context. *Information Systems Frontiers*, 19(2), 213–229. <https://doi.org/10.1007/s10796-016-9673-7>.
- Provost, F., & Fawcett, T. (2013). *Data Science for Business: What You Need to Know About Data Mining and Data-analytic Thinking*, 1st edn. Newton: O'Reilly Media, Inc.
- Schuff, D., Corral, K., St Louis, R.D., Schymik, G. (2018). Enabling self-service bi: a methodology and a case study for a model management warehouse. *Information Systems Frontiers*, 20(2), 275–288. <https://doi.org/10.1007/s10796-016-9722-2>.
- Sieber, R.E., & Johnson, P.A. (2015). Civic open data at a crossroads: Dominant models and current challenges. *Government Information Quarterly*, 32(3), 308–315. <https://doi.org/10.1016/j.giq.2015.05.003>. <http://www.sciencedirect.com/science/article/pii/S0740624X15000611>.
- Thorsby, J., Stowers, G.N., Wolslegel, K., Tumbuan, E. (2017). Understanding the content and features of open data portals in american cities. *Government Information Quarterly*, 34(1), 53–61. <https://doi.org/10.1016/j.giq.2016.07.001>. <http://www.sciencedirect.com/science/article/pii/S0740624X16301071>.
- Twitter (2019). Twitter developers portal. <https://developer.twitter.com/>.
- Zaragozaí, B., Giménez-Font, P., Navarro-Carrión, J.T., Rico-Amorós, A., Ramón-Morte, A., Belda-Antolí, A. (2015). A proposal for design and implementation of a hybrid navigation system based on open data, augmented reality and big data: Applications for the smart cities. In *2015 1st International Conference on Geographical Information Systems Theory, Applications and Management (GISTAM)* (pp. 1–6).
- Zeleti, F.A., & Ojo, A. (2017). Open data value capability architecture. *Information Systems Frontiers*, 19(2), 337–360. <https://doi.org/10.1007/s10796-016-9711-5>.
- Zuiderwijk, A., & Janssen, M. (2014). Open data policies, their implementation and impact: a framework for comparison. *Government Information Quarterly*, 31(1), 17–29. <https://doi.org/10.1016/j.giq.2013.04.003>. <http://www.sciencedirect.com/science/article/pii/S0740624X13001202>.
- Zuiderwijk, A., Helbig, N., Gil-García, J.R., Janssen, M. (2014). Special issue on innovation through open data: a review of the state-of-the-art and an emerging research agenda: Guest editors' introduction. *J Theor Appl Electron Commer Res* 9(2):i–xiii. <https://doi.org/10.4067/S0718-18762014000200001>.

Zuiderwijk, A., Janssen, M., Dwivedi, Y.K. (2015). Acceptance and use predictors of open data technologies: Drawing upon the unified theory of acceptance and use of technology. *Government Information Quarterly*, 32(4), 429–440. <https://doi.org/10.1016/j.giq.2015.09.005>. <http://www.sciencedirect.com/science/article/pii/S0740624X15300101>.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Maciej Grzenda is an Associate Professor in the Faculty of Mathematics and Information Science at the Warsaw University of Technology (WUT). He received his Ph.D. from the Warsaw University of Technology. His primary research interests include data preprocessing for machine learning, and the use of machine learning techniques for industrial applications. He has published in international journals on the applications of intelligent data analysis including *Applied Soft Computing* and the *Journal of Intelligent Manufacturing*. He has been also planning and managing R&D projects focused on intelligent data processing for academia and business including Orange Polska telecom. Dr. Grzenda was appointed Vice Dean for Development of the Faculty of Mathematics and Information Science at WUT in 2016. He has been supervising the development of Data Science studies at the faculty since then.

Jaroslaw Legierski received his M.Sc. in electronics and telecommunication and his Ph.D. degree in electronics from the Technical University of Lodz. Since 1998, he has worked in the telecommunications industry. He is currently R&D Expert in IoT and Advanced Technologies at Orange Polska. Jaroslaw Legierski is the co-creator of the Open Middleware 2.0 Community (www.openmiddleware.pl). His research interests include open application programming interfaces (APIs), Open (Big) Data, and next-generation telecommunication services. He is the author of publications in the area of Open APIs. He was the main architect of Warsaw open data platform developed during *Apps for Warsaw* project (<http://www.apps4warsaw.org/>), used currently to provision over 250 open data resources of the city of Warsaw including data streams and big data resources