University of Zurich[UZH]

Impact of Changes on Operations over Knowledge Graphs

Master Thesis, August 20, 2018

Romana Pernischova
of Zlaté Moravce, Slovakia

Student-ID: 12-917-332
pernischova@ifi.uzh.ch

Advisor: **Daniele Dell'Aglio**

Prof. Abraham Bernstein, PhD
Institut für Informatik
Universität Zürich
http://www.ifi.uzh.ch/ddis

Acknowledgements

I would like to show my gratitude towards Professor Bernstein for allowing me to work on such an interesting topic and guiding me in many decisions that were involved in producing this work. Further, I thank my advisor, Daniele Dell'Aglio for all the meetings and valuable discussions about this research as well as the additional inputs and comments. I am also very grateful for the corrections that were given to me by Floyd Basler. They were greatly appreciated. For the never ending patience and encouragements throughout the duration of this thesis, I wish to thank Marc Laville and my family.

Zusammenfassung

Diese Masterarbeit beinhaltet einen Forschungsvorschlag und einen akademischen Artikel, um die Pflichen des Fast Track Programms zu erfüllen.

Wissensgraphen organisieren Informationen in einem Graphen, welcher sich aus tausenden Knoten, die Konzepte und Entitäten darstellen, und aus Verbindungen, die das Verhälnis zwischen den Knoten repräsentieren, zusammenstellt. Sie tragen erfolgreich zur Verbesserung verschiedener Dienstleistungen bei, wie die Websuche oder Faktennachweis. Die Verarbeitung eines Wissensgraphen ist jedoch mit einem grossen Zeit- und Resourcenaufwand verbunden. Dies wird zu einem kritischen Problem, weil Wissensgraphen sich sie weiterentwickeln. Wegen der Evolution des Graphen können vorhergehend berechnete Resultate invalidiert werden. Eine mögliche Lösung ist die Neuberechnung der Operation sobald eine erhebliche Auswirkung wegen der Veränderungen am Wissensgraphen erwartet wird. In meiner Forschung werde ich somit nach Methoden suchen, mit welchen sich die Auswirkung der Evolution des Graphen auf das Resultat einer Operation feststellen lässt.

Ein Beispiel einer Operation ist die Materialisierung des Wisssensgraphen. Ich habe [0]Randić Auswirkung von Veränderungen in Wissensgraphen auf die Materialisierung mittels einer Stützvector Regressionsmodels vorausgesagt. Dazu wurden deskriptive Graphmasse und Veränderungsmasse als Modeleigenschaften verwendet. Nur ein Model hat die Anforderungen von einem RSME kleiner als 0.2 und R-qaured grösser als 0.7 erfüllt. Es ist jedoch wichtig zu betonen, dass die Experimente nicht repräsentativ sind, da 90 Versionen der Gene Ontologie verwendet wurden und die Herangehensweise nicht an einem weiteren Wissensgraphen getestet wurde.

Abstract

This master thesis includes a PhD proposal and an academic paper to fulfill the requirements of the fast track program.

Knowledge graphs (KGs) organize information in graphs, composed of thousands of nodes representing concepts or entities, and edges capturing relations among them. They successfully contributed to different scenarios, including knowledge discovery, Web search engine improvements and fact-checking. However, the operations executed over KGs take large quantities of time and computational resources. This fact becomes a critical issue when KGs receive updates since the evolution of knowledge might invalidate the previously calculated result. A possible solution to this problem is to rerun an operation only when the changes on the KGs have a considerable impact on the result of such an operation. In my research, I investigate methods and approaches to infer the impact of KG changes to the results of operations.

As a first step, I consider the materialization, an operation that computes the deductive closure of the logical axioms contained in the KG. I consider the ^{0}Randić and the Randić topological indexes as two measures of impact, and I study if it is possible to build models to predict them. As input, the model receives features extracted from the KG and the change actions. The best learning method is support vector regression with a linear kernel. Experiments on a real KG (the Gene Ontology) show that the ^{0}Randić is better than Randić in capturing the impact of materialization. The model predicting ^{0}Randić shows a RSME below 0.2 and R-squared above 0.7. The current approach shows several limitations: it considers one ontology and a sequence of 90 versions of it. It will be therefore needed to study if results generalize by repeating the experiments with other ontologies.

Table of Contents

1

PhD Proposal

1.1 Introduction

Various Companies, experts, or communities build knowledge graphs (KG) like Google's Knowledge Graph [1], Microsoft's Satori [2], Facebook's Entity Graph [3], the Gene Ontology [4], or DBpedia [5]. Services such as the Google Search or Facebook's recommendation algorithm profit from KG. When a user queries the Google or Bing search engine, the shown result include advertisements, websites and sometimes a summary. The summary is displayed in a box that shows more information about the searched object. If the results include such a summary, the query was processed over the KG.

If the KG changes, so does the content of the box. Imagine a search for the actress Anne Hathaway, where the summary-box shows her birthday, home town, movies, and more. When Hathaway plays in a new movie, it needs to also be displayed accordingly. For this to happen, the new information about the movie has to be added to KG. Such changes occur on a regular basis and old results can not be reused. The query executes each time a search is done. However, more complex queries and other operations might take much more time to compute and therefore it is not desired to redo the computation every time new information is added, if the information is not relevant for the user.

The answering of a query is an example of an operation over a KG. The inference of a logical entailment for consistency checking, computation of embeddings for feature representations, and estimating recommendations are more examples of operations using KG. They are complex and computationally intensive. Knowing the impact of the evolution would support the decision or recomputation of any of these operations. This means, if the impact of a change is not significant, the result does not have to be computed again. After several minor changes, the KG has possibly evolved too much and a recomputation becomes necessary. The impact would indicate this and recommend the execution of the operation over the new version of the graph. This approach would save an enormous amount of computation power and time.

[1] *https://www.google.com/intl/bn/insidesearch*
[2] *https://blogs.bing.com/search/2013/03/21/understand-your-world-with-bing*
[3] *https://www.facebook.com/notes/facebook-engineering/under-the-hood-the-entities-graph*
[4] *http://www.geneontology.org*
[5] *https://wiki.dbpedia.org*

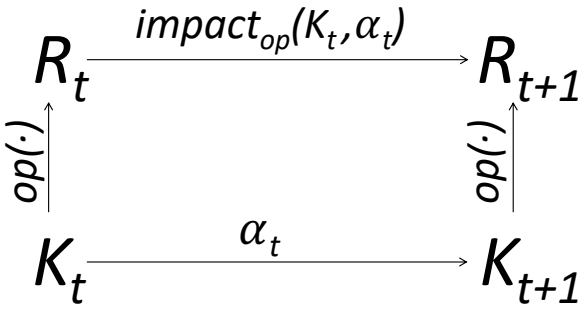


Figure 1.1: Context and problem shown in terms of time and the connection between the different variables included in the problem.

In my PhD studies, I want to investigate the impact of changes on the result of different operations over KG and predict it. The studying of the different operations might show that there is a general model that can be applied, or that the operations require separate models. Further, each KG is different in the way it is used and what kind of information it holds. Depending on its purpose, the model might differ because the models could be dominated by different features.

In the next section, I present my problem statement and explain the necessary variables. In Section 1.3, I discuss work which has been done so far in respect to my proposed topic. I state my research questions and explain their purpose in Section 1.4. Further, I explain how I will tackle each of the questions, what methods I will apply, as well as how I plan to evaluate the results in Section 1.5. Lastly, in Section 1.6 you will be able to read about potential risks and stopping conditions.

1.2 Problem Statement

Let K be a KG, and K_t and K_{t+1} be two versions of K at time instants t and $t+1$. K_{t+1} is obtained by applying a set of changes α to K_t. The problem I am going to investigate in my PhD project is: **How do changes impact the result of an operation over a KG?** Figure 1.1 helps in explaining the setting into which my problem is situated.

Starting from a KG, one can apply different operations to it. In Figure 1.1 the operation is abbreviated by $op(\cdot)$, which is applied to the KG K and returns the result R.

Between K_t and K_{t+1} a change can be an addition, deletion, or modification of a concept, relation or attribute. Further, changes are aggregated into actions, e.g. addition an inner node, movement of a relation, node merge, or subgraph deletion. Two consecutive versions of the graph are edited by a set of actions α.

R_t is the result of the operation $op(\cdot)$ over K_t at time t. R_{t+1} is then the result of $op(\cdot)$ of K_{t+1}. I call the distance between R_t and R_{t+1} impact, $impact(\cdot)$, and it can be defined as a function of the operation $op(\cdot)$, the KG K_t, and the set of actions α as shown in Figure 1.1.

1.3 Related Work

Many researchers have focused on topics close to my proposed research. Initially, I will address the description of a KG or ontology with measures. Second, selected research on the evolution of KGs is stated. Subsequently, a few possible operations over KGs are described. And lastly, I report research done about the impact of KG evolution.

1.3.1 Features of a KG

Social network analysis is an established field that provides many useful measures in terms of describing a graph. John Scott introduces many of these measures in his book "Social Network Analysis" [28]. The number of edges and nodes is used to report the size of a graph. From these two measures, the density of the graph can be calculated easily. Centrality measures are used to talk about the information flow inside a graph. Average degree, average betweenness and average closeness are local measures and refer to the position of one node in respect to all other nodes. Global measures, such as transitivity, address the information flow in the entire graph. Average Shortest Path also represents the information flow. Clustering inside a graph indicates reports the ability and likelihood of the graph to form clusters. More measures about the clustering ability includes the partitioning of the graph into center and periphery and comparing their sizes.

Graph entropy describes the information value in a graph [9]. There are different ways of calculating entropy and the base measure depends on what type of information should be captured [9]. In their paper, Dehmer and Mowshowitz [9] discuss all the different entropy measures used in the past and highlight relevant differences.

Ontology specific measures need to be addressed more particularly. Simple measures include the number of concepts, sub-concepts, and predicates [35]. Further, one can also report the number of entities vs the number of concepts, to compare the size of the schema to the data [35]. Design complexity measures combine multiple of aforementioned values [37].

1.3.2 Evolution of KGs

A large amount of research has focused on studying the evolution of Wikipedia, such as [2, 19, 25, 30]. Almeida, Mozafarim, and Cho [2] found that most users only contribute to one article in a single interaction and do not edit multiple pages in one session. Kaltenbrunner and Laniado [19] focus on the analysis of the discussions and find that complexity of discussion varies greatly among articles. Further, Mestyan et al. [25] make use of the Wikipedia page edit behavior to predict the box office success of movies. Steiner et al. [30] investigate Wikipedia edits in terms of who the contributors are. They analyze the edit of bots versus people as well as anonymous edits compared to logged in users [30].

DBpedia is a KG, which has been missing the option of accessing previous versions up to 2015 [11], which made an analysis of its evolution impossible. Fernandez et al. published the BDpedia Wayback Machine and a WebUI and a RESTful API

to access old versions and revisions to follow the Linked Open Data principles [11]. However, they did not analyze DBpedia in any way.

The work on the detection of ontology changes and its classifications sparked my interest. OntoDiff [33] is a tool that enables the user to detect changes between two versions of the same graph. Their approach is based on the identification of semantically equivalent elements between the ontologies [33]. COnto-Diff [16] and the integrated CODEX [15] is another tool offering the computation of changes. Besides detecting changes, they also group together low level changes into high level change actions and therefore, provide a simple classification. Goncalves, Parsia, and Sattler [13] also addressed the problem of categorizing changes between ontologies. However, their categorization is based on the logical impact of changes [13].

Klein and Noy [20] developed an ontology, in which they describe 80 basic changes. They also introduce complex changes and show how they help in the interpretation of consequences for data and entities [20].

ChangeDistiller [12] presents an improved algorithms to detect and categorize changes in a tree. The authors apply it to software code, but for change detection they transform code into the tree [12]. The change types that ChangeDistiller [12] extracts are aligned with the simple changes detected in ontologies with COnto-Diff [16].

1.3.3 Operations using KGs

Trivedi et al [32] develop deep temporal reasoning over KGs. They do not ignore the evolution of the graph and account for it by adding a time component in their reasoning [32]. They successfully test their approach on the temporal prediction of links [32].

Machine learning is another operation which is executable over a KG. Chen et al. [4] learn models over KG streams and asses their accuracy as the stream evolves. Their work is explained in more detail in Section 1.3.4. Relational machine learning is an approach to predict links in a graph [26]. Using a KG it is possible to learn a statistical model that predicts new knowledge [26]. Nickel et al. [26] discuss various algorithms and methods in this domain. Yao et al. [36] investigate how probabilistic topic models improve when using Wikipedia knowledge as additional input. However, they do not use an actual KG but Wikipedia articles to improve their learning process [36].

Machine learning is also used in the process of learning embeddings for KGs. The goal of embeddings is to represent the graph in a smaller vector space [18]. Ji et al. [18] introduce a novel method for embedding a graph by adding a dynamic mapping matrix. They use two vectors instead of one, where the second vector is used for the construction of a dynamic mapping [18]. On the other hand, embeddings can be improved by targeting a specific task for which the embedding is then used. Lin et al. [22] discuss the improvements of embeddings for graph completion. Their method performs better than state of the art when comparing link predictions [22]. Zhu et al. [38] propose a novel method using generalized hyperplanes for embedding KGs. They improve the representation of interactions between entities and relations and preserve a good compression rate at the same time [38]. By testing on link

predictions and triple classification, they compare their method to the state of the art [38] but not to the results by Lin et al. [22].

Another operation is the answering of queries. Ren et al. [27] collect competency questions for the evaluation of KGs. Their goal is to define a way of testing KGs against a set of questions to asses requirements [27]. Fact checking is traditionally done by journalists, yet a new trend of computational fact checking is emerging [34]. Computational fact checking is a subcategory of query answering. Wu et al. [34] built a framework for formulating fact-checking queries for databases and propose an approach using counter claims. Ciampaglia et al. [5] built their own reduced KG from Wikipedia's info boxes and checked facts with a very high accuracy using the query formulating approach of Wu et al. [34]. They used shortest paths and other methods from network analysis [5].

1.3.4 Impact of KG Evolution

Gross et al. [14] examine how the changes in an ontology have an impact on previously conducted functional analysis [14]. However, they examining the impact with a stability measure that is specifically chosen to evaluate the task of functional analysis. A functional analysis as an operation is very specific to the Gene Ontology. We focus on a larger spectrum, where our proposed approach can be used across different domains of KGs.

Know-Evolve [32] is a model that enables deep temporal reasoning. The authors introduce their novel idea of reasoning for dynamic KGs and are able to apply machine learning and at the same time, taking advantage of the evolution of the graph [32]. The time component directly affects the results of the reasoning and can therefore also be seen as impact.

Further, SemaDrift [29] is a tool that lets the user calculate the semantic drift between versions of ontologies. It provides the calculations of various measures to the user [29]. Depending on the chosen aspect, the authors deploy the different measures to report semantic drift between two related ontologies [29]. They apply different methods of calculation and they distinguish between an exact, inexact and hybrid matching approach [29].

Chen et al. [4] discuss how learned models become less accurate as a stream evolves semantically. Their work is directly related but they use machine learning as their operation instead of the materialization. Impact is measured in terms of accuracy loss and changes are addressed using concept drift.

1.4 Research Questions

The final goal of my research is the prediction of impact based on the old version of the KG and the changes that will be applied to it. To construct a general model of impact, I need to investigate how it depends on different operations. Questions 1 through 3 will therefore be investigated iteratively for each operation. I plan on addressing three different operations.

1.4.1 What is impact?

My first research question addresses the problem of finding the best way to measure the impact. As Figure 1.1 shows, impact is the difference between R_t and R_{t+1}. The answer of this question is very dependent on the particular operation and will therefore be measured in a different way for each such operation

RQ 1. *What is the notion of impact over the results of op(·)?*

For each operation there will be different measures that can be used and each of them will have to be tested. Taking the materialization as an example the following hypothesis can be formulated by substituting [measure] with a chosen impact measure and [*op*(·)] with an operation. I will use the normal distribution in the range between 0 and 1 as the indicator.

H 1.1. *The [measure] captures the impact on results of [op(·)] with a normal distribution.*

1.4.2 What affects the impact?

Once the impact is defined, it is of interest to know how it behaves compared to graph and change features. These features capture different aspects of the graph.

RQ 2. *How is the impact affected by K and α?*

- ***(a)*** *Given the impact(op(·),K,α), how does α affect the impact on op(·)?*
- ***(b)*** *Given the impact(op(·),K,α), how does K affect the impact on op(·)?*

These two research questions have to be answered separately for each operation. The impact will differ between the operations, but the feature vectors describing the KG and the change actions will be the same. Every change affects multiple nodes and for each of the affected nodes multiple features can be recorded. The following two hypotheses will help in answering the second research question. The keyword [measure] has to be substituted with a change feature and [*op*(·)] with the operation over which it will be evaluated. In the second hypothesis, [feature] is the measure describing the graph. I will evaluate the hypotheses with Pearson correlation analysis.

H 2.1. *The [measure] of nodes directly affected by α correlates with the impact on [op(·)].*

H 2.2. *The [feature] of K correlates with the impact on [op(·)].*

1.4.3 Can the impact be predicted?

After analyzing the correlation between the descriptive measures and the impact, a prediction model can be built.

RQ 3. *Can the impact be predicted given $op(\cdot)$, K, and α?*

The corresponding hypothesis include various feature models. One model will include all features, another one will be built using correlating features from the previous research question and one will be defined using a feature selection algorithm like ridge regression. I will use multiple machine learning algorithms to compare the models. I plan to test general linear regression, support vector machine with different kernels and random forest.

The hypothesis should be read by substituting the keywords [feature model], [impact measure], [$op(\cdot)$], and [algorithm].

H 3.1. *The prediction learned with [algorithm] of the [impact measure] on [$op(\cdot)$] using the [feature model] has a RMSE below 0.2 and a R-squared above 0.7.*

1.4.4 Can the impact be generalized?

Finally, I will compare all results across operations. If the models that are built for each of the operations are similar to each other, then a generalization will be possible. Otherwise, it will be summarized by showing the differences between operations and how the impact changes. This would mean, that I would remove *op* from the calculation of the impact, which would leave me with the impact being the function of K and α - $impact(K, \alpha)$.

RQ 4. *Is it possible to predict the impact of changes done on a KG independently from the operation?*

First the impact between the different operations has to be assessed. I will examine the correlating features across operation and I hope that there are several features in common. Lastly, if there is a notion of impact that can be used across operations and there are also features that correlate with it, a prediction can be tested.

1.5 Approach and Evaluation plan

In this section, I will first talk about the data that I intend to use for answering my research questions. Further, I present my approach to calculating the impact, followed by how features will be extracted. For the prediction of the impact, feature selection is also necessary, which is addressed in the last part of this section.

1.5.1 Data Preparation

Using different knowledge bases with the same operation is essential. Each KG is unique in the way it is used and edited. If each operation was tested on one KG, no generalization would be possible at all.

For some KGs archived releases can be downloaded but that holds a vast amount of changes between releases. Taking the changes that have been applied to a KG

in the past, a version will be created for each change. This creates the possibility of analyzing each change individually instead of as a set. For the calculation of the changes, I will use the code from the tool CODEX [15] and COnto-Diff [16] developed by the same group. These tools were developed for the Gene Ontology, but I will change and generalize it for usage with other KGs. The authors classify changes into high level change action that give some meaning to low level changes such as adding or deleting attributes. Move, substitute, merge, add inner, add sub graph, or delete leaf are high level change actions, that each encompass at least three low level changes. Some action types are very specific for the Gene Ontology and therefore, will be excluded.

Datasets of various KGs have to be prepared. There is the possibility of reducing a KG by extracting a part from it and treating this part as a whole. The changes and their impacts can be observed on a smaller scale and it will be easier to draw conclusions.

1.5.2 Calculating Impact

Each operation requires a different measure of distance between the results. Therefore, state of the art will be used to calculate the impact and address Research Question 2. I will use multiple measures to cover different aspects of the impact.

I will look at the distribution of the chosen impact measures. The distribution should not be skewed and should range from 0 to 1. I will randomly choose versions to check the impact and changes. This will help in understanding how the impact measure is behaving with respect to the different change types.

1.5.3 Generating Feature Vectors

Features describe the KG and the change actions. I will calculate various features to address different aspects of the graph and changes. For each research question, these will remain the same, since they are chosen independent of the operation.

To be able to use the KG as the input to predict the impact, features have to be extracted. Such features describe the various aspects of the KG, e.g. the structure, density, or connectivity. Descriptive measures mentioned in Section 1.3.1 will be calculated for each version.

Additionally, each action will also be described with a feature vector. This is necessary because the format of the actions would not allow for usage in machine learning. For this feature vector, information on node level will be recorded. All nodes that are directly affected by the action will be examined and various structural and descriptive measures calculated.

1.5.4 Predicting Impact

Given the feature vectors, a correlation analysis will answer the second research question. With that answer it is possible to choose features for a second feature model. The first model will include all the calculated features. Other feature models can be done by using state-of-the-art feature selection algorithms like ridge regression.

Using different machine learning algorithms, I will evaluate across the feature models. I expect to use general linear regresssion, support vector machine with a linear and radial kernel, and random forest. Other algorithms might be added to this list at a later time.

For the answering of Research Question 4, the results of all the steps described above have to be taken into account. There are different possible outcomes, where the best case would be a general model over all operations and KGs. To accomplish it, a comparison between the KGs and operations will be done to asses the models that will be built.

1.6 Reflections

I want to investigate three operations and answer Research Questions 1 through 3. The fourth research question is more open and is extremely dependent on the outcome of my research up to that point. Although I want to investigate a general model, the outcome is very uncertain.

I see the highest risk in the fact, that it is possible, that there are no features that will correlate with the impact, neither describing the KG or the changes. In this case, I will calculate a different impact measure and add features that describe the change of the graph in terms of graph measures. It is also possible that, once the KG is too big, no change will have a big impact and the impact measure will show a skewed distribution. In this case, the KG can be partitioned across topics or actions can be grouped together.

2

How Changes in a Knowledge Graph Impact its Materialization

2.1 Introduction

Knowledge graphs (KG), like Google's Knowledge Graphs [1], Facebook's Entity Graph [2] or the Gene Ontology [3] change over time. They represent the knowledge of a universe that grows. These graphs are usually maintained by experts, which insert new knowledge into the graph and remove or change outdated information. The Gene Ontology (GO) evolves as the experts add gene annotations of living organisms that are not included yet. Researchers then execute analysis and functions over this graph. The computation of the logical entailment, also called materialization is such a function applied to the GO. This operation makes the assessment of logical consequences and consistency possible. Adding an axiom to the KG that contrasts other axioms defined in the schema leads to the conclusion of inconsistency within the ontology. A materialized graph is bigger than the original graph and its computation consumes vast amounts of time and power resources. Consequentially, it may not be desired to compute it after every minor change, but we would rather be interested in knowing when it is necessary to recompute the materialization. The expectation of a big difference from the old materialization to the new one would signal the necessity for recomputation.

Once the changes to the KG have a significant impact on the materialization, it is time for the computation on the new version of the graph. In this work we study if we can automate this process and predict the impact of changes. We define impact based on measures of graph distance, which assesses the structure and not the semantics. Impacts on the semantics of a KG is usually intentional and the prediction of such an impact will seam obvious. Structural changes are more hidden and therefore, we focus on those.

Therefore, our first research question addresses the impact and the problem of assessing the chosen measure that expresses impact between two materializations. We take into account that such a measure has not been discussed before this context, and chose to examine two graph distance measures based on topological indexes introduced by Dehmer and Mowshowitz in [8].

[1] *https://www.google.com/intl/bn/insidesearch*
[2] *https://www.facebook.com/notes/facebook-engineering/under-the-hood-the-entities-graph*
[3] *http://www.geneontology.org*

RQ 5. *How can impact over versions of materializations be defined and quantified?*

Once the impact is calculated, it is of interest to know how it behaves compared to various measures that act as descriptive features. Two research questions are necessary, since we distinguish between measures that describe the KG and ones describing the changes leading to the new version of the graph. In the following, the materialization operator is denoted with $mat(\cdot)$, K stands for the description of the KG, and α represents the change measures. $impact(\cdot)$ is the impact dependent on the operation $mat(\cdot)$, the KG K and the change operations α.

RQ 6. *How is the impact affected by K and α?*

- ***(a)*** *Given the impact($mat(\cdot)$,K,α), how does α affect the impact?*
- ***(b)*** *Given the impact($mat(\cdot)$,K,α), how does K affect the impact?*

After analyzing the influence of the descriptive measures on the impact, a prediction model can be built. We tested three different feature models against each other. One includes all the features that were calculated, the second consists of the features that correlate with the impact significantly and the features for the last model were selected with the ridge regression.

RQ 7. *Can the $impact(mat(\cdot), K, \alpha)$ be predicted given $mat(\cdot)$, K, and α?*

In the next section, we introduce related work about the evolution of KGs, graph and distance measures, and impact analysis. In Section 2.3, we discuss the data, feature vectors, impact, and how it was evaluated. We introduce our results and formulate a discussion in Section 2.4. The last section then states conclusions, limitations, and future work.

2.2 Related Work

This section presents related topics to different aspects of our work. First, we address the evolution of KGs, followed by graph measures and distance, and lastly research on impact of KG evolution.

2.2.1 On the Evolution of Knowledge Graphs

A big amount of research focused on studying the evolution of Wikipedia, such as [2, 19]. These two studies focus mostly on the content evolution and the behavior of the contributors. Almeida, Mozafarim, and Cho [2] found that users only contribute to one article in a single interaction and do not edit multiple pages in one session. Kaltenbrunner and Laniado [19] focus on the analysis of the discussions and find that

complexity of discussion varies greatly among articles. Further, Mestyan et al. [25] make use of the edit behaviour of movie pages to predict the box office success. However, they all focus on the edits on Wikipedia articles, the contributors, and the content.

DBpedia is a KG based on Wikipedias structured content. It was impossible to analyze the evolution of DBpedia because it has been missing the option to access previous versions until 2015 [11]. Fernandez et al. published the BDpedia Wayback Machine which provides a WebUI and a RESTful API to access old versions and revisions following the Linked Open Data principles [11]. Howerver, they do not analyze DBpedia in any way.

Ontologies are KGs as well and they grow just like KGs do. What sparked our interest is the work on the detection of ontology changes and its classifications. Various tools have been developed. OntoDiff [33] is one such tool that enables the user to detect changes between two versions of the same graph. COnto-Diff [16] and the integrated CODEX [15] is another tool offering the computation of changes. Besides detecting changes, they also group together low level changes into high level change actions and therefore, provide a simple classification. Goncalves, Parsia, and Sattler [13] also addressed the problem of categorizing changes between ontologies. However, their categorization is concerned with changes that have a logical impact versus changes that do not [13].

Klein and Noy [20] developed an ontology of changes, where they describe 80 basic changes. They also introduce complex changes and show how they help interpretation of consequences for data and entities [20].

2.2.2 Graph Distance and Ontology Similarity

Dehmer, Emmert-Streib, and Shi [8] introduce a notion of graph distance based on different topological indexes of graphs. They analyze the index distance theoretically and provide limited numerical experiments [8]. Some older work of Dehmer and Emmert-Streib [7] also suggests other graph similarity measures that return extremly similar results to the well studied Graph Edit Distance. They use differences of degree vectors to cover the structural information of the graphs [7].

The majority of papers that focus on semantic similarity within ontologies are only interested in similarity between concepts and not entire ontologies. Lord et al. [23] explore the Gene Ontology (GO) using semantic similarity and examine the similarity over different aspects inside the GO. Lee et al. [21] show a comparison of different semantic similarity measures on similar concepts. Just like [23], they do not apply their approach to entire ontologies [21].

Ontology distance can also be expressed using a modification matrix [1]. The exploration of the proposed matrix is limited and does not show enough detail to be reused [1]. Algosaibi and Melton [1] use the concept from Yang, Zhang, and Ye [35] which describes an ontology in terms of complexity by using its hierarchy. The metrics are extremly primitive. Zhang, Li, and Tan [37] use more complex measures based on a proper graph representation of an ontology. Meadche and Staab [24] take an approach in measuring similarity between ontologies that is based on a two layer concept, where an ontology consists of a lexical and conceptual layer. They apply

the proposed measures in an empirical experiment and they do not compare to any other established similarity measure [24].

David and Euzenat [6] compare various ontology distance measures. They take into account similarity measures based on the vector space model, distance between entities, and collection distances [6].

It is important to mention SemaDrift [29] which is also included in the next section. This tool provides measures related to semantic drift [29]. Semantic drift can be interpreted as the distance between ontologies, under the assumption that the ontologies are related.

2.2.3 Impact of Knowledge Graph Evolution

Gross et al. [14] examine how the changes in an ontology have an impact on previously conducted functional analysis [14]. Even though they have the same goal of this research, their methods are limited to examining the impact with a stability measure that is specifically calculated for the GO. We focus on a larger spectrum, where our proposed approach can be used across different domains of KGs.

Know-Evolve [32] is a model that enables deep temporal reasoning over dynamic KGs. The authors are able to apply machine learning over the graph and predict re-occurrence of events [32]. The time component directly affects the results of the reasoning and can therefore also be seen as impact.

Further, SemaDrift [29] is a tool that lets the user calculate the semantic drift between versions of ontologies. It provides the calculations of various measures to the user [29]. Depending on the chosen aspect, the authors deploy the different measures to report semantic drift between two related ontologies [29]. They apply different methods of calculation and they distinguish between exact, inexact and a hybrid matching approach [29].

Chen et al. [4] discuss how learned models become less accurate as a stream evolves semantically. Their work is directly related but they use machine learning as their operation instead of the materialization. Impact is measured in terms of accuracy loss and changes are addressed using concept drift.

2.3 Case Study of the Gene Ontology

To study the research questions introduced in Section 2.1, we looked for a KG that provides multiple versions. In addition, we looked for a graph of medium size. We chose the Gene Ontology (GO), a small ontology compared to the size of dbpedia or Wikidata. The Gene Ontology Consortium maintains the GO since 2000. It provides a precise and common vocabulary to describe the role of genes and gene products in any organism [3]. At the beginning, the GO Consortium provided three separate ontologies. There have been many improvements and expansions to the ontologies and they can now be used as one ontology, which underlines our statement about the evolution of knowledge bases [31]. Because of the maintenance by experts, vandalism is not present in the ontology [17]. This allows for an ad-hoc exploration of the graph and the concepts within the graph are limited to the descriptions of genes. The homogeneous topic withing the ontology makes interpretation of the

results simpler. We calculated feature vectors over multiple aspects of the GO, which are explained in the subsections below.

We were able to download and use 99 versions. Due to complications in various stages, we ended up with 90 cases of consecutive versions. Table in A.4 shows the GO versions and the number change actions to the next version.

2.3.1 Impact

The impact is calculated on the materialization of the KG. Since a materialization is a graph, graph distance measures also apply to it. Based on [10], we decided to use the work in inexact graph matching due to the computation complexity and size of the GO. When ignoring labels, we focus on the structure of the graph rather then on the semantics. Dehmer, Emmert-Streib, and Shi [8] propose the approach of using the graph distance with the Randić, and zeroth order Randić index. Both are topological indexes and for each version of the GO we calculated their according values. The two indexes were implemented in Python 3.5 using the NetworkX [4] library. The following equation returns the distance between the versions by taking the indexes and σ as input [8]:

$$D_1(K_t, K_{t+1}) = 1 - e^{-\frac{(I_{K_t} - I_{K_{t+1}})^2}{\sigma^2}} \tag{2.1}$$

where I_{K_t} and $I_{K_{t+1}}$ are the topological index of K_t and K_{t+1}.

We formulate the following hypotheses using the distance and the two indexes:

H 5.1. *The graph distance based on the Randić index calculated with*

$$R(K_t) = \sum_{uv \in E(K)} \frac{1}{\sqrt{(d(u) \times d(v))}}$$

captures the impact on mat(K_{t_1}) of a change with a normal distribution.

H 5.2. *The graph distance based on the* ^{0}*Randić index calculated with*

$$^0R(K_t) = \sum_{u \in V(K)} \frac{1}{\sqrt{d(u)}}$$

captures the impact on mat(K_{t_1}) of a change with a normal distribution.

uv is an edge in the graph K and $d(u)$ is the degree of the node u and v respectively. For these hypotheses, we report diagrams that show how the impact changes over the evolution of the Gene Ontology and their distributions.

[4] *https://networkx.github.io*

Action type	Mean	Standard Deviation
move	126.7065	143.9922
split	3.0870	2.7372
merge	6.8153	9.8906
add subgraph	30.1630	19.9779
add inner	19.0326	12.2461
add leaf	62.4783	35.1594
to obsolete	5.7065	7.7999

Table 2.1: Average occurrences of change action types on the Gene Ontology for multiple versions.

2.3.2 Change Action Measures

We use [16] to find changes between two versions of the GO. They define nine low level change actions, which are add, delete, map a concept; add, delete, map a relation; and add, delete, map an attribute A concept is a class in the ontology, a relation connects two concepts with each other and an attribute is attached to a concept and serves as additional information to the concept. These nine actions are then grouped together to produce high level actions, which are: move, merge, split, add and delete inner node, add and delete leaf, add and delete subgraph, and to and revoke obsolete.

After analyzing these high level actions, it became clear that high level actions can include other high level actions. However, since we are only interested in the highest level, we modified the code of COnto-Diff [5] to return only those. Since we have snapshots of the GO at the beginning of each month, multiple change actions are grouped together between two versions. In Table 2.1 we report the mean count of each of the change action types present between two consecutive versions retrieved and returned by the our function. All the versions and the number of retrieved highest level actions are reported in Table in A.4.

We used Pearson correlation to analyze the relationship between the impact and the features to answer the Research question 6. First, we address change actions that are concerned with an addition of either an inner node, leaf or entire subgraph. Different aspects of each of these actions are of concern. The number of such changes between two versions is one of the important measures. The remaining eight measures are mean and standard deviation of the degree, degree centrality, closeness, and betweenness. Together these nine measures are investigated for every action type. Table 2.2 shows the hypotheses numbers and should be read in the following way, substituting the keywords *measure* and *action type* with the corresponding row and column: *The [measure] of [action type] inside α correlates with the impact.*

A change action for adding inner nodes takes the following form: *addInner,GO:0044278.* This is partitioned and the implemented algorithm returns the degree of the directly affected nodes. For the addition of an inner node, we look for the the neighbors in the new version of the graph. For all the direct neighbors, degree is returned from

[5] *http://dbserv2.informatik.uni-leipzig.de:8080/webdifftool/WebDiffTool.html*

Measure / Action type	addInner	addLeaf	addSubgraph	merge	move	split	toObsolete	all
number of actions	H 6.2a	**H 6.3a**	**H 6.4a**	**H 6.5a**	**H 6.6a**	**H 6.7a**	**H 6.8a**	**H 6.9a**
mean of degree	H 6.2b	H 6.3b	H 6.4b	**H 6.5b**	H 6.6b	H 6.7b	H 6.8b	H 6.9b
std of degree	H 6.2c	H 6.3c	H 6.4c	**H 6.5c**	H 6.6c	H 6.7c	H 6.8c	H 6.9c
mean of degree centrality	H 6.2d	H 6.3d	H 6.4d	H 6.5d	H 6.6d	H 6.7d	H 6.8d	H 6.9d
std of degree centrality	H 6.2e	H 6.3e	H 6.4e	H 6.5e	H 6.6e	H 6.7e	H 6.8e	H 6.9e
mean of closeness	**H 6.2f**	H 6.3f	H 6.4f	H 6.5f	H 6.6f	H 6.7f	H 6.8f	H 6.9f
std of closeness	**H 6.2g**	H 6.3g	H 6.4g	H 6.5g	H 6.6g	H 6.7g	H 6.8g	**H 6.9g**
mean of betweenness	H 6.2h	H 6.3h	H 6.4h	H 6.5h	H 6.6h	H 6.7h	H 6.8h	H 6.9h
std of betweenness	H 6.2i	H 6.3i	H 6.4i	H 6.5i	H 6.6i	H 6.7i	H 6.8i	H 6.9i

Table 2.2: Hypotheses table for change measures, to be read in the following way:
The [measure] of [action type] inside α correlates with the impact.
Accepted hypotheses in bold.

Algorithm / Index, Model	R all	^{0}R all	R corr	^{0}R corr	R fs	^{0}R fs
Linear Regression	H 7.1	H 7.2	H 7.3	H 7.4	H 7.5	H 7.6
SVM Linear	H 7.7	H 7.8	H 7.9	**H 7.10**	H 7.11	H 7.12
SVM Radial	H 7.13	H 7.14	H 7.15	H 7.16	H 7.17	H 7.18
Random Forest	H 7.19	H 7.20	H 7.21	H 7.22	H 7.23	H 7.24

Table 2.3: Hypotheses table for impact predictions, to be read in the following way:
The [index] impact prediction on $mat(K_{t_1})$ using the [model] and [algorithm] has a RMSE below 0.2 and R-squared above 0.7.
Accepted hypotheses in bold.

the old and new version. Then, the calculation of the mean of the nodes across the neighbors is executed and the absolute value of the difference is reported. We repeat this procedure for every addition of a inner node between two consecutive versions. Finally, the mean and standard deviation of all inner node additions is calculated to report one number. This number is the input for the correlation analysis. Same principle applies to the calculation of degree centrality, closeness, and betweenness.

We repeated this algorithm for the addition of a leaf, where there are fewer neighbors than with inner nodes, as well as with subgraph. Specifically with the latter, we only calculate the degree, degree centrality, closeness, and betweenness for nodes which are not part of the new subgraph that has been added with the change action. Hypotheses 6.2, 6.3, 6.4 include all the above explained measures.

Since in the *merge* change action, we know which nodes are directly affected, because multiple nodes are merged into one, we calculate the measures for those nodes in the old and new version. There are multiple affected nodes in the old version and just one node in the new version. Hypotheses 6.5 refer to the measures calculated for the action *merge*. The action *move* also reports which nodes are affected directly. Therefore, we calculate and aggregate over two nodes in each of the versions. Hypotheses 6.6 state all hypotheses concerning the action *move*. *Split* is the counter action to merge and its hypotheses are denoted with the number 6.7. The action splits one node from the old version into multiple nodes in the new versions. The action *toObsolete* always affects only one node. Here, only the difference is necessary and no aggregation is needed. All hypotheses for the action *toObsolete* are stated under the number 6.8. At last, we also aggregate over all the different change actions.

To give the reader an idea about the measured values, Table 2.4 shows the mean of the mean and standard deviation over the considered versions of the GO. One can see that some measures such as *move*, *merge*, and *split* show a very small change in closeness. This means, that closeness is not affected much. However, bigger effects are expected from actions such as *addInner*, *addSubgraph* or *toObsolete*. This does not mean that the effect will be also big on the materialization of the new version.

2.3.3 Graph Measures

Different measures have been established in the community of network analysis. We use those to describe the ontology, even though they are not intended for labeled graphs. Additionally, we looked into graph entropy measures [9] and selected to use entropy based on closeness and degree centrality as a result. The measures that were calculated form the feature vector later. Table 2.5 lists all calculated measures. These chosen measures do not represent a complete list but are rather a selection that was decided upon. All measures were calculated using the Python package NetworkX except for sparsness, for which we used NumPy.

Hypotheses 6.10 and 6.11 investigate the relationship between the impact and the size of K_t.

H 6.10. *The number vertexes in K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.11. *The number edges in K_t correlates with the impact on $mat(K_{t_1})$.*

Action Type	Measure	Mean	Std
addInner	degree	2.9302	3.649
	degree centrality	0.5832	0.7743
	closeness	0.0003	0.0005
	betweenness	0.5832	0.7743
addLeaf	degree	1.7011	2.084
	degree centrality	0.009	0.0512
	closeness	< 0.0001	0.0001
	betweenness	0.009	0.0512
addSubgraph	degree	1.9333	1.8034
	degree centrality	0.1519	0.402
	closeness	0.0001	0.0002
	betweenness	0.1519	.402
merge	degree	0.9366	1.0924
	degree centrality	.0179	0.0458
	closeness	0.0001	.0002
	betweenness	0.0179	.0458
move	degree	0.3925	0.9467
	degree centrality	0	0
	closeness	< 0.0001	< 0.0001
	betweenness	0	0
split	degree	0.6456	0.3951
	degree centrality	0	0
	closeness	< 0.0001	< 0.0001
	betweenness	0	0
toObsolete	degree	1.7638	1.0465
	degree centrality	0.5653	0.4987
	closeness	0.0002	0.0002
	betweenness	0.5653	0.4987
all types	degree	1.3196	2.5608
	degree centrality	0.1081	0.4129
	closeness	0.0001	0.0002
	betweenness	0.1081	0.4129

Table 2.4: Mean and standard deviation of change action measures executed between two versions of the Gene Ontology.

Measure	Mean	Standard Deviation
vertex count	$38'273.4444$	$4'546.8417$
edge count	$76'226.9778$	$12'541.2092$
average degree	3.9607	0.2035
average degree centrality	0.0001	7.89×10^{-6}
average closeness	0.0002	7.63×10^{-6}
average betweenness	3.61×10^{-8}	4.94×10^{-9}
degree connectivity	1.7949	0.0799
assortativity	−0.0298	0.0294
average clustering coefficient	0.0558	0.0047
transitivity	0.0517	0.0025
number of strong components	$38'273.444$	$4'546.8417$
number of cliques	3.6709	0.1694
average shortest path length	6.9951	0.0427
longest shortest path length	14.3000	0.4819
entropy (centrality)	14.5305	0.1706
entropy (closeness)	11.3413	0.0968
sparseness	5.22×10^{-5}	3.95×10^{-6}

Table 2.5: Calculated graph measures over the Gene Ontology reporting the mean and standard deviation over several versions.

We are further interested in showing the relationship between general centrality measures such as degree, closeness, and betweenness of K_t and the impact. These are addressed in Hypotheses 6.12, 6.14, and 6.15. More complex measures that address the information flow are assortativity, transitivity, and average shortest path length in Hypotheses 6.16, 6.17, and 6.18.

H 6.12. *The average degree of K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.13. *The average degree centrality of K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.14. *The average closeness of K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.15. *The average betweenness of K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.16. *The assortativity of K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.17. *The transitivity of K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.18. *The average shortest path of K_t correlates with the impact on $mat(K_{t_1})$.*

Further, clustering inside a graph is also of interest, because it brings about the structure and density of the graph. Hypothesis 6.19 addresses the clustering coefficient, which represents the clustering ability of the graph. Hypothesis 6.20 then investigates the number of detectable clusters inside the graphs compared to the impact.

H 6.19. *The clustering coefficient of K_t correlate with the impact on $mat(K_{t_1})$.*

H 6.20. *The number of clusters in K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.21. *The number of strongly connected components in K_t correlates with the impact on $mat(K_{t_1})$.*

Entropy is a semantic measure and addresses the information content [9]. In Hypotheses 6.22 and 6.23 we report correlations of two different entropies with the impact, where one is based on the degree centrality and the other based on closeness. The last hypothesis investigates the sparseness of the adjacency matrix of the graph.

H 6.22. *The degree centrality based entropy of K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.23. *The closeness based entropy of K_t correlates with the impact on $mat(K_{t_1})$.*

H 6.24. *The sparseness of K_t correlates with the impact on $mat(K_{t_1})$*

2.3.4 Prediction of Impact

We test 3 feature models with 2 impact measures using the 90 cases calculated in the previous steps and 4 different machine learning algorithms.

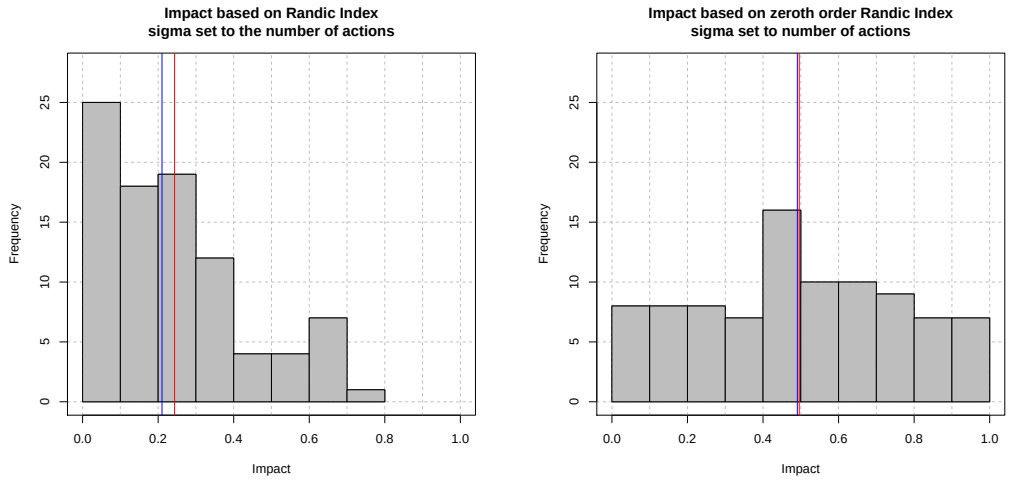
The first feature model is the *all* model, where all calculated features are used. The second feature model is based on the correlation analysis that was done to answer all the hypotheses concerned with the effect of the features on the impact. Since the set of correlating features are non-identical for the the two measures of impact, Randi and ^{0}Randi, the *corr* feature models will use these two differing sets accordingly The third feature model is based on the feature selection executed using ridge regression and it is shortly called *fs*.

We use the described feature models first with general linear regression (GLM). Support Vector Machine is also suitable for regression (SVR) and we decided to use the algorithm with the linear kernel and the radial kernel. In addition, Random Forest (RF) is used. We test all the prediction models using a 10-fold cross validation.

We will compare all prediction results using the RMSE and R-squared and accept a hypothesis if the RMSE is below 0.2 and R-squared is above 0.7. Additionally, for a prediction hypothesis to be accepted, the corresponding impact hypothesis has to be accepted as well. This means, that if an impact measure is declared unsuitable for the task at hand, the predictions of this impact will not be accepted. All hypothesis are stated in the Appendix A.3 and the Table 2.3 shows their reference numbers. The table is to be read by substituting *algorithm*, *index*, and *model* with the corresponding column and row in the sentence: *The [index] impact prediction on $mat(K_{t_1})$ using the [model] and [algorithm] has a RMSE below 0.2 and R-squared above 0.7.*

2.4 Results and Discussion

In this section, we present the results of the analysis of the impact, correlations and prediction. We revisit the hypotheses from Section 2.3 and asses if they hold. We also address the research questions and discuss the results.



(a) Impact with Randić Index

(b) Impact with ^{0}Randić Index

Figure 2.1: Distributions of impact calculated with the graph distance and two different topological indices, the Randić and the ^{0}Randić index.

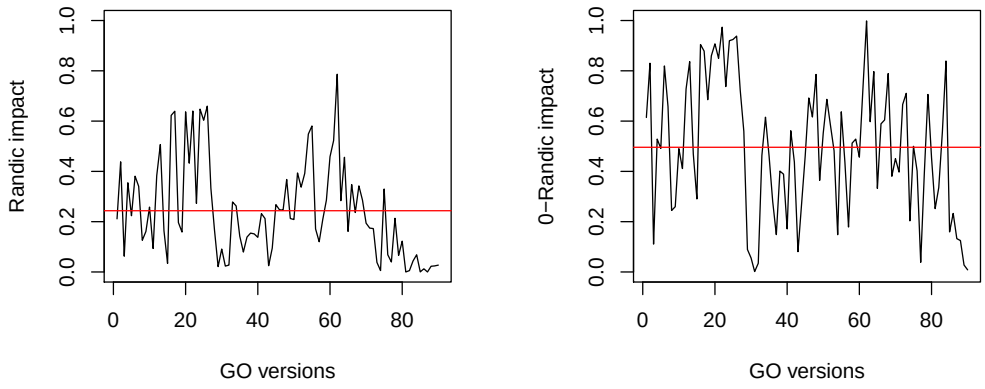
2.4.1 Definition of Impact

Figure 2.1 shows the distribution of the two calculated impact measures. Impact based on the Randić index shows a biased distribution towards zero, while impact based on ^{0}Randić index is centered and evenly distributed between 0 and 1. We set σ in Equation 2.1 equal to the number of change actions between the two versions. This normalizes the impact based on how many changes have been undergone by the ontology. This can be seen in Figure 2.1, where both measures of impact show a reasonable distribution.

Impact calculated based on the Randić Index is biased towards zero as seen in Figure 2.1a, whereas the impact that was calculated based on the ^{0}Randić Index is more centered and equally distributed, as shown in Figure 2.1b. In both figures, the red line represents the mean and the blue line the median.

Neither of these two impacts present a clean normal distribution. Therefore, we continue an analysis with both of these impact measures. This absence of a normal distribution can be explained by the small amount of data points. With more versions, the ^{0}Randić impact would likely evolve into a normal distribution since the peak at 0.5 is already present. For the Randić impact, we assume that we have not found the optimal σ. An optimal σ would shift the distribution further to the right, bringing the mean and median closer to 0.5.

Addressing the Research Question 5, we reject Hypothesis 5.1, because the distribution is skewed and the mean and median are not aligned. Since the mean and median are still within 0.1 of each other, we continue the investigation of the other research questions with this measure. We accept Hypothesis 5.2 because it shows a normal distribution in Figure 2.1b and also the alignment of the mean and median is given. Therefore, the impact can be represented with the Randić and ^{0}Randić Index and they both show promising distributions. In future work, we will use other



(a) Timeline of impact with Randić Index (b) Timeline of impact with ^{0}Randić Index

Figure 2.2: Timeline of the evolution of the two impacts in black and the mean impact in red.

notions of impact, that will contrast the two measures of this work, to be able to compare them to each other.

2.4.2 Effects on Impact

As explained in Section 2.3.2 and 2.3.3, measures were calculated and evaluated against the impact. Table 2.6 reports all the significant correlations between the features and the two impact measures. The first column states the feature that has been either calculated over the old version of the KG or is the aggregation or the change actions. The second column is the correlation of the feature and the impact calculated using the Randić index (R). The third column is the corresponding p-value. The forth and fifth columns are then the correlation and p-value of the feature and the impact calculated using the ^{0}Randić index. The last column reports the reference to the hypotheses that are being accepted because of the significant correlations.

Out of 95 features, 22 of them bring evidence that the corresponding hypotheses hold. Out of the 22 features, 15 can be accepted for both notions of impact. The accepted hypothesis concerning change action features are denoted in bold in Table 2.2. All correlations that are significant for both measures go in the same direction and are at most 0.11 apart from each other. This shows that the two impact measures do not contradict each other, where one measure claims high and the other low impact. Furthermore, there are features that describe the impact that are common among the two measures.

Answering Research Question 6, we found 12 change action features that correlate with the chosen impact measures. Additionally, 9 graph features correlate with the impact. Together, these features will be used for the prediction.

Feature	R **corr**	R **p-value**	${}^0R$ **corr**	${}^0R$ **p-value**	**H**
addInner_mean_closeness			−0.3131	0.0028	6.2f
addInner_std_closeness			−0.3345	0.0014	6.2g
actions_addLeaf			0.2705	0.0099	6.3a
actions_addSubGraph	0.5748	< 0.0001	0.5106	< 0.0001	6.4a
actions_merge	−0.3749	0.0003	−0.3595	0.0005	6.5a
merge_mean_degree			−0.3160	0.0077	6.5b
merge_std_degree	−0.3114	0.0087	−0.3686	0.0017	6.5c
actions_move	−0.4245	< 0.0001	−0.4951	< 0.0001	6.6a
actions_split	0.2900	0.0056			6.7a
actions_toObsolete	−0.3705	0.0003	−0.3168	0.0023	6.8a
actions_all	−0.3426	0.0009	−0.3535	0.0006	6.9a
all_std_closeness			−0.3372	0.0012	6.9g
vertex_count	−0.3366	0.0012	−0.2966	0.0045	6.10
edge_count	−0.3067	0.0033	−0.2768	0.0083	6.11
avg_degree_centr	0.3583	0.0005	0.3074	0.0032	6.13
avg_closeness	0.3849	0.0002	0.3113	0.0028	6.14
avg_between	0.3375	0.0011	0.2817	0.0072	6.15
transitivity	0.4199	< 0.0001	0.2966	0.0045	6.17
num_strong_components	−0.3366	0.0012	−0.2966	0.0045	6.21
centr_entropy	−0.3224	0.0019	−0.2789	0.0078	6.22
clos_entropy	−0.3221	0.0020			6.23
sparseness	0.3596	0.0005	0.3086	0.0031	6.24

Table 2.6: Significant correlations between impact and features and accepted hypotheses.

		R all	^{0}R all	R corr	^{0}R corr	R fs	^{0}R fs
GLM	RMSE	0.3353	0.3750	0.1026	0.1528	0.1007	0.1793
	R^2	0.1573	0.2706	0.7326	0.6683	0.7229	0.5997
SVM Lin	RMSE	0.6245	0.5623	0.1068	**0.1520**	0.1085	0.1957
	R^2	0.0269	0.1444	0.7116	**0.7272**	0.7158	0.5557
SVM Rad	RMSE	0.1579	0.2180	0.1083	0.1598	0.1187	0.1696
	R^2	0.4005	0.3535	0.7217	0.6687	0.6483	0.6331
RF	RMSE	0.1199	0.1840	0.1085	0.1750	0.1285	0.1854
	R^2	0.6378	0.5529	0.7061	0.6041	0.5988	0.5638

Table 2.7: Prediction evaluation table, displaying the RMSE and R^2 of all prediction models that were built.

2.4.3 Prediction of Impact

We built various models predicting the impact using four different machine learning approaches. Table 2.7 shows the RMSE and R-squared for the predictions of the different models. A 10-fold cross validation was used with the four algorithms general linear model (GLM), support vector machine with linear (SVM Lin) and radial kernel (SVM Rad) and random forest (RF). Figure 2.3, 2.4, 2.5, 2.6, 2.7 and 2.8 each show these four algorithms. The top left diagram shows the results of GLM, top right is SVM Lin, bottom left is SVM Rad and bottom left visualizes RF.

We started with the *all* model, where all calculated features are used. As expected, the model performs badly with linear approaches for both measures of impact. This can be seen in the Figure 2.3 and 2.4. The root squared mean error (RSME) is roughly 0.35 for the GLM and 0.59 for SVM Lin. R-squared, which reports the amount of variance that is being explained by the model, is extremely low, smaller than 0.3 for each of the models. Thus, we reject Hypotheses 7.1, 7.7, 7.2, and 7.8. The non-linear models perform slightly better. Randić impact is being predicted with a RSME of 0.1579 with the SVM Rad and 0.1199 with the random forest. However, the R-squared is not high enough at 0.4005 for SVM Rad and 0.6378 for RF to be able to accept Hypothesis 7.13 or Hypothesis 7.19. For the ^{0}Randić impact the models perform just as badly as for the linear case. With around 0.2 RMSE and around 0.4 R-squared, we reject Hypotheses 7.14 and 7.20. Therefore, all hypothesis concerning the *all* model were rejected.

The *corr* model performs significantly better as shown in Figure 2.5 and 2.6. With an average RSME of 0.106 and R-squared of 0.72 we should accept all four hypothesis concerning the *corr* feature model and the Randić impact, namely Hypothesis 7.3, 7.9, 7.15, and 7.21. However, because of the rejection of Hypothesis 5.1 we reject all four. The distribution of the Randić impact is skewed which makes it easier to predict. Because of the skewness, most values ly between 0 and 0.4. Therefore, by predicting a value in this range, a model is already going to perform well enough. This can be seen in the figures of the Randić predictions in Figure 2.3, 2.5, and 2.7 in the left column. All points are located in the lower left quadrant of the diagram, except of the first two prediction models, which are the GLM and SVM Linear with all features. The predictions of ^{0}Randić with *corr* are slightly worse. We

accept Hypothesis 7.10 because the RMSE is 0.1520 and R-squared is 0.7272. At the same time, this is the only prediction model that performed well enough. All other prediction models for this impact measures yielded unsatisfactory results and their hypotheses are rejected.

The last model is the feature model built by the ridge regression for feature selection. Results are displayed in Figure 2.7 and 2.8. With RMSE 0.1007 and R-squared 0.7229, the GLM achieved the best results for Randić impact. Again, we reject Hypothesis 7.5 because we rejected Hypothesis 5.1. Therefore, we also reject Hypothesis 7.11 of the SVM Lin model for Randić impact even though it shows a RMSE 0.1085 and R-squared 0.7158. The Hypotheses 7.17 and 7.24 are also rejected. None of the predictions for ^{0}Randić performed sufficiently, which leads to the rejection of Hypothesis 7.6, 7.12, 7.17 and 7.24.

We were able to accept 1 out of 24 prediction hypotheses. This one hypothesis and results are denoted in bold in Table 2.3 and 2.7 respectively. The conditions for acceptance are set very low and all of the accepted prediction models barely pass them. It is possible that the features do not capture the necessary aspects of the graphs and changes to predict the impact. It is also possible that the impact measure is not well suited for this task. More investigation is needed to determine which of these two aspects are responsible for the poor performance of the predictors.

We answer Research Question 7 with an affirmation. It is possible to predict the impact using descriptive features of the graph and change actions. However, the prediction could be improved in the future by adding features that indicate the difference in graph measures or focus on stability and semantics of the graph.

2.5 Conclusions

Using descriptive graph measures and change actions, we are able to predict the impact using general linear regression, support vector regression with a linear and radial kernel and a random forest. The impact was defined by the distance of the Randic Index of two materialized GO versions and also by the ^{0}Randic Index. Seven models performed well enough and reached an average RSME of 0.1125 and an average R-squared of 0.7197. However, only one model was accepted because we concluded that the Randić impact measures is not suitable for prediction based on its distribution.

The prediction models of the Randić impact perform better because of the skewed distribution. In future work we will define other notions of impact, that address different aspects of the graph, like the Wiener Index. It is focused on distance between nodes, rather than on the degree [8]. A distance measure that is specific for ontologies will also be included.

Other features will need to be included in the prediction models as well because this might be the other reason for the overall poor performance of the predictions. We expect them to improve when we add features that describe the change in graph measures. For this, we would take the difference of a measure between the new and old version of the graph. Features that describe the materialization could also be included, since they are much closer to the impact in terms of space. Prediction

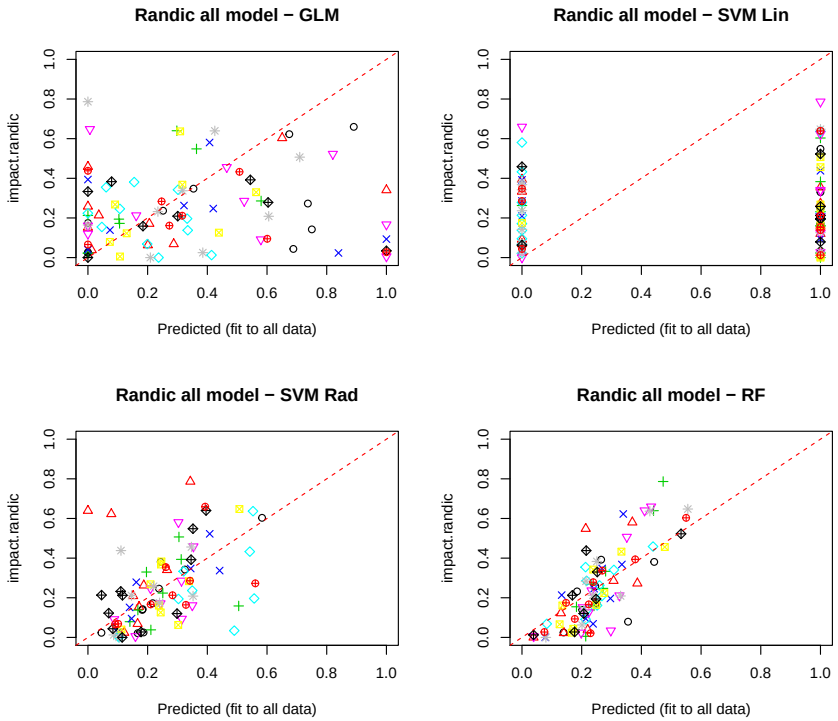


Figure 2.3: Predicted vs. real Randić impact with *all* model

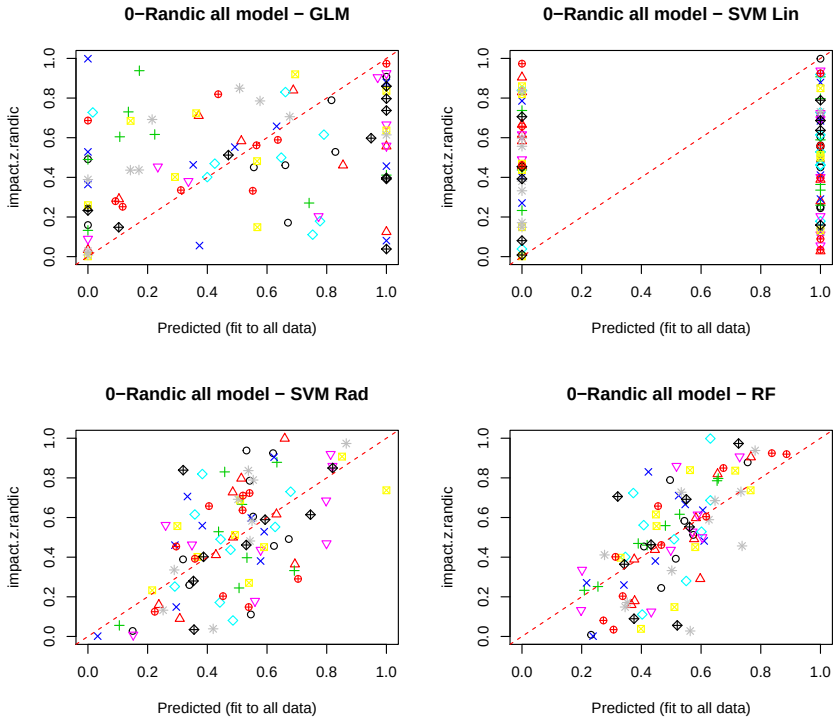


Figure 2.4: Predicted vs. real ^{0}Randić impact with *all* model

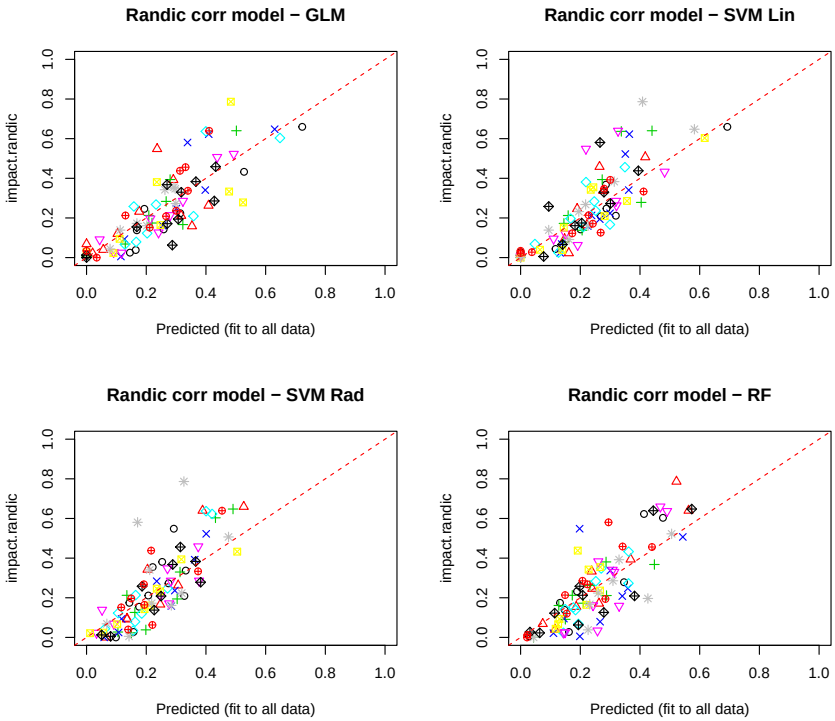


Figure 2.5: Predicted vs. real Randić impact with *corr* model

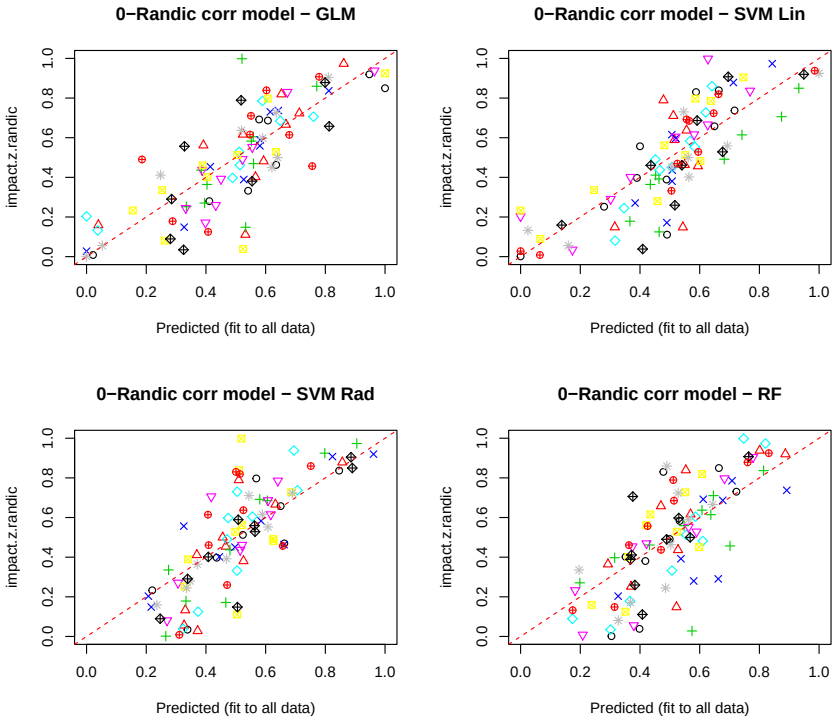


Figure 2.6: Predicted vs. real ^{0}Randić impact with *corr* model

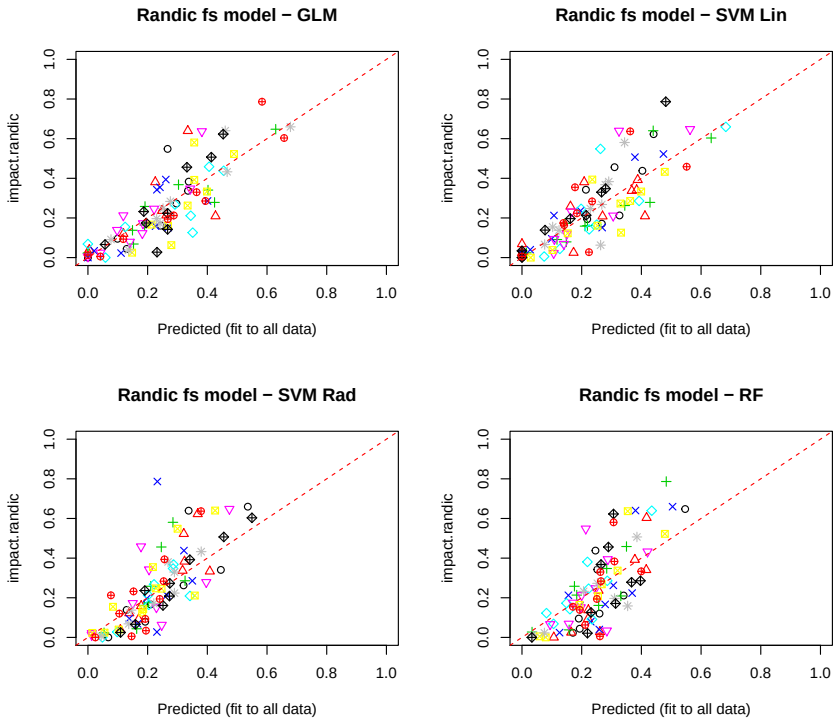


Figure 2.7: Predicted vs. real Randić impact with *fs* model

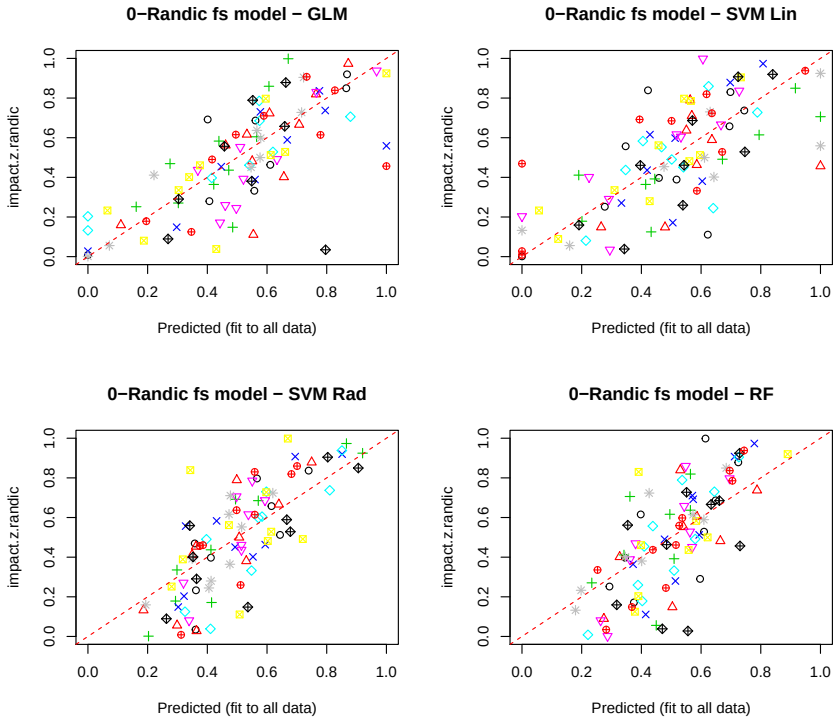


Figure 2.8: Predicted vs. real ^{0}Randić impact with *fs* model

models that include such features are being considered for future work.

Another limitation of this work and also an explanation of the poor results is a relatively small dataset. With numerous ontologies including detailed evolutionary data the relevance and generalisability of our findings can be increased. It is of interest to test the same hypotheses on a second KG preferably with many more versions than the GO. Future work will include the repetition of this study over a different data source and also the extension of the GO to be able to redo this analysis on versions that are closer together in terms of time. This would enable us to verify if our findings, generalize over the materialization operation, and draw proper conclusions on the methods and feasibility of our approach in predicting the impact.

References

[1] Algosaibi, A. A., and Melton, A. C. Three dimensions ontology modification matrix. In *2016 2nd International Conference on Information Management (ICIM)* (May 2016), pp. 77–83.

[2] Almeida, R. B., Mozafari, B., and Cho, J. On the Evolution of Wikipedia. In *ICWSM* (2007).

[3] Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. Gene Ontology: tool for the unification of biology. *Nature Genetics 25* (2000), 25 EP –.

[4] Chen, J., Lecue, F., Pan, J. Z., and Chen, H. Learning from Ontology Streams with Semantic Concept Drift. In *IJCAI* (2017), ijcai.org, pp. 957–963.

[5] Ciampaglia, G. L., Shiralkar, P., Rocha, L., Bollen, J., Menczer, F., and Flammini, A. Computational Fact Checking from Knowledge Networks. *PLoS ONE 10*, 6 (2015).

[6] David, J., and Euzenat, J. Comparison between Ontology Distances (Preliminary Results). In *The Semantic Web - ISWC 2008* (Oct. 2008), Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 245–260.

[7] Dehmer, M., and Emmert-Streib, F. Comparing large graphs efficiently by margins of feature vectors. *Applied Mathematics and Computation 188*, 2 (May 2007), 1699–1710.

[8] Dehmer, M., Emmert-Streib, F., and Shi, Y. Interrelations of Graph Distance Measures Based on Topological Indices. *PLOS ONE 9*, 4 (Apr. 2014), e94985.

[9] Dehmer, M., and Mowshowitz, A. A history of graph entropy measures. *Inf. Sci. 181*, 1 (2011), 57–78.

[10] Emmert-Streib, F., Dehmer, M., and Shi, Y. Fifty years of graph matching, network alignment and network comparison. *Information Sciences 346-347* (June 2016), 180–197.

[11] Fernández, J. D., Schneider, P., and Umbrich, J. The DBpedia wayback machine. In *SEMANTICS* (2015), ACM, pp. 192–195.

[12] Fluri, B., Würsch, M., Pinzger, M., and Gall, H. C. Change Distilling: Tree Differencing for Fine-Grained Source Code Change Extraction. *IEEE Trans. Software Eng. 33*, 11 (2007), 725–743.

[13] Goncalves, R. S., Parsia, B., and Sattler, U. Categorising logical differences between OWL ontologies. In *CIKM* (2011), ACM, pp. 1541–1546.

[14] Gross, A., Hartung, M., Prüfer, K., Kelso, J., and Rahm, E. Impact of ontology evolution on functional analyses. *Bioinformatics 28*, 20 (2012), 2671–2677.

[15] Hartung, M., Gross, A., and Rahm, E. CODEX: exploration of semantic changes between ontology versions. *Bioinformatics 28*, 6 (Mar. 2012), 895–896.

[16] Hartung, M., Gross, A., and Rahm, E. COnto-Diff: generation of complex evolution mappings for life science ontologies. *Journal of Biomedical Informatics 46*, 1 (Feb. 2013), 15–32.

[17] Heindorf, S., Potthast, M., Stein, B., and Engels, G. Towards Vandalism Detection in Knowledge Bases: Corpus Construction and Analysis. In *SIGIR* (2015), ACM, pp. 831–834.

[18] Ji, G., He, S., Xu, L., Liu, K., and Zhao, J. Knowledge Graph Embedding via Dynamic Mapping Matrix. In *ACL (1)* (2015), The Association for Computer Linguistics, pp. 687–696.

[19] Kaltenbrunner, A., and Laniado, D. There is No Deadline - Time Evolution of Wikipedia Discussions. *CoRR abs/1204.3453* (2012).

[20] Klein, M., and Noy, N. F. A component-based framework for ontology evolution. In *Workshop on Ontologies and Distributed Systems at IJCAI* (2003), vol. 3, p. 4.

[21] Lee, W.-N., Shah, N., Sundlass, K., and Musen, M. A. Comparison of Ontology-based Semantic-Similarity Measures. In *AMIA* (2008), AMIA.

[22] Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In *AAAI* (2015), AAAI Press, pp. 2181–2187.

[23] Lord, P. W., Stevens, R. D., Brass, A., and Goble, C. A. Semantic Similarity Measures as Tools for Exploring the Gene Ontology. In *Pacific Symposium on Biocomputing* (2003), pp. 601–612.

[24] Maedche, A., and Staab, S. Measuring Similarity between Ontologies. In *EKAW* (2002), vol. 2473 of *Lecture Notes in Computer Science*, Springer, pp. 251–263.

[25] Mestyán, M., Yasseri, T., and Kertesz, J. Early Prediction of Movie Box Office Success based on Wikipedia Activity Big Data. *CoRR abs/1211.0970* (2012).

[26] Nickel, M., Murphy, K., Tresp, V., and Gabrilovich, E. A Review of Relational Machine Learning for Knowledge Graphs. *Proceedings of the IEEE 104*, 1 (Jan. 2016), 11–33.

[27] Ren, Y., Parvizi, A., Mellish, C., Pan, J. Z., Deemter, K. v., and Stevens, R. Towards Competency Question-Driven Ontology Authoring. In *ESWC* (2014), vol. 8465 of *Lecture Notes in Computer Science*, Springer, pp. 752–767.

[28] Scott, J. *Social Network Analysis*. SAGE, Feb. 2017. Google-Books-ID: i5EmDgAAQBAJ.

[29] Stavropoulos, T. G., Andreadis, S., Kontopoulos, E., and Kompatsiaris, I. SemaDrift: A hybrid method and visual tools to measure semantic drift in ontologies. *Journal of Web Semantics* (June 2018).

[30] Steiner, T. Bots vs. Wikipedians, Anons vs. Logged-Ins (Redux): A Global Study of Edit Activity on Wikipedia and Wikidata. In *OpenSym* (2014), ACM, pp. 25:1–25:7.

[31] The Gene Ontology Consortium. Gene Ontology Consortium: going forward. *Nucleic Acids Research 43*, D1 (Jan. 2015), D1049–D1056.

[32] Trivedi, R., Dai, H., Wang, Y., and Song, L. Know-Evolve: Deep Temporal Reasoning for Dynamic Knowledge Graphs. In *ICML* (2017), vol. 70 of *Proceedings of Machine Learning Research*, PMLR, pp. 3462–3471.

[33] Tury, M., and Bieliková, M. An approach to detection ontology changes. In *ICWE Workshops* (2006), vol. 155 of *ACM International Conference Proceeding Series*, ACM, p. 14.

[34] Wu, Y., Agarwal, P. K., Li, C., Yang, J., and Yu, C. Toward Computational Fact-checking. *Proc. VLDB Endow. 7*, 7 (Mar. 2014), 589–600.

[35] Yang, Z., Zhang, D., and Ye, C. Evaluation Metrics for Ontology Complexity and Evolution Analysis. In *ICEBE* (2006), IEEE Computer Society, pp. 162–170.

[36] Yao, L., Zhang, Y., Wei, B., Li, L., Wu, F., Zhang, P., and Bian, Y. Concept over time: the combination of probabilistic topic model with wikipedia knowledge. *Expert Syst. Appl. 60* (2016), 27–38.

[37] Zhang, H., Li, Y.-F., and Tan, H. B. K. Measuring design complexity of semantic web ontologies. *Journal of Systems and Software 83*, 5 (May 2010), 803–814.

[38] Zhu, Q., Zhou, X., Tan, J., Liu, P., and Guo, L. Learning Knowledge Graph Embeddings via Generalized Hyperplanes. In *ICCS (1)* (2018), vol. 10860 of *Lecture Notes in Computer Science*, Springer, pp. 624–638.

A

Appendix

A.1 Contents of the CD

- German summary in an unformatted text-file
- Abstract in an unformatted text-file
- PDF-file of this master thesis
- Java code for the calculation of the materialization
- Java code for the calculation of the change actions
- Python code for the calculation of the graph measures
- Python code for the calculation of the change measures
- Python code for the calculation of the impact
- CSV files used in the analysis with R
- R scripts used for the analysis
- Text file of the feature selection results
- Text files with the results of the cross validation
- Latex files of this thesis

A.2 Change Action Hypotheses

H 6.2. *Addressing correlation between inner node additions and the impact:*

(a) The number of additions of inner nodes inside α correlates with the impact on $mat(K_{t_1})$?

(b) The mean degree of the additions of inner nodes inside α correlates with the impact on $mat(K_{t_1})$?

(c) The standard deviation of the degree of additions of inner nodes inside α correlates with the impact on $mat(K_{t_1})$?

(d) The mean degree centrality of the additions of inner nodes inside α correlates with the impact on $mat(K_{t_1})$?

(e) The standard deviation of the degree centrality of additions of inner nodes inside α correlates with the impact on $mat(K_{t_1})$?

(f) The mean closeness of the additions of inner nodes inside α correlates with the impact on $mat(K_{t_1})$?

(g) The standard deviation of the closeness of additions of inner nodes inside α correlates with the impact on $mat(K_{t_1})$?

(h) The mean betweenness of the additions of inner nodes inside α correlates with the impact on $mat(K_{t_1})$?

(i) The standard deviation of the betweenness of additions of inner nodes inside α correlates with the impact on $mat(K_{t_1})$?

H 6.3. *Addressing correlation between leave additions and the impact:*

(a) The number of additions of leaves inside α correlates with the impact on $mat(K_{t_1})$?

(b) The mean degree of the additions of leaves inside α correlates with the impact on $mat(K_{t_1})$?

(c) The standard deviation of the degree of additions of leaves inside α correlates with the impact on $mat(K_{t_1})$?

(d) The mean degree centrality of the additions of leaves inside α correlates with the impact on $mat(K_{t_1})$?

(e) The standard deviation of the degree centrality of additions of leaves inside α correlates with the impact on $mat(K_{t_1})$?

(f) The mean closeness of the additions of leaves inside α correlates with the impact on $mat(K_{t_1})$?

(g) The standard deviation of the closeness of additions of leaves inside α correlates with the impact on $mat(K_{t_1})$?

(h) The mean betweenness of the additions of leaves inside α correlates with the impact on $mat(K_{t_1})$?

(i) The standard deviation of the betweenness of additions of leaves inside α correlates with the impact on $mat(K_{t_1})$?

H 6.4. *Addressing correlation between subgraph additions and the impact:*

(a) The number of addition of subgraphs inside α correlates with the impact on $mat(K_{t_1})$?

(b) The mean degree of the additions of subgraphs inside α correlates with the impact on $mat(K_{t_1})$?

(c) The standard deviation of the degree of additions of subgraphs inside α correlates with the impact on $mat(K_{t_1})$?

(d) The mean degree centrality of the additions of subgraphs inside α correlates with the impact on $mat(K_{t_1})$?

(e) The standard deviation of the degree centrality of additions of subgraphs inside α correlates with the impact on $mat(K_{t_1})$?

(f) The mean closeness of the additions of subgraphs inside α correlates with the impact on $mat(K_{t_1})$?

(g) The standard deviation of the closeness of additions of subgraphs inside α correlates with the impact on $mat(K_{t_1})$?

(h) The mean betweenness of the additions of subgraphs inside α correlates with the impact on $mat(K_{t_1})$?

(i) The standard deviation of the betweenness of additions of subgraphs inside α correlates with the impact on $mat(K_{t_1})$?

H 6.5. *Addressing correlation between merge actions and the impact:*

(a) The number of merges inside α correlates with the impact on $mat(K_{t_1})$?

(b) The mean degree of merges inside α correlates with the impact on $mat(K_{t_1})$?

(c) The standard deviation of merges inside α correlates with the impact on $mat(K_{t_1})$?

(d) The mean degree centrality of merges inside α correlates with the impact on $mat(K_{t_1})$?

(e) The standard deviation of the degree centrality merges inside α correlates with the impact on $mat(K_{t_1})$?

(f) The mean closeness of merges inside α correlates with the impact on $mat(K_{t_1})$?

(g) The standard deviation of the closeness of merges inside α correlates with the impact on $mat(K_{t_1})$?

(h) *The mean betweenness of merges inside α correlates with the impact on $mat(K_{t_1})$?*

(i) *The standard deviation of the betweenness of merges inside α correlates with the impact on $mat(K_{t_1})$?*

H 6.6. *Addressing correlation between move actions and the impact:*

(a) *The number of moves inside α correlates with the impact on $mat(K_{t_1})$?*

(b) *The mean degree of moves inside α correlates with the impact on $mat(K_{t_1})$?*

(c) *The standard deviation of moves inside α correlates with the impact on $mat(K_{t_1})$?*

(d) *The mean degree centrality of moves inside α correlates with the impact on $mat(K_{t_1})$?*

(e) *The standard deviation of the degree centrality moves inside α correlates with the impact on $mat(K_{t_1})$?*

(f) *The mean closeness of moves inside α correlates with the impact on $mat(K_{t_1})$?*

(g) *The standard deviation of the closeness of moves inside α correlates with the impact on $mat(K_{t_1})$?*

(h) *The mean betweenness of moves inside α correlates with the impact on $mat(K_{t_1})$?*

(i) *The standard deviation of the betweenness of moves inside α correlates with the impact on $mat(K_{t_1})$?*

H 6.7. *Addressing correlation between split actions and the impact:*

(a) *The number of splits inside α correlates with the impact on $mat(K_{t_1})$?*

(b) *The mean degree of splits inside α correlates with the impact on $mat(K_{t_1})$?*

(c) *The standard deviation of splits inside α correlates with the impact on $mat(K_{t_1})$?*

(d) *The mean degree centrality of splits inside α correlates with the impact on $mat(K_{t_1})$?*

(e) *The standard deviation of the degree centrality splits inside α correlates with the impact on $mat(K_{t_1})$?*

(f) *The mean closeness of splits inside α correlates with the impact on $mat(K_{t_1})$?*

(g) *The standard deviation of the closeness of splits inside α correlates with the impact on $mat(K_{t_1})$?*

(h) *The mean betweenness of splits inside α correlates with the impact on $mat(K_{t_1})$?*

(i) *The standard deviation of the betweenness of splits inside α correlates with the impact on $mat(K_{t_1})$?*

H 6.8. *Addressing correlation between to-obsolete actions and the impact:*

(a) *The number of o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

(b) *The mean degree of o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

(c) *The standard deviation of o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

(d) *The mean degree centrality of o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

(e) *The standard deviation of the degree centrality o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

(f) *The mean closeness of o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

(g) *The standard deviation of the closeness of o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

(h) *The mean betweenness of o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

(i) *The standard deviation of the betweenness of o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

H 6.9. *Addressing correlation between overall actions and the impact:*

(a) *The number of all actions inside α correlates with the impact on $mat(K_{t_1})$?*

(b) *The mean degree of all actions inside α correlates with the impact on $mat(K_{t_1})$?*

(c) *The standard deviation of all actions inside α correlates with the impact on $mat(K_{t_1})$?*

(d) *The mean degree centrality of all actions inside α correlates with the impact on $mat(K_{t_1})$?*

(e) *The standard deviation of the degree centrality all actions inside α correlates with the impact on $mat(K_{t_1})$?*

(f) *The mean closeness of o-obsolete inside α correlates with the impact on $mat(K_{t_1})$?*

(g) *The standard deviation of the closeness of all actions inside α correlates with the impact on $mat(K_{t_1})$?*

(h) *The mean betweenness of all actions inside α correlates with the impact on $mat(K_{t_1})$?*

(i) *The standard deviation of the betweenness of all actions inside α correlates with the impact on $mat(K_{t_1})$?*

A.3 Prediction Hypotheses

A.3.1 Hypotheses with Linear Regression

H 7.1. *The Randić impact prediction on $mat(K_{t_1})$ using the all feature model and linear regression has a RMSE below 0.2 and R-squared above 0.7.*

H 7.2. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the all feature model and linear regression has a RMSE below 0.2 and R-squared above 0.7.*

H 7.3. *The Randić impact prediction on $mat(K_{t_1})$ using the correlation feature model and linear regression has a RMSE below 0.2 and R-squared above 0.7.*

H 7.4. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the correlation feature model and linear regression has a RMSE below 0.2 and R-squared above 0.7.*

H 7.5. *The Randić impact prediction on $mat(K_{t_1})$ using the fs model and linear regression has a RMSE below 0.2 and R-squared above 0.7.*

H 7.6. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the fs model and linear regression has a RMSE below 0.2 and R-squared above 0.7.*

A.3.2 Hypotheses with Support Vector Regression, Linear Kernel

H 7.7. *The Randić impact prediction on $mat(K_{t_1})$ using the all feature model and support vector regression with a linear kernel has a RMSE below 0.2 and R-squared above 0.7.*

H 7.8. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the all feature model and support vector regression with a linear kernel has a RMSE below 0.2 and R-squared above 0.7.*

H 7.9. *The Randić impact prediction on $mat(K_{t_1})$ using the correlation feature model and support vector regression with a linear kernel has a RMSE below 0.2 and R-squared above 0.7.*

H 7.10. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the correlation feature model and support vector regression with a linear kernel has a RMSE below 0.2 and R-squared above 0.7.*

H 7.11. *The Randić impact prediction on $mat(K_{t_1})$ using the fs model and support vector regression with a linear kernel has a RMSE below 0.2 and R-squared above 0.7.*

H 7.12. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the fs model and support vector regression with a linear kernel has a RMSE below 0.2 and R-squared above 0.7.*

A.3.3 Hypotheses with Support Vector Regression, Radial Kernel

H 7.13. *The Randić impact prediction on $mat(K_{t_1})$ using the all feature model and support vector regression with a linear radial has a RMSE below 0.2 and R-squared above 0.7.*

H 7.14. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the all feature model and support vector regression with a radial kernel has a RMSE below 0.2 and R-squared above 0.7.*

H 7.15. *The Randić impact prediction on $mat(K_{t_1})$ using the correlation feature model and support vector regression with a radial kernel has a RMSE below 0.2 and R-squared above 0.7.*

H 7.16. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the correlation feature model and support vector regression with a radial kernel has a RMSE below 0.2 and R-squared above 0.7.*

H 7.17. *The Randić impact prediction on $mat(K_{t_1})$ using the fs model and support vector regression with a radial kernel has a RMSE below 0.2 and R-squared above 0.7.*

H 7.18. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the fs feature model and support vector regression with a radial kernel has a RMSE below 0.2 and R-squared above 0.7.*

A.3.4 Hypotheses with Random Forest

H 7.19. *The Randić impact prediction on $mat(K_{t_1})$ using the all feature model and random forest has a RMSE below 0.2 and R-squared above 0.7.*

H 7.20. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the all feature model and random forest has a RMSE below 0.2 and R-squared above 0.7.*

H 7.21. *The Randić impact prediction on $mat(K_{t_1})$ using the correlation feature model and random forest has a RMSE below 0.2 and R-squared above 0.7.*

H 7.22. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the correlation feature model and random forest has a RMSE below 0.2 and R-squared above 0.7.*

H 7.23. *The Randić impact prediction on $mat(K_{t_1})$ using the fs model and random forest has a RMSE below 0.2 and R-squared above 0.7.*

H 7.24. *The ^{0}Randić impact prediction on $mat(K_{t_1})$ using the fs model and random forest has a RMSE below 0.2 and R-squared above 0.7.*

A.4 Data

version	actions		version	actions		version	actions
2010-01	220		2013-10	217		2017-05	158
2010-02	403		2013-11	278		2017-06	261
2010-05	369		2013-12	288		2017-07	253
2010-06	60		2014-01	166		2017-08	445
2010-07	166		2014-02	132		2017-11	411
2010-08	203		2014-03	234		2017-12	165
2010-09	480		2014-04	185		2018-01	58
2010-10	362		2014-05	320		2018-03	213
2010-11	268		2014-06	104			
2010-12	274		2014-07	195			
2011-01	333		2014-08	294			
2011-02	210		2014-09	187			
2011-03	176		2014-10	193			
2011-04	292		2014-11	152			
2011-05	211		2014-12	177			
2011-06	161		2015-01	250			
2011-07	149		2015-02	169			
2011-09	175		2015-03	255			
2011-10	200		2015-04	123			
2011-11	91		2015-05	141			
2011-12	330		2015-06	158			
2012-02	225		2015-07	197			
2012-03	283		2015-08	129			
2012-04	249		2015-09	161			
2012-05	280		2015-10	137			
2012-06	251		2015-11	103			
2012-07	295		2015-12	62			
2012-08	342		2016-01	229			
2012-09	388		2016-02	94			
2012-10	524		2016-03	287			
2012-11	949		2016-04	250			
2012-12	277		2016-05	151			
2013-01	172		2016-06	1157			
2013-02	242		2016-07	98			
2013-03	172		2016-10	401			
2013-04	135		2016-11	770			
2013-05	288		2016-12	125			
2013-06	240		2017-01	440			
2013-07	183		2017-02	238			
2013-08	214		2017-03	302			
2013-09	199		2017-04	254			

List of Figures

List of Tables