Collaborative Project

FROCKG - Fact Checking for Large Enterprise Knowledge Graphs

Project Number: E FROCKG Start Date of Project: 2020/01/01 Duration: 36 months

Deliverable 1.3: Requirements specification

Dissemination Level	Public
Due Date of Deliverable	Month 6, 2020/06/30
Actual Submission Date	Month 6, 2020/06/30
Work Package	WP1, Requirements Elicitation
Task	T1.2
Type	Report
Approval Status	Work in progress
Version	1.0
Number of Pages	23

Abstract:

This report presents the final version of the requirement specification for the FROCKG use cases. It presents the deduced requirements per use case as well as user stories describing the use cases in more details from an end user perspective.

Project by Eurostars.

History

Version	Date	Reason	Revised by
1.0	30/06/2020	Draft revised	Wolfgang Schell

Author List

Organization	Name	Contact Information
metaphacts GmbH	Peter Haase	ph@metaphacts.com
metaphacts GmbH	Wolfgang Schell	ws@metaphacts.com
metaphacts GmbH	Jeen Broekstra	jb@metaphacts.com
Paderborn University	Michael Röder	michael.roeder@upb.de
Sirma AI EAD	Nikola Rusinov	nikola.rusinov@ontotext.com
Sirma AI EAD	Todor Primov	todor.primov@ontotext.com
Zazuko GmbH	Adrian Gschwend	adrian.gschwend@zazuko.com

Contents

Introduction

The requirements presented below are derived by first formulating user stories for each use case and then breaking each down in requirements. There is an overlap of requirements from multiple user stories which shows that they have some common functionality and can be combined to cover multiple tasks and user stories. This also means that the requirements from the current user stories will likely be able to cover additional user stories in the future and therefore provide a future-proof base for evolution of the solutions of the Cultural Heritage, Pharma Fake News Detection, and Linked Open Data use cases.

The requirements are listed with a priority which defines the importance of its implementation. The following priorities are defined:

MUST	This requirement must be satisfied in the overall solution/implementation
SHOULD	This requirement should be satisfied in the overall solution/implementation unless there is a really important reason not to do it, e.g. legal obligations, project risks, etc.
CAN/OPTIONAL	This requirement should be satisfied if possible, but can be safely skipped if there is no time or some other reason not to implement it
NICE-TO-HAVE	This requirement shows some useful functionality and is mostly documented for future extension.

Use Case Descriptions

Pharma Fake News Detection

In the pharmaceutical industry, the product to market process is long, research and development costs are high, and often the overall success is put under high risk. Thus, the results from clinical trials are often identified as key inflection points in the process around which investors and business as overall receive early indications on drug product's chances for success. The release of clinical trial results has meaningful effects on market value for biopharmaceutical companies.
Creation and spreading of both "positive" and "negative" fake news about performance of clinical drug products is a common practice to manipulate the stock market and to achieve easy gains due to rapid drop or raise in the stock shares prices. Thus being able to tell real from fake news is of big importance to avoid falling for market manipulation and act only on factual and proven information.

User Stories

ID	PFN-1
Title	Drug product knowledge graph
Summary	As a pharmaceutical industry domain expert I would like to have a comprehensive drug product dossiers collected from public authoritative sources
Target user group / persona	Pharmaceutical industry domain expert (could be involved in marketing, portfolio management, drug safety)
Entry point	Search and Exploration UI
User intent, motivation	Discover up-to-date drug product information from authoritative sources
Description / flow	1. the user performs drug product search in data exploration application 2. [optional] the user performs filtering of results 3. the user selects a particular drug product 4. the drug product dashboard is loaded
Involved components	● Knowledge graph ● Visualization/Exploration components
Inputs and Outputs	Input: ● Search criteria (objects from the knowledge graph) ● Filtering criteria (objects from the knowledge graph) Output: ● Drug product sub-graph rendered as a dashboard
Additional comments	Build KG based on external datasets (such as DrugBank, clinicaltrials.gov), e.g. using NLP and data integration pipelines, incl. data normalization to be used as reference data set.

ID	PFN-2
Title	Annotation of news article
Summary	As a pharmaceutical industry domain expert I would like to extract all important facts (termination vs. completion of a clinical trial; withdrawal of a drug from market; reported new serious adverse events; etc) about a drug product mentioned in a single article

Target user group / persona	Pharmaceutical industry domain expert (could be involved in marketing, portfolio management, drug safety)
Entry point	Annotation module
User intent, motivation	Discover new important facts about a drug product mentioned in news articles.
Description / flow	1. the user load a single news article 2. the system automatically annotate the content with named entities and relations between them 3. [optional] the user can validate/correct some of the automatically created annotations 4. both the annotated document and the extracted facts as triples are saved in the system
Involved components	● Knowledge graph ● Text analysis pipeline ● Visualization component ● Annotation functionality
Inputs and Outputs	Input: ● News article content (text) Output: ● Annotated document (GATE XML with inline annotations) ● Extracted facts as triples (RDF)
Additional comments	Articles may also be provided via feeds or other sources and ingested automatically.

ID	PFN-3
Title	Validation of extracted facts
Summary	As a pharmaceutical industry domain expert I would like to easily validate the correctness of the facts mentioned in a news article towards the currently available information from authoritative sources.
Target user group / persona	Pharmaceutical industry domain expert (could be involved in marketing, portfolio management, drug safety)
Entry point	Annotation module
User intent, motivation	Validate extracted facts
Description /	1. the user load an annotated article

flow	2. the user requests validation of the identified facts 3. the extracted facts are validated/invalidated based on the information available in the knowledge graph
Involved components	● Knowledge graph ● Visualization/Exploration components ● Fact checking component
Inputs and Outputs	Input: ● Annotated document (GATE XML with inline annotations) ● Extracted facts as triples (RDF) Output: ● List of validated/invalidated facts as triples (RDF)
Additional comments	Articles may also be validated automatically after ingestion from feeds or other sources. Example for fact check: Claim: Drug XYZ was stopped in clinical trial phase 3 because it failed to demonstrate efficacy in treatment of disease ABC Fact Check: ● is there information about a clinical trial for this drug and indication recorded in the reference data set? Comments: ● this assumes the reference data set is a closed world, i.e. the KG would know about ALL clinical trials and if the trial is not known we assume the fact is invalid

ID	PFN-4
Title	Monitoring of mentioned facts for a drug product in the news
Summary	As a pharmaceutical industry domain expert I would like to have an overview of all important facts mentioned in the news and identify possible relations between them.
Target user group / persona	Pharmaceutical industry domain expert (could be involved in marketing, portfolio management, drug safety)
Entry point	Search and exploration UI
User intent, motivation	Validate extracted facts
Description / flow	1. the user performs drug product search in data exploration application 2. [optional] the user performs filtering of results 3. the user selects a particular drug product

	4. the drug product dashboard is loaded 5. the user select the news monitoring module
Involved components	● Knowledge graph ● Visualization/Exploration components ● Fact checking component
Inputs and Outputs	Input: ● Search criteria (objects from the knowledge graph) ● Filtering criteria (objects from the knowledge graph) Output: ● Sub-graph of all drug product related extracted facts from news articles with provenance sources (original news articles)
Additional comments	none

Requirements

ID	Title	Description	Priority
PFN-1	Core ontologies	Integrate core ontologies and terminologies for semantic normalization of datasets	Must
PFN-1	Key data sets	Integrate key domain rich datasets covering drug products, clinical trials and reported adverse events	Must
PFN-1 and PFN-4	Faceted search	Provide faceted search user interface that will allow identification of precise subsets of entities based on their properties	Must
PFN-1	Graph exploration	Provide graph exploration capabilities that will allow navigation among closely related objects in the graph	Can
PFN-1 and PFN-4	Drug product centric dashboard	Provide analytical dashboards with aggregated data for a drug product	Must
PFN-2	News feeder	Configurable feeder that will regularly retrieve the newly published content on a collection of pharma news portals.	Must

PFN-2	Article annotation service	A NLP pipeline that will process the content of pharma news articles and will create semantic annotations based on the KG	Must
PFN-2	RDF-ization service	An ETL service that will transform the semantic annotations into RDF	Must
PFN-2 and PFN-3	Manual content annotation/validation module	A visualization module for the annotated content with optional functionality for manual annotation and validation of automatic extractions.	Can
PFN-3	Fact validation service	An automated fact validation service that will check all extracted facts from an article against the knowledge available in the KG	Must

Cultural Heritage and Archiving

Organizations such as galleries, libraries, archives, and museums (GLAM sector) collect information on historical and current events, people, places, as well as artifacts and works of art for research, presentation, preservation, and creating narratives. Also, In the last decade, public archives started to publish their catalogs on the web.
As this information is collected from sources of different quality, provenance and trustworthiness, it is important to verify any facts gathered and cross-reference them with other known facts or pieces of information.

User Stories

ID	CHA-1
Titel	Fact checking from exploration
Summary	As a researcher I want to verify a fact while exploring a public dataset such as Wikidata in order to provide evidence for a theory/statement/finding.
Target user group / persona	Researcher
Entry point	Dashboard or Exploration UI
User intent, motivation	A researcher wants to document a finding and provide evidence that proves this statement
Description / flow	1. explore data in metaphactory UI, e.g. from search or through graphical exploration using Ontodia 2. when a certain link/fact was identified, the user starts an action

	(e.g. via context menu or button) to verify that fact 3. The system calls the Fact Checking module which provides feedback on veracity together with an explanation 4. The user can store the fact checking results as an annotation, e.g. using the W3C annotation model (possibly using RDF*) using an annotation UI. 5. The user may also provide additional data such as comments or references, e.g. through manual editing, lookup of reference data - see CHA-2, CHA-4 6. the created (or edited) annotation is visualized within the exploration UI from which the fact checking was triggered - see CHA-3.
Involved components	● Visualization/Exploration components ● Fact Checking service ● Annotation component ● Authoring component (form-based) ● LookupService
Inputs and Outputs	Input: ● fact(s)/triple(s) to check, possibly multiple facts combined into a “group” or shape ● optionally additional control or meta information, e.g. level of check or target data set to check against, ... Output: ● verification and explanation from FactChecking component ● Annotation data
Additional comments	Depending on the responsiveness of the FactChecking operation this might need to be an asynchronous task. If so, this would require additional thought on how to visualize the pending operation and offer revisiting the results.

ID	CHA-2
Title	Manual annotation of facts and knowledge bases
Summary	As a researcher I want to manually provide additional data such as comments or references on facts in order to provide evidence for a theory/statement/finding.
Target user group / persona	Researcher
Entry point	Dashboard or Exploration UI
User intent, motivation	A researcher wants to document a finding and provide evidence that proves this statement
Description /	1. explore data in metaphactory UI, e.g. from search or through

flow	graphical exploration using Ontodia 2. when a certain link/fact was identified, the user starts an action (e.g. via context menu or button) to annotate that fact 3. The user can store the annotation, e.g. using the W3C annotation model (possibly using RDF*) using an annotation UI.
Involved components	● Visualization/Exploration components ● Annotation component ● Authoring component (form-based) ● LookupService
Inputs and Outputs	Input: ● fact(s)/triple(s) to check, possibly multiple facts combined into a “group” or shape ● additional control or meta information, e.g. level of check or target data set to check against, ... Output: ● Annotation data
Additional comments	

ID	CHA-3
Title	Visualization of annotations
Summary	As a researcher I want to get a visual overview of fact checking results and other annotations, so that I can quickly explore these annotations.
Target user group / persona	Researcher
Entry point	Dashboard or Exploration UI
User intent, motivation	A researcher wants to explore a dataset and the associated fact checking results and other available annotations.
Description / flow	
Involved components	● Visualization/Exploration components ● Annotation component
Inputs and Outputs	
Additional comments	

ID	CHA-4
Titel	Cross-linking of data
Summary	As an author I want to lookup data from verified sources in order to cross-reference content I am authoring.
Target user group / persona	Author / Content provider
Entry point	Dashboard or Exploration UI
User intent, motivation	An author wants to provide proper canonical identifiers of authored data or references.
Description / flow	1. start editing data about an event 2. entering of information such as event name, description etc. using a form-based approach 3. when referring to other entities such as people, places, organizations, etc. data should be cross-referenced with canonical identifiers so they are properly linked. This can be done by searching for the entities based on searching for them in a reference data set and providing additional help such as auto-suggestion and disambiguation based on additional information about the candidate entities, such as type, description, etc.
Involved components	● Authoring component (form-based) ● LookupService
Inputs and Outputs	Input: ● search terms ● optionally additional control or meta information, e.g. type of entity to search for Output: ● result candidates
Additional comments	

ID	CHA-5
Titel	Data validation
Summary	As a researcher I want to ensure that data being edited is valid and complete
Target user group / persona	Author / Content provider

Entry point	Authoring UI
User intent, motivation	A researcher wants to document a finding and provide evidence that proves this statement
Description / flow	1. start editing data about an event 2. entering of information such as event name, description etc. using a form-based approach 3. when saving data the form is validated to ensure it adheres to defined structure, rules and constraints for the target type. This can be defined in so-called shapes, e.g. using the SHACL notation
Involved components	● Authoring component (form-based) ● ValidationService ● SPARQL engine/graph database (e.g. GraphDB)
Inputs and Outputs	Input ● authored data (e.g. contents in a form) Output ● validation result, e.g. hints for missing or invalid data, formatting/syntax (timestamps, numbers, …), missing references
Additional comments	

ID	CHA-6
Titel	Use KGs to annotate 3rd party datasets
Summary	As a historian, I want to annotate an archival record so that I can link it to people, events and places in existing knowledge graphs.
Target user group / persona	Historian, museum curator, art curator, etc.
Entry point	UI that presents an archival record, art work, etc.
User intent, motivation	Descriptions of those records might be very minimal, anyone interacting with a record might add additional information to improve search, interlinking, etc.
Description / flow	1. Explore records in the search/view UI 2. Bookmark the record if it's something the user wants to inspect closer (for example dive into the archival record itself, not just the metadata) 3. Come back to the record to annotate it 4. Annotate people/places/events

Involved components	● Search/Explore UI ● Component that provides details of a search result ● Bookmark component ● Annotation component
Inputs and Outputs	Input ● URI of a record that is to be annotated ● Type of annotation (people/place/event) ● Context that can be found at the annotation URI, like date, plaintext description, existing tags and annotations, etc. Output ● An annotation body ● A veracity of this annotation
Additional comments	

Requirements

ID	Title	Description	Priority
CHA-1 CHA-6	Fact annotation dialog	Enable displaying and editing of fact annotations	Must
CHA-1	Fact checker integration	Interact with the fact checking service to retrieve fact checking results	Must
CHA-1	Fact checking result persistence	Enable persisting fact checking annotation independently from the dataset being annotated	Must
CHA-1	Fact checking annotation model	A consistent modeling approach and shared vocabulary for rich annotation and fact checking results	Must
CHA-1 CHA-2 CHA-3 CHA-6	annotation exploration component	Enable exploration of existing annotations and fact checking results	Should
CHA-2 CHA-3 CHA-5	Validation integration	Validate newly (manually) entered data against predefined constraints	Must
CHA-2 CHA-3 CHA-6	Dynamic editing components	Dynamically generate UI components for visualizing and editing of knowledge graphs and annotations	Should

CHA-4	Entity Lookup/Search	Search/Lookup of entities based on search term (e.g. name) and type	Should
CHA-6	Bookmarking of facts	Enable bookmarking and visiting bookmarks	Should

Linked Open Data

The amount of Linked Open Data available on the web grows continuously. This data is readily available and used by companies to offer services. Since the datasets in the Linked Open Data Cloud can be published by any arbitrary body, it is crucial to be able to figure out whether a dataset that is available online contains true facts or spreads erroneous knowledge. Therefore, it is necessary to check Linked Open Datasets regarding their veracity before using them.

The aim of this use case is consequently to offer an easy-to-deploy Fact Checking platform and corresponding REST interfaces that can be used to check either single facts or complete datasets based on user-defined reference data (i.e., a reference corpus or a reference dataset). With this framework, we aim to demonstrate the usefulness of FROCKG technologies for consumers of Linked Open Data based on a subset of the Linked Open Data cloud by tackling the main challenges of checking the veracity of Linked Open Datasets and using Linked Open Datasets as reference knowledge base.

User Stories

ID	LOD-1
Titel	Veracity check of an open knowledge base
Summary	As a user of the linked open data available on the web, I want to be able to check the veracity of a knowledge base so that I can ensure that the knowledge base can be used by me. As a provider of an open knowledge base, I want to ensure the veracity of my knowledge base so that I can ensure the usefulness of my data and, hence, a high impact of the knowledge base.
Target user group / persona	Users of Linked Open Data, e.g., developers of systems or services relying on such data. Providers of Linked Open Data.
Entry point	On a machine that contains the knowledge base the user would like to check.

User intent, motivation	Before making use of a knowledge base, it is good to check whether it is trustworthy or not. Before publishing and advertising a newly created knowledge base, the user would like to ensure the veracity of the created data.
Description / flow	1. Setup the FROCKG system 2. Choose reference knowledge bases / corpora 3. Execute knowledge graph veracity check
Involved components	1. Knowledge graph veracity component
Inputs and Outputs	Input: - knowledge base that should be checked Output: - veracity score - explanation of the score
Additional comments	

ID	LOD-2
Titel	Veracity check of newly added facts
Summary	As an editor of a dataset, I want to be able to check the veracity of facts that are added to the dataset to avoid mistakes. As a provider of Linked Open Data that is manually curated, I would like to ensure that the work of the editors is checked in an efficient way without adding too much manual effort.
Target user group / persona	Editors of RDF datasets. Providers of Linked Open Data.
Entry point	We assume that the user has some kind of editor to add new facts or edit existing facts of the knowledge graph.
User intent, motivation	Before changes (added or edited facts) take effect, the newly added facts are checked and the veracity of edited facts is compared to the veracity they had before editing.
Description / flow	1. The editor edits facts and submit additions and changes. 2. Before the changes take effect, the newly added facts are checked and the veracity of edited facts is compared to the veracity they had before editing. To this end, the facts are sent to a web service that returns the veracity values. a. For newly added facts, a veracity threshold should be

	defined. b. For edited facts, the veracity should increase compared to their previous version. 3. If changes are fine, they are applied to the knowledge base. If the veracity values are not good enough, the changes are paused and need a further, manual check.
Involved components	1. Fact Checking component
Inputs and Outputs	Input: - newly added facts - new and old version of edited facts Output: - veracity score - explanation of the scores, which can be used by a human reviewer to understand the decision of the algorithm to stop the update process
Additional comments	- One of the challenges of this user story is the handling of deleted facts. It might be worth having a system that suggests to stop deleting a fact if the fact is true. - This user story covers several functionalities of a tool to edit RDF knowledge graphs. Not all of these functionalities are directly related to the FROCKG project. To this end, it will be made sure that an implementation of the user story within the FROCKG project focuses on triggering the fact checking workflow and integrating its result into the editing workflow, e.g., by extending an existing tool.

ID	LOD-3
Titel	LOD veracity overview
Summary	As a user of Linked Open Data, I would like to get an overview of the veracity of available datasets without being forced to run the evaluation by myself. Instead, I would like to have a web page or an online portal which shows me the latest results for publicly available datasets. In the best case, I could even add a dataset to be checked.
Target user group / persona	Consumers and providers of Linked Open Data.
Entry point	Website of the fact checking portal
User intent, motivation	Get an overview of the veracity of publicly accessible knowledge bases before using them.
Description / flow	The user can browse through the different knowledge bases and get an overview of their veracity. The single veracity scores are further explained using natural language.

	In addition, the user could add his/her own dataset to be checked.
Involved components	● A portal component that allows the user interaction ● The knowledge base veracity component as backend / provider of the scores
Inputs and Outputs	Input: - Linked Open datasets Output: - veracity score - explanation of the scores
Additional comments	This use case potentially provides a good demonstration use case of the final project results.

Requirements

ID	Title	Description	Priority
LOD-1	Fact checking service	A service has to be provided that takes a single fact (or a set of facts), checks it and returns a veracity value and an explanation. The exact methods will be defined in the LOD-1.x requirements	MUST
LOD-1.1	Fact checking method (resources)	The service must provide a method for checking a single triple. The fact comprises a triple of URI resources. Input: ● Fact (s, p, o) where s, p and o are URI resources. For the resources, the following must hold: ○ All three resources have to be available in the reference KB ○ A reference corpus has been given, the subject and object come with labels, the Fact Checking algorithm has been trained for the property of the fact Output: ● veracity score ● natural language explanation	MUST
LOD-	Temporal fact checking	The service must provide a method for	CAN

1.2	method (resources)	checking a single triple that comes with additional temporal information. The fact comprises a triple of URI resources. Input: ● Annotated fact (s, p, o) where s, p and o are URI resources. The annotation contains temporal information. For the resources, the following must hold: ○ All three resources have to be available in the reference KB ○ A reference corpus has been given, the subject and object come with labels, the Fact Checking algorithm has been trained for the property of the fact Output as LOD-1.1	
LOD-1.3	Fact checking method for sub graphs (resources)	It would be nice if the service could provide a method for checking a set of triples. The subset comprises triples with URI resources. Input: ● set of facts where all subjects, predicates and objects are URI resources. For the resources, the following must hold: ○ All resources have to be available in the reference KB ○ A reference corpus has been given, the subject and object come with labels, the Fact Checking algorithm has been trained for the properties of the fact Output as LOD-1.1	NICE-TO-HAVE
LOD-2	Knowledge Base veracity service	A service has to be provided that takes a knowledge base, checks it and returns a veracity value and an explanation. Input: ● Access to the KB that should be checked (e.g., in form of an uploaded file or a SPARQL endpoint). ● A reference KB ● A reference corpus has been given, the subject and object come with labels, the Fact Checking algorithm has been trained for the property of	MUST

		the fact Output: ● veracity score ● natural language explanation	

Evaluation of Modules

To drive the development cycle, we aim at an agile and test-driven development approach supported by continuous testing and deployment mechanisms.

Evaluation criteria for WP2 and WP3

The Fact Checking services developed within WP2 are constantly benchmarked using established benchmarking datasets as well as use-case-related datasets that are created during the project. As a benchmarking framework, GERBIL-KBC[1] is used to measure the effectiveness of the single Fact Checking approaches. The main metric, which will be used for that, is the area under the curve of the ROC curve (ROC-AUC) that is created based on sorting the single facts of the benchmarking dataset with respect to their veracity score and comparing this order with their true veracity label. We expect that the services developed within the project have an effectiveness close to the start of the art for established benchmarks.[2] For measuring the efficiency, the average runtime per fact will be used.

The explanation generation component developed within WP3 will be evaluated using established measures implemented within GERBIL-MT.[3] These measures are BLEU, BLEU NLTK, METEOR, chrF++ and TER. All these measures are based on the comparison of the generated explanation with a given, manually created explanation. To this end, we will create a benchmark dataset based on the use cases of the project.

[1] http://gerbil-kbc.aksw.org/gerbil/
[2] Zafar Habeeb Syed, Michael Röder, and Axel-Cyrille Ngonga Ngomo: “FactCheck: Validating RDF Triples using Textual Evidence”. In Proceedings of the International Conference on Information and Knowledge Management (CIKM), 2018.
[3] https://github.com/dice-group/gerbil/tree/gerbil-MT

Evaluation criteria for WP4

The components developed as part of WP4 will be evaluated for their usability and composability and flexibility for various scenarios as described above.

Based on the requirements and user stories described above we will conduct testing with end users from the target audience of the respective use cases to ensure that the provided building blocks and tools meet the following non-functional requirements:

- completeness, i.e. they suffice to perform the required tasks
- usability, i.e. they follow a logical and easy to understand flow
- performance, i.e. the tasks can be performed in an efficient manner

Additionally, the provided components and building blocks will be evaluated regarding their flexibility, i.e. the ability to be used to compose them in a different way to implement new related user stories. This ensures that the provided solutions not just satisfy the exact requirements specified in this document but are also applicable for new uses with minimal adjustments.

Evaluation criteria for WP5

Gold Standard Corpus

A central challenge of semantic annotation is the lack of objective standards for assessing the success of the process of converting data to higher-level information.

Once the initial corpus is annotated by at least two people, the GOLD standard corpus can be created. It is the final version of the annotated data. It uses the most up-to-date specifications that were created during the annotation process, and it has everything tagged correctly according to the most recent guidelines.
Note that the GOLD standard does not assess whether the system accurately classifies phenomena, but the extent to which the system agrees with the human annotators where those classifications are concerned. Here it is important to decide what level of agreement is acceptable knowing that humans are bound to make mistakes.

Evaluation Metrics

The evaluation metric mathematically defines how to measure the system's performance against the manually annotated gold standard:

- Precision - what percentage of the annotations proposed by the system are correct (when compared to the gold standard)
- Recall - what percentage of the annotations in the gold standard were correctly identified by the system

- F- score - an average that rewards precision and recall values that are close together. (For many annotation tasks, we are interested in obtaining high levels of precision in conjunction with high, or at least reasonable, levels of recall)

Sometimes traditional methods for IE are not sufficient for ontology-based IE as the distinction between right and wrong is less obvious. For example, recognising a Person as a Location is clearly wrong, but recognising a Research Assistant as a Lecturer is not so wrong.

Inter-Annotator Agreement (IAA)

In order to assess how well an annotation task is defined, Inter-Annotator Agreement (IAA) scores can be used to see how individual annotators compare to each other. If an IAA score is high, that is an indication that the task is well-defined and other annotators will be able to continue the work.

- The clearer the guidelines – the better Inter-Annotator Agreement you are likely to achieve
- The higher the IAA – the better automatic results can be obtained (less noise!)

Having a high IAA score doesn't necessarily mean that the annotations are correct. It simply means that the annotators are all interpreting the instructions provided in the Annotation Guidelines consistently in the same way. The task may still need to be revised even if the IAA scores are high.

Evaluation Criteria for WP6 Use Cases

The evaluation criteria of Pharma Fake news detection use case will focus on ensuring the completeness of the so called "ground truth" knowledge graph compiled from trustful sources like FDA, CT.gov and other reputed and reliable organizations under strict governance. This will ensure the effectiveness of checking certain facts from suspicious news posts. The automation of the news monitoring process and facts validation are other important features of the use case that need to be evaluated too.

The evaluation of the Cultural Heritage and Archiving use case implementation will cover the completeness of implementation of the user stories described in this document. The workflows and end user tasks described in the user stories will be tested within an example environment with real data and the level of completeness as well as possible extensions be documented.

The evaluation of the LOD use case implementation will cover the efficiency as well as the effectiveness of the checking of single facts and complete knowledge graphs. The main goal

is to use knowledge graphs of the Linked Open Data cloud. However, since a ground truth has to be available for the most evaluation measures, we may rely on subsets of datasets.

Conclusions and future work

In this document, we presented the requirements for the three use cases of the industrial partners of the FROCKG project as well as the academic Fact Checking component that will drive the module development. Next, we introduced the alignment of our proposed requirements to our work packages as well as we matched the requirements to work packages

After month 6 of the FROCKG project, we will start with the implementation of the work package 2, 3, 4, and 5. In work package 6 we will use the components developed in the other work packages and use them to implement the use cases based on the user stories described in this document.