Universität Leipzig
Faculty of Mathematics and Computer Science
Department of Computer Science, Database Group

Connecting GOMMA with STROMA: An Approach for Semantic Ontology Mapping in the Biomedical Domain

Bachelor Thesis

Leipzig, November 2015

submitted by

Möller, Maximilian B.
Matr. No.: 1801278
Course: Informatik (BSc)

Referee: Prof. Dr. Erhard Rahm
Supervisor: MSc Patrick Arnold

Acknowledgment

First and foremost, I wish to express my deep gratitude to my supervisor Patrick Arnold for his encouraging support and feedback. Initially, he helped me to conceptualize the topic of this work. He had always a sympathetic ear for my questions such that my understanding of semantic enrichment and the STROMA system benefited from his advice. Moreover, his proposed corrections concerning the formal aspects of this thesis give me helpful impetus.

Furthermore, I would like to sincerely thank Dr. Anika Groß for her revealing introduction to GOMMA and insightful responses to biomedical ontologies. Without her, the work would have been more stony.

Gratitude is owed to Prof. Dr. Erhard Rahm, who gave me the opportunity to work and write this thesis in the Database Group.

And last but not least, cordial thanks are due to all who motivated me in times of sluggishness, gave me valuable hints for my thesis, or just brightened up the day. Thanks for your care.

List of Abbreviations

API Application Programming Interface

AUI Atom Unique Identifier

CUI Concept Unique Identifier

ER model Entity-Relationship model

FMA Foundational Model of Anatomy

GOMET Gomma Mapping Enrichment Tool

GOMMA Generic Ontology Matching and Mapping Management

JDBC Java Database Connectivity

LUI Lexical Unique Identifier

MA Adult Mouse Anatomy Ontology

MeSH Medical Subject Headings

NCIt National Cancer Institute Thesaurus

OAEI Ontology Alignment Evaluation Initiative

OWL Web Ontology Language

RDF Resource Description Framework

RDFS Resource Description Framework Schema

SemRep Semantic Repository

SNOMED CT Systematized Nomenclature of Human and Veterinary Medicine Clinical Terms

SPARQL SPARQL Protocol and RDF Query Language

STROMA Semantic Refinement of Ontology Mappings

SUI String Unique Identifier

UML Unified Modeling Language

UMLS Unified Medical Language System

Contents

1 Introduction **1**

2 Ontologies within Information Technology **4**

2.1 Basics of Ontologies 4
2.2 Representation of Ontologies 9
2.3 Application of Ontologies and their Benefits 12
2.4 Relating Ontologies: Matching 15
2.4.1 The Idea behind Matching 15
2.4.2 Techniques for Matching 18

3 Mapping and Enrichment Tools **20**

3.1 Short Introduction to GOMMA 20
3.1.1 Basic Architecture 20
3.1.2 The GOMMA Format of Mappings 21
3.2 Short Introduction to STROMA 22
3.2.1 Basic Architecture 22
3.2.2 Strategies for Semantic Type Detection 23
3.2.3 Post-processing: Control Type Computation 26

4 Semantic Enrichment of GOMMA Mappings **27**

4.1 UMLS as Background Knowledge Source 27
4.1.1 Overview about UMLS 27
4.1.2 Relation Extraction from UMLS 30
4.1.3 Interim Evaluation of Integrating UMLS to SemRep 32
4.2 GOMET: Extending GOMMA Mappings 35

5 Evaluation and Discussion **38**

5.1 Parametrizing the Evaluation 38
5.1.1 Independent and Dependent Variables 38
5.1.2 Evaluation Measures 40
5.2 Testing GOMET 41
5.2.1 A Small but Rich Mapping: MA-to-Wikipedia 41
5.2.2 A Real World Case: MA-to-NCIt 47
5.2.3 Further Mappings 50
5.3 Focussing Benefits and Problems 51

6 Conclusion **54**

Bibliography **55**

1 Introduction

Walking through the world, a human being is surrounded by a lot of objects, and their sensations continuously assail him. Nevertheless, the sensations are seldom disordered. Moreover, they get structured and are abstracted and classified. Thus, the human being conceptualizes its surrounding and builds a mental model of the world. In the digital society, a huge amount of knowledge is shared and automatically computed – often in real time. This process is sped up if the underlying models of the knowledge are shared and machine-readable, because they help to understand the shared knowledge. A data structure which describes the model of a specific domain of interest is called an ontology. As they model the meaning of things, ontologies represent a semantic technology.
If more and more objects (often called *entities*) are observed, these entities can be grouped together to different concepts with a set of attributes, which define the characteristics of each concept. Additionally, concepts can be related to each other. For instance, paleontologists have been discovered many fossils in the last centuries. The more fossils – which represent the entities of the paleontological domain – are excavated, the easier it is to group the fossils and define families, genera, and species. The fact that a defined family is a subfamily of another one exemplarily represents a relation between concepts of the given domain. Nevertheless, the experts might be disagree with the concept definitions or the relations between concepts. This is the case for paleontologists. At least two different ontologies are established within the discipline: [Ben06], a Linnaean taxonomy, as well as [WDO04], a cladistic systematics. Consequently, an ontology models the knowledge of a specific domain under a particular perspective. Ideally, this model is consistent and up to date.
In addition to ontology application in scientific context, ontologies are used in enterprise applications as well. For instance, the oil and gas industry uses an ontology to model their vocabulary and thus their business domain. The ontology mediates the information exchange between offshore platforms (field data assessment) and onshore participants (operators and vendors), see [Obe14, KSVS08]. Thus, the ontology serves as a conceptualization of a world's segment.
The previous assumptions implicitly set forth the main characteristics and the importance of ontologies. In short, an ontology can be defined as a "formal, explicit specification of a shared conceptualisation" [SBF98]. As such, it is machine-readable and can be used for varying communication and translation tasks within a given domain. For example, communication between software developers is facilitated when it is based on a shared ontology. Furthermore, translation tasks, or more formally the mapping of one ontology to another, is a key requirement in data integration. A mapping is useful in order to integrate an ontology to a different one, for example. This might be the case when one company buys into another one and their product catalogues shall be merged.
A mapping consists of a source and target ontology as well as of a set of correspondences. A correspondence associates a concept of the source ontology to a concept of the target one. It is commonly assumed that the association is meaningful, i.e., both concepts denote the same or very similar things. Determining correspondences is a complex task. On one side, an ontology may contain hundreds of thousands concepts and consequently, an automatic processing is recommended. On the other side, an ontology describes such a very specific domain that this might be mastered

only by experts. But the following consideration illustrates that a pure manual matching is impractical. Let 3,000 concepts be in the source and in the target ontology, respectively. Assuming an n-to-m mapping, $n \cdot m$ comparisons has to be considered in order to compare each concept of the source ontology with each concept of the target ontology. Thus, 9 million comparisons are necessary. Assume that an expert needs 5 seconds in order to determine whether two given concepts are equal, one concept is a more general term for the other one, or both concepts have nothing in common. Then the expert's evaluation would take about 1.5 years of uninterrupted work. This implicates that a mapping task cannot be done manually. Rather, it has to be made automatic or at least semi-automatic with seldom user interaction. However, strategies have to be established which take account of the very domain specific language.
There are a lot of techniques for automatically determining the correspondence set. For example, the similarity between two concepts can be calculated based on the similarity of their names or with the help of an intermediary ontology to which target and source ontology are anchored. Furthermore, correspondences are enriched by semantic relation types. A semantic relation type is the kind of relation which holds between the concepts of a correspondence. Those strategies may use linguistic knowledge, e.g., information about the structure of words (*left eye* and *right eye* are both specialisations of *eye*), or they use a background knowledge resource for denoting the type (*hypothalamus* is a part of the *brain*). It depends on the framework which relation types are denoted. Some differ only between three types (*equal*, *less general*, *more general*) whereby some frameworks differentiate more fine-grained between *is-a*, *has-a*, *inverse is-a*, and *part-of*. The first two are kinds of *less general*, the latter are specializations of *more general*. Additionally, the relation *related-to* is introduced for concepts which are only loosely connected.

This thesis focuses on ontology mappings within the biomedical domain. Mappings within this domain are useful, e.g., for a study in comparative anatomy where two anatomical ontologies are connected to each other. However, the denotation of semantic relation types (called *semantic enrichment*) between two concepts is often ignored. Thus, this thesis establishes and evaluates an enrichment approach for biomedical mappings. Two steps are necessary for this approach. First, suitable background knowledge has to be extracted and prepared in such a way that it can be integrated into a repository. That repository will be accessed during the enrichment process. Second, the integration of the systems GOMMA and STROMA has to be implemented. GOMMA is a mapping tool which determines a set of correspondences from two input ontologies. STROMA is a system for semantic enrichment of a given mapping. Thus, a GOMMA mapping is committed to STROMA and the types denoted by STROMA are eventually written to it. Finally, the enriched mappings are evaluated. As it will be seen, semantic enrichment within the biomedical domain is not equivalent to less specialised mappings, like a mapping between clothing categories. Well-established linguistic strategies within the latter domain fail if they are applied to biomedical mappings. Other linguistic strategies might be useful. Furthermore, background knowledge seems to play a more crucial role. The results are better for a higher weight of background knowledge regarding the sum of the other strategies' weight.
The thesis is structured as follows. In chapter 2 the theoretical background is set forth. It concentrates on the role of ontologies within computer science. The most formal part is section 2.1 where the conception of an ontology is defined as a logical theory. That procedure has two advantages. First, the definition precisely describes the basic idea of an ontology. Thus, the philosophical basics

lead to a more insightful understanding of ontologies. Second and as a consequence of the first, that definition may evoke a sense for problems and weak points of ontologies. The next section 2.2 introduces two common representations of ontologies, namely as directed graph and with the help of the knowledge representation language OWL. After shortly pointing out advantages of ontologies in business applications (section 2.3), in section 2.4 the matching task is explained and important techniques for matching are presented.
Chapter 3 gives on overview about the used systems for enriching a biomedical mapping. GOMMA (section 3.1) is a powerful infrastructure which executes a matching task. Besides that it is applicable for the analysis of ontology evolution. After generating a mapping with GOMMA, that mapping is input for STROMA, which is described in section 3.2. STROMA is an enrichment tool which denotes a mapping with semantic relation type information.
The additional implementations which are necessary for the semantic enrichment of GOMMA mappings are set forth in chapter 4. At first, section 4.1 deals with the integration of UMLS into the repository. An interim evaluation of the integration is part of this section. Subsequently, GOMET is introduced in section 4.2. GOMET is an implementation for connecting GOMMA with STROMA.
The evaluation of GOMET takes place in chapter 5. The independent and dependent variables as well as evaluations measures are defined in section 5.1. The next section 5.2 presents the experimental tests. Two mappings are discussed in more detail. Furthermore, problems with further mappings are outlined. Eventually, section 5.3 discusses the results and shows potential improvements of enrichment strategies for biomedical mappings. An overall conclusion is given in chapter 6.

2 Ontologies within Information Technology

This chapter sets forth important theoretical concepts as well as software tools based on these concepts which are essential part of the present thesis. The most fundamental theoretical concept is the notion of an ontology, which is introduced first. Subsequently, the concept of connecting two ontologies via detecting conceptual correspondences (which is called matching) is defined.
The information scientific notion of an ontology derives its origin from the same-named philosophical discipline of investigating the nature of being. This section provides a formal (information scientific) definition of 'ontology' in order to target two things: first, getting an insight into the philosophical dimension of ontological models, and second, understanding the important role of ontologies within information technology.[1] More precisely, in information science ontologies are used for describing a domain of interest, i.e., a specific part of the real world. Section 2.1 elucidates which formal assumptions underlie this possibility of description. The next section 2.2 outlines some representation possibilities. Section 2.3 sketches common application fields of ontologies and advantages of ontological models. Finally, ontologies are put into context of the topic of this thesis. Hence, in section 2.4 the notion of ontology matching is explained.

2.1 Basics of Ontologies

An ontology describes a specific domain of interest, or to put it another way: an ontology specifies a conceptualization of said domain which contains domain specific entities and their relations to each other. Such a conceptualization has two characteristics. First, it is formulated as an explicit specification, i.e., not a mental one in someone's mind, [Gru93, GOS09]. Second, the conceptualization is formally specified, i.e., the ontology specification has to be machine-readable, [Bor97, GOS09]. Furthermore, the conceptualization itself has to be a shared one [Bor97, UG96, GOS09], meaning that the users of the ontology agree on the (linguistic or symbolic) primitives the conceptualization is built on. That last point guarantees successful interoperability.
The following paragraphs give accurate and formal definitions of the previously mentioned aspects. Together they provide a definition of 'ontology', following [GOS09], but with a different running example taken from the biomedical domain. The basics of the current example are described as follows: Let M be an infinite set of mice $m_1, \dots m_n$. It is possible to know the anatomy of each mouse. The aim is the design of a mouse anatomy ontology.[2] The basic ontological entities are the anatomical parts of each mouse like its outer ear, blood plasma, or cardiac muscle tissue. Possible relationships between these entities are *part-of*, like *blood plasma is part of blood*, and *is-a*, like *blood is a body fluid or substance*.

[1] Note that also that within information technology the term 'ontology' is used in a variety of readings. An overview about this term is given in [GG95].

[2] Such an ontology has been developed by Hayamizu et al. as part of the Gene Expression Database Project and is accessible at http://www.informatics.jax.org/searches/AMA_form.shtml.

Conceptualization As defined in definition 2.2 according to [GOS09, 6-7], a conceptualization **C** consists of a set of domain entities (called *universe of discourse* D), e.g. blood plasma of mouse m_i, cardiac muscle tissue of mouse m_j (with $i, j \in \{1, \ldots, n\}$). For instance, be $n = 3$ and the only anatomical parts of a mouse are its eyes and legs, then (assuming that each mouse is a typical one) $|D| = 3 \cdot 2 + 3 \cdot 4 = 18$, i.e. the universe of discourse contains 18 elements since there are three mice with two eyes and four legs.
Furthermore **C** contains a set of possible worlds W. The notion of possible worlds is a well-established concept within formal semantics and a premise of intensional approaches (for an introduction see [FH11]). Besides the state of affairs in the world we live in, there exist other worlds which are more or less similar to our particular world. For example, in one world people say that lips are part of the mouth, but in another world lips are categorized as something different, hence they are not regarded as part of the mouth.
Finally, **C** has to specify which relation holds between the domain entities of D in a given world. In the example: Are lips categorized as part of the mouth? The mapping from a particular world to its state of affairs concerning a particular relation (called extensional) is done by a conceptual relation ρ as it is defined in definition 2.1 according to [GOS09, 6]. Note that an extensional relation models a specific world state regarding D, i.e., it states which "concrete" entities within D participate to the relation itself.

Definition 2.1. Let D be the universe of discourse and W the set of all possible worlds. Then a *conceptual relation* (also called *intensional relation*) ρ^n of arity n on $\langle D, W \rangle$ is a total function $\rho^n : W \to 2^{D^n}$ from the set W into the set of all n-ary (extensional) relations on D. [GOS09]

Definition 2.2. A *conceptualization* is a triple $\mathbf{C} = (D, W, \Re)$ with

- D a universe of discourse,
- W a set of possible worlds,
- $\Re$ a set of conceptual relations on the domain space $\langle D, W \rangle$. [GOS09]

It is important to note that *blood plasma, blood, outer ear, ...* are unary relations and that there is a difference between the entity (part of the extension of these words) and the word itself (whose extension depends on the chosen world). Consequently, depending on the world *blood*, or every other name for an anatomical part of a mouse, can mean something different from world to world. Hence, it would be helpful to identify each element of the universe of discourse, i.e. the anatomical parts of each mouse, with an identification number. Let ap_i ($i \in \mathbb{N}$) be such an identification number for each anatomical part. That yields the following (anatomical) conceptualization of mice:

- $D = \{\mathrm{ap}_1, \mathrm{ap}_2, \mathrm{ap}_3, \mathrm{ap}_4, \ldots, \mathrm{ap}_m\}$
- $W = \{w_1, w_2, \ldots\}$
- $\Re = \{\mathrm{anatomicalPart}^1, \mathrm{blood}^1, \mathrm{outerEar}^1, \mathrm{bloodPlasma}^1, \ldots, \text{part-of}^2, \text{is-a}^2\}$, whereby the conceptual relations can be defined in the following way:

- $\forall w \in W : \text{anatomicalPart}^1(w) = D$
- $\text{blood}^1(w_1) = \text{blood}^1(w_3) = \text{blood}^1(w_5) = \ldots = \{\text{ap}_1, \text{ap}_{11}, \text{ap}_{111}, \text{ap}_{1111}, \ldots\}$
  $\text{blood}^1(w_2) = \text{blood}^1(w_4) = \text{blood}^1(w_6) = \ldots = \{\text{ap}_2, \text{ap}_{22}, \text{ap}_{222}, \text{ap}_{2222}, \ldots\}$

...

- $\text{part-of}^2(w_1) = \{(\text{ap}_1, \text{ap}_{42}), (\text{ap}_1, \text{ap}_{105}), \ldots\}$
  $\vdots$
- $\text{is-a}^2(w_1) = \{(\text{ap}_1, \text{ap}_2), (\text{ap}_1, \text{ap}_{26}), \ldots\}$
  $\vdots$

This conceptualization only categorizes anatomical parts of mice. In world w_1, ap_1 is the blood of a specific mouse. In world w_2, ap_2 is the blood of a different mouse; but it is not explicitly mentioned above what is the property of ap_2 in w_2. It is merely depicted that is-a^2 holds between ap_1 and ap_2 in w_1.

Formal, Explicit Specification The paragraph above sets forth the conceptualization $\mathbf{C}$ of our domain of interest. But this conceptualization has to be made explicit in order to apply or communicate it. A language $\mathbf{L}$ commits to $\mathbf{C}$ if $\mathbf{L}$ provides a vocabulary $\mathbf{V}$ such that the elements of $\mathbf{V}$ represent a certain conceptual relation of $\Re$ in the intended way. In order to make sure that the words in $\mathbf{V}$ mean the right thing, $\mathbf{C}$ can be intensionally specified. Hence, meaning postulates (axioms) constrain $\mathbf{L}$ in a suitable way. For example, is-a^2 is a transitive relation whereby part-of^2 is not transitive in general.[3] Both relations are asymmetric, but only is-a^2 is irreflexive since a human (an unborn child), for example, is part of a human (its mother). Consequently, it is possible to say which models fit better to the intended conceptualization than others, i.e. the result is "an *approximate* specification of a conceptualization" [GOS09, 8]. Moreover, because the specification has to be formal, natural language is excluded and hence a logical language is chosen for $\mathbf{L}$.
These aspects are formalized in definition 2.3 as the definition of an ontological commitment, after [GOS09, 10]. In the running example, the ontological commitment makes sure that the language symbol *blood* is mapped to the conceptual relation blood^1, *part-of* to part-of^2, and so on. Figure 2.1 illustrates the set of facts. Real world phenomena are perceived and conceptualized. A good ontology captures the intended models whereby a bad ontology is not able to represent the given conceptualization. Thus, an ontology builds on an extract of the real world, which is conceptualized mentally and has to be represented by a formal language in order to enable communication about and application of this perceived real world domain of interest.

Definition 2.3. Let $\mathbf{L}$ be a first-order logical language with vocabulary $\mathbf{V}$ and $\mathbf{C} = (D, W, \Re)$ a conceptualization. An *ontological commitment* (also called *intensional first order structure*) for $\mathbf{L}$ is a tuple $\mathbf{K} = (\mathbf{C}, \mathcal{I})$, where $\mathcal{I}$ (called *intensional interpretation function*) is a total function $\mathcal{I} : \mathbf{V} \to D \cup \Re$ that maps each vocabulary symbol of $\mathbf{V}$ to either an element of D or an intensional relation belonging to the set $\Re$. [GOS09]

[3] Transitivity of parthood is frequently discussed in literature, see [Gui09] and cited literature there. Two counterexamples against transitivity are depicted in the following:

- (Berlin, Germany), (Germany, United Nations) $\in$ part-of^2, but: Berlin is no part of the United Nations.
- (heart, musician), (musician, orchestra) $\in$ part-of^2, but: the heart is not a part of the orchestra.

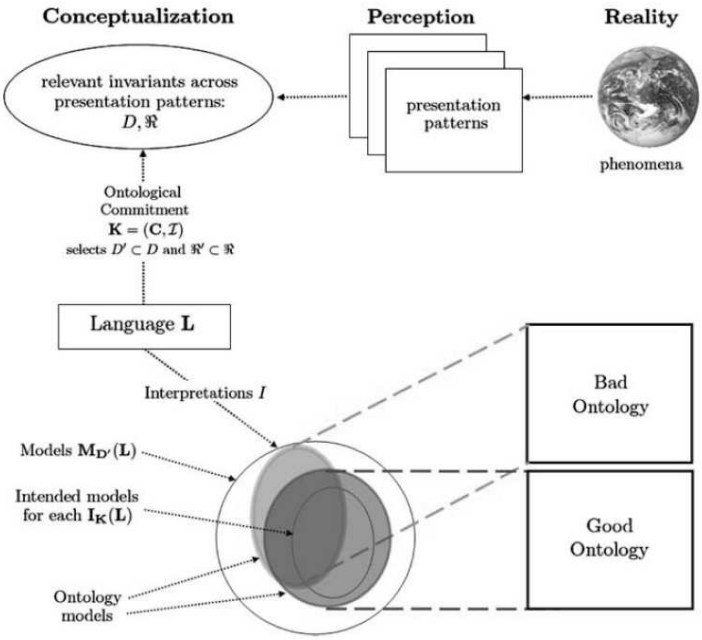

Figure 2.1: Overview about the connection of phenomena occurring in the real world, their conceptualization and explicit representation as an ontology. [GOS09, 9]

Ontology The concept of an intended model is introduced in definition 2.4 according to [GOS09, 10-11]. It connects the intensional with the extensional conception of meaning. An intended model is a model of the world which is compatible with the ontological commitment and thus, with the conceptualization provided by the experts. Finally, an ontology can be defined as a logical theory, see definition 2.5 (after [GOS09, 11]).

Definition 2.4. Let $\mathbf{R}$ be a set of relations on D, $\mathbf{C} = (D, W, \Re)$ a conceptualization, $\mathbf{L}$ a first-order logical language with vocabulary $\mathbf{V}$ and ontological commitment $\mathbf{K} = (C, \mathcal{I})$. A model $M = (S, I)$, with $S = (D, \mathbf{R})$ and $I : \mathbf{V} \to D \cup \mathbf{R}$, is called an *intended model* of $\mathbf{L}$ according to $\mathbf{K}$ iff

1. $\forall$ constant symbol $c \in \mathbf{V} : I(c) = \mathcal{I}(c)$,
2. $\exists w \in W \; \forall$ predicate symbol $v \in \mathbf{V} \; \exists \rho \in \Re : \mathcal{I}(v) = \rho \; \wedge \; I(v) = \rho(w)$

The set $\mathbf{I_K(L)}$ of all models of $\mathbf{L}$ that are compatible with $\mathbf{K}$ is called the set of *intended models* of $\mathbf{L}$ according to $\mathbf{K}$. [GOS09]

Definition 2.5. Let $\mathbf{C}$ be a conceptualization, and $\mathbf{L}$ a logical language with vocabulary $\mathbf{V}$ and ontological commitment $\mathbf{K}$. An *ontology* $\mathbf{O_K}$ for $\mathbf{C}$ with vocabulary $\mathbf{V}$ and ontological commitment $\mathbf{K}$ is a logical theory consisting of a set of formulas of $\mathbf{L}$, designed so that the set of its models approximates as well as possible the set of intended models of $\mathbf{L}$ according to $\mathbf{K}$. [GOS09]

For the running example, an ontology O_0 is created which consists of a set of formulae. is-a^2 constitutes a partial order[4] whereby part-of^2 is a strict (partial) order[5] – if transitivity and irreflexivity within the biomedical domain is assumed, see formulae **o3**.ii, **o4**.ii.[6] It seems to be doubtful whether formulae like **o1**.iv should be specified as taxonomic information since a) it is result of the research process and too specific and concrete to be a meaning postulate, and b) this information is encoded in the is-a^2 relation.

o1 Taxonomic Information:

i $\text{blood}(x) \rightarrow \text{anatomicalPart}(x)$

ii $\text{outerEar}(x) \rightarrow \text{anatomicalPart}(x)$

iii $\text{bodyFluid}(x) \rightarrow \text{anatomicalPart}(x)$

iv $\text{blood}(x) \rightarrow \text{bodyFluid}(x)$

$\vdots$

o2 Domains and Ranges:

i $\text{isa}(x, y) \rightarrow \text{anatomicalPart}(x) \wedge \text{anatomicalPart}(y)$

ii $\text{partOf}(x, y) \rightarrow \text{anatomicalPart}(x) \wedge \text{anatomicalPart}(y)$

o4 Antisymmetry and Reflexivity:

i $\text{isa}(x, y) \wedge \text{isa}(y, x) \rightarrow x = y$

ii $\text{isa}(x, x)$

o3 Asymmetry and Irreflexivity:

i $\text{partOf}(x, y) \rightarrow \neg\text{partOf}(y, x)$

ii $\neg\text{partOf}(x, x)$

o4 Transitivity:

i $\text{isa}(x, y) \wedge \text{isa}(y, z) \rightarrow \text{isa}(x, z)$

ii $\text{partOf}(x, y) \wedge \text{partOf}(y, z) \rightarrow \text{partOf}(x, z)$

o5 Disjointness:

i $\text{isa}(x, y) \rightarrow \neg(\text{partOf}(x, y) \vee \text{partOf}(y, x))$

[4] A (non-strict) partial order is a binary relation which is reflexive, antisymmetric, and transitive. Reflexivity holds because by saying *Some* x *is a* x. it is meant that some entity with property X is an element of the set of all entities which have property X. Obviously, this is true. In the case of antisymmetricity, it is assumed that all entities $x \in X$ have property Y (i.e., are part of Y), and all entities $y \in Y$ have property X (i.e., are part of X). But this can only be possible if X and Y are the same property. Similar arguments hold for transitivity. But is-a^2 is not total since, for example, there is no order between the properties *bone* and *blood* (totality would require that *bone is blood* or *blood is a bone* holds).

[5] A strict partial order is an irreflexive, transitive, and asymmetric binary relation. It might be questionable to what extent it can be assumed that part-of^2 is irreflexive (counterexample: *A matryoshka doll is part of a matryoshka doll.*, *A human is part of human, namely when a woman is pregnant.*) and transitive. But it is obvious that part-of^2 is not symmetric.

[6] Examples for the relation characteristics:

	(ir-)reflexive	anti-/asymmetric	transitive
is-a^2	An ear is an ear.	If the ear bone is an auditory bone, and the auditory bone is an ear bone, then ear bone and auditory bone are the same.	A mouse is a mammal, and a mammal is a vertebrate. Thus, a mouse is a vertebrate.
part-of^2	An ear is no part of the ear.	An ear is part of the face, but the face is no part of the ear.	An ear is part of the face, and the face is part of the head. Thus, an ear is part of the head.

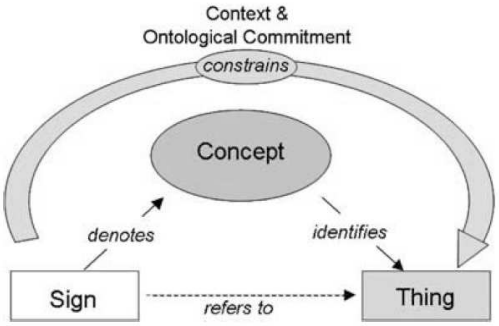

Figure 2.2: Semiotic triangle as it is revised in [GOS09, 16]. Context and ontological commitment disambiguate the sign.

Shared Conceptualization Since conceptualizations are mental representations of the world, it is necessary to communicate by examples what the primitives of the chosen language **L** mean. For example, how it can be guaranteed that two different humans share the same understanding of the *part-of* relation whilst taking into account ontology O_0 with its specification in **o1**-**o5**? Such humans can explain to each other how a world state must be designed such that the relation holds by means of an example of the actual world.[7] Hence it is necessary that the primitives are well-founded, i.e., the users of an ontology agree on their meaning. As a result, good approximations of conceptualizations can be shared and enable large-scale interoperability. These aspects are illustrated within the semiotic triangle [OR89], in figure 2.2 modified by [GOS09]. A sign is an element of **V**, a thing is a concrete object of the current world, and a concept is a part of the conceptualization which is invoked by the sign. Ideally, the same concept is evoked in the listener's mind as the speaker has intended to use it. Context and an ontological commitment make sure that this is guaranteed.

Summary The previous paragraphs define an ontology as a shared conceptualization which is explicitly and formally specified. The definition itself is formally stated. The aim of this procedure is to give an philosophical overview and thus a deeper understanding of ontologies in order to recognise the complexity and potentially problematic aspects of ontologies. For example, there are a lot of mapping processes – the reality is mapped to our perception, our perception is mapped to a cognitive conceptualization, a conceptualization has to be made explicit, etc. – and it has to be made sure that no important information is lost or mistakenly modified. One main problem is that eventually predicate names are chosen by resorting to natural language and thus ambiguity and vagueness become part of an ontology.

2.2 Representation of Ontologies

Besides the definition of an ontology as logical theory which might be a less practical way, an ontology can be described by means of a directed (acyclic) graph $O = \langle C, R, A \rangle$ [Gro14]. It consists of a set of concepts C as vertices and a set of relations R (the directed edges) among these concepts.

[7] Maybe part-of[2] is a more or less "well-founded" relation. But assume in another ontology there is the relation cooperatesWith[2]: in one reading, this relation holds when two humans share the same goal; in another reading, these two humans have also to act to achieve this goal [GOS09].

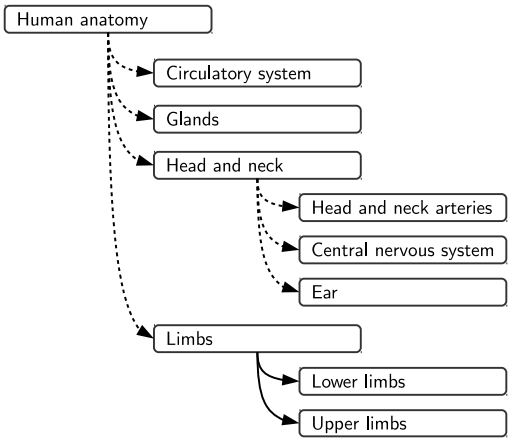

Figure 2.3: Graphical Representation of an Ontology. The concepts are Wikipedia categories. Dashed edges represent *part-of* and normal edges *is-a* relations.

Furthermore, a set A contains attributes which specify the given concepts, e.g. the ID (or accession number), the preferred label of the concept, synonyms, or a definition. A relation $r \in R$ between two concepts c_1, c_2 is a triple $\langle c_1, t, c_2 \rangle$ meaning that a relation of type $t \in T$ holds between c_1 and c_2. For example, blood plasma is part of the blood, thus $\langle$*blood plasma*, *part-of*, *blood*$\rangle$. An ontology graph is acyclic iff it contains neither i) symmetric relations, ii) reflexive relations, nor iii) inverse relations of already given ones. Since it is assumed that T consists of *part-of* and *is-a*, their inverse counterparts are *has-a* and *inverse-isa*. A symmetric relation is e.g. *related-to* or *is-synonym-of*. Although *is-a* is reflexive, in the ontologies used in this thesis, the edge from a concept to itself is not drawn as it represents only trivial knowledge. Hence, only directed graphs are considered which are acyclic. An example is given in figure 2.3. The root concept is *Human anatomy*. The relations $\langle$*Limbs*, *part-of*, *Human anatomy*$\rangle$ and $\langle$*Lower limbs*, *is-a*, *Limbs*$\rangle$ are elements of R.
Furthermore, an ontology (especially within the context of the semantic web) can be described by using an ontology markup language whose syntax is XML-based. Some examples are RDF (Resource Description Framework)[8], RDFS (Resource Description Framework Schema)[9], an extension of RDF, and OWL (Web Ontology Language)[10], which is based on the previous ones (for an introduction to these markup languages compare [GPFLC04, 199ff.]). An extract of an OWL ontology, the NCIt (National Cancer Institute Thesaurus)[11], is depicted in listing 2.1. After declaring the XML version and the document type (line 1-3), the RDF root node begins which contains the specification of namespaces (*xmlns*). General information of the ontology are given within *owl:Ontology* (line 9-12): the ontology documentation (*rds:comment*), and the ontology version (*owl:versionInfo*). Binary relations are described by *owl:ObjectProperty* (line 15-24) which contains a label and domain and range of the relation. The latter are concepts (called classes in OWL) and are marked with *owl:Class* (line 25-52). If there is more than one *rdfs:label* within a class, these labels represents different synonyms. The hierarchic structure of the concepts (*is-a*

[8] http://www.w3.org/RDF/
[9] http://www.w3.org/TR/rdf-schema/
[10] http://www.w3.org/TR/owl2-overview/
[11] The cited version can be downloaded from OAEI (Ontology Alignment Evaluation Initiative) 2013, http://oaei.ontologymatching.org/2013/ within the large biomedical ontology track. The current NCIt is online accessible via http://cbiit.nci.nih.gov/evs-download/thesaurus-downloads.

relations) are determined by *rdfs:subClassOf* which means that the current concept is a subclass of the given *rdf:resource*. As can be seen in line 44ff, there are concepts with no parent concept; hence, NCIt has different root nodes (*anatomy kind*, *biological process kind*, ...) which are disjoint to each other.

```
1  <?xml version="1.0"?>
2  <!DOCTYPE rdf:RDF [ [...]
3  ]>
4  <rdf:RDF xmlns="http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#"
5       xml:base="http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl"
6       xmlns:Thesaurus="http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#"
7       [...]>
8
9    <owl:Ontology rdf:about="http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl">
10     <rdfs:comment>NCI Thesaurus, [...]</rdfs:comment>
11     <owl:versionInfo>08.05d</owl:versionInfo>
12   </owl:Ontology>
13
14   [...]
15   [<!-- Object Properties -->]
16
17   <!-- http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#
       Anatomic_Structure_Is_Physical_Part_Of -->
18     <owl:ObjectProperty
       rdf:about="&Thesaurus;Anatomic_Structure_Is_Physical_Part_Of">
19         <rdfs:label>Anatomic_Structure_Is_Physical_Part_Of</rdfs:label>
20         <rdfs:domain rdf:resource="&Thesaurus;Anatomy_Kind"/>
21         <rdfs:range rdf:resource="&Thesaurus;Anatomy_Kind"/>
22     </owl:ObjectProperty>
23
24   [...]
25   [<!-- Classes -->]
26
27   [...]
28   <!-- http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#A-007 -->
29     <owl:Class rdf:about="&Thesaurus;A-007">
30         <rdfs:label xml:lang="en">A-007</rdfs:label>
31         <rdfs:label xml:lang="en">Aryl Hydrazone A-007 Gel</rdfs:label>
32         <rdfs:subClassOf rdf:resource="&Thesaurus;Immunostimulant"/>
33     </owl:Class>
34
35   [...]
36   <!-- http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#Immunostimulant -->
37     <owl:Class rdf:about="&Thesaurus;Immunostimulant">
38         <rdfs:label xml:lang="en">Immunostimulant</rdfs:label>
39         <rdfs:subClassOf rdf:resource="&Thesaurus;Biological_Response_Modifier"/>
40     </owl:Class>
41
42   [...]
43   <!-- http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#Anatomy_Kind -->
44     <owl:Class rdf:about="&Thesaurus;Anatomy_Kind">
45         <rdfs:label xml:lang="en">Anatomy_Kind</rdfs:label>
46         <owl:disjointWith rdf:resource="&Thesaurus;Biological_Process_Kind"/>
47         <owl:disjointWith rdf:resource="&Thesaurus;Chemicals_and_Drugs_Kind"/>
48         <owl:disjointWith rdf:resource="&Thesaurus;Chemotherapy_Regimen_Kind"/>
49         <owl:disjointWith
```

```
    rdf:resource="&Thesaurus;Diagnostic_and_Prognostic_Factors_Kind"/>
50          <owl:disjointWith rdf:resource="&Thesaurus;EO_Anatomy_Kind"/>
51          [...]
52      </owl:Class>
53  </rdf:RDF>
```

Listing 2.1: Describing an ontology with OWL: extract from the NCIt.

2.3 Application of Ontologies and their Benefits

In the previous subsection, it has been stated that an ontology is an explicit and formal specification of a shared conceptualization [SBF98]. Although this represents the philosophical core of each ontology design pattern, in current scientific research ontologies are applied for varying tasks (natural language processing, knowledge management, e-commerce, the Semantic Web, etc.) and in different communities (knowledge engineering, databases and software engineering) [GPFLC04]. Consequently, there are various types and categorizations of ontologies. For example, one can distinguish between *lightweight* and *heavyweight* ontologies [GPFLC04, 8]. The first one is more or a less a taxonomy, i.e., it describes concepts and their relationship to each other. The latter adds axioms and constraints to such taxonomies in order to yield a more fine grained definition of the used terms.
Ontologies may be characterized by two dimensions: i) the richness of their internal structure, and ii) the subject of their conceptualization [GPFLC04]. The first dimension is frequently discussed within literature [GOS09, LM01, UG04]. Ontologies are characterized along a continuum as it is depicted in figure 2.4. From left to right the ontology type becomes more formal and can express more meaning aspects. For example, a thesaurus is a vocabulary with additional semantic information between the vocabulary terms (e.g., synonym relations) but lack an explicit hierarchy of the terms. A formal taxonomy is a strict subclass hierarchy, which enables inheritance.

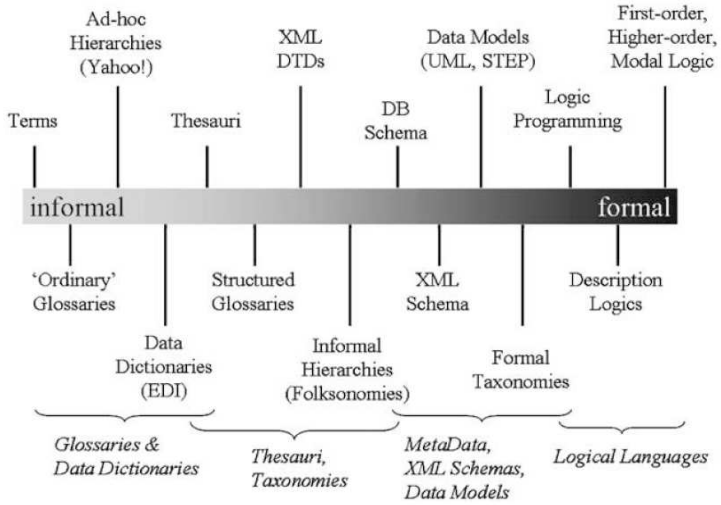

Figure 2.4: Dimension 1 for categorizing ontologies. From left to right the amount of specified meaning as well as the degree of formality increases. [GOS09, p.13]

The second dimension for categorizing ontologies is the subject of their conceptualization. The following types are taken from [GPFLC04, 29-34]:

Knowledge representation ontology "captures the representation primitives used to formalize knowledge under a given [...] [knowledge representation] paradigm".

General/ Common ontology "represent[s] the common sense knowledge reusable across domains".

Top-level/ Upper-level ontology "describe[s] very general concepts and provide[s] general notions under which all root terms in existing ontologies should be linked".

Domain ontology "provide[s] vocabularies about concepts within a domain and their relationship" and is "reusable in [...] [the] given specific domain (medical, pharmaceutical, engineering, law [...])".

Task ontology "describe[s] the vocabulary related to a generic task or activity (like diagnosing, scheduling, selling, etc.)".

Domain-task ontology is "reusable in a given domain, but not across domains. [...] [It is] application-independent."

Method ontology "give[s] definitions of the relevant concepts and relations applied to specify a reasoning process so as to achieve a particular task".

Application ontology "contain[s] all the definitions needed to model the knowledge required for a particular application".

The ontologies used within this thesis are lightweight ontologies with a more formal but not logically defined internal structure. They can be located near 'formal is-a hierarchy' at dimension 1. In dimension 2, they represent domain ontologies from the biomedical domain. Ontologies within the medical domain are of concern since they enable sharing, transmitting, as well as annotating patient or experiment data. For these purposes it is necessary that the communicated concepts are unambiguous. Common ontologies within this area are UMLS (Unified Medical Language System)[12], NCIt[13], SNOMED CT (Systematized Nomenclature of Human and Veterinary Medicine Clinical Terms)[14], MA (Adult Mouse Anatomy Ontology)[15], and FMA (Foundational Model of Anatomy)[16].
Each knowledge and technique is confronted at some time with the question whether and how it can be successfully integrated into the methods of current companies and task forces. The above characterized types of ontologies are not only applied for scientifc research but are also of considerable use for enterprise applications. [UG04] introduces four main use cases for ontology application:

Neutral authoring The basic idea is that a company uses a huge amount of non-interoperable tools and software. Hence, the company desings a neutral ontology for their own use and then, the terminoly of each target system can be computed/ translated from the company ontology.

[12] http://www.nlm.nih.gov/research/umls/
[13] https://ncit.nci.nih.gov/ncitbrowser/
[14] http://www.ihtsdo.org/snomed-ct
[15] http://www.informatics.jax.org/searches/AMA_form.shtml
[16] http://sig.biostr.washington.edu/projects/fm/

Ontology-based specification Ontologies are used in the process of software engineering as a knowledge representation (specification) and within software development – knwon as ontology-driven software engineering [HS06, WC12].

Common access to information The basic situation is similar to neutral authoring, i.e., there are different (legacy) software systems with varying terminology. An ontology O_c is designed which is used as a connector between source ontology O_s and target ontology O_t. Hence, the source format is translated from O_s to O_c and then from O_c to O_t.

Ontology-based search An ontology can describe the categories within an information repository and thus represents an indexing mechanism.

Another study of ontologies in enterprise applications is situated in the SAP Research [Obe14]. Eight characterizing technological features are attributed to an ontology, which determine its important role for enterprise application. Some of these features and application scenarios have been already discussed above, some are new aspects which provide further arguments why ontological modeling is a valuable semantic technology for solving business problems. [Obe14, 475ff] identifies the following features characterizing an ontology:

Conceptual Modeling Similar to an ER model (Entity-Relationship model) or an UML (Unified Modeling Language) diagram, an ontology models a domain of interest in an intuitive way that facilitates communication between different agents. The model contains classes/ concepts, properties/ relations, instances/ objects, rules and axioms.

Flexibility An ontology is flexible regarding the conceptual model, i.e., "classes, properties, rules & axioms as well as instances [...] can be managed at run time of an application" [Obe14, 475]. Main tools for this purpose are a suitable API (Application Programming Interface), automated evolution strategies, and dynamic direct programming.

Direct Interaction User-friendly interfaces can be graphical (tree-based), wiki-like, or based on (controlled) natural language.

Reuse A shared conceptualization simplifies that an ontology is applied to multiple applications of the modeled domain. There are three levels of reusing: individual (only one person benefits), community (a particular group or company frequently uses the ontology), world (an online ontology can be accessed by anybody).

Best Practices In order to simplify the ontology design process and build different ontologies on the same basis, abstract aspects of ontology are predefined within the best practices i) foundational ontologies, ii) ontology design patterns, iii) quality criteria.

Web Compliance Web compliance is achieved by standards due to recommendations of the W3C (World Wide Web Consortium)[17]. Thus, publications of ontologies are formulated in RDF and OWL; querying is enabled by SPARQL (SPARQL Protocol and RDF Query Language)[18]; annotations of web resources can be done with ontologies.

Formality [Obe14] accepts only logical language for describing ontology and distinguishes between logic programming languages (like F-Logic), description logics (like OWL-DL), and first-order logics.

[17] http://www.w3.org/0
[18] http://www.w3.org/TR/sparql11-overview/

Reasoning An ontology with a formal internal structure enables reasoning such as subsumption checking (identifies super- and subclass relations), consistency checking, instance classification, and instance retrieval (queries for an instance).

These features are profitably involved in the following enterprise applications [Obe14, 478ff]:

Creating New Business Scenarios [Obe14, 479] defines a business scenario as a description of "future business circumstances based on past and present trends, uncertainties, and assumptions." Hence, web compliance and reuse together with the conceptual model of a domain enables that new markets are accessed.

Increased Productivity of Information Workers This aspect is achieved due to more efficient access (visualization, interaction) to required data.

Improved Enterprise Information Management Ontologies are used to manage information within an organization, e.g., making data for decision making available. Hence, knowledge out of varying resources has to be combined and should be flexibly accessed. This is similar to the above mentioned application scenario "Common Access to Information".

Increased Productivity of Software Engineering Main aspects within this ambition are quality improvement, cost and time reduction, and the development of semantic web services.

The previous lines show that ontologies are a powerful tool for modeling knowledge of a domain of interest. Benefits are mainly the improvement of communication and interoperationality on several layers (human user vs. domain data, knowledge from different sources but out of the same domain etc.). Nevertheless, creating an ontology, integrating the new system and introducing it to the employees can be very expensive and it has to be well investigated whether costs and benefits define a good ratio.

2.4 Relating Ontologies: Matching

This subsection gives an introduction to an use case which is important for dealing with ontologies, namely aligning two ontologies as it is required, for instance, in a lot of the above describe scenarios. The alignment takes place between the concepts of an ontology. Furthermore, the instances of a concept are called entities (or individuals) and are not considered in the following.

2.4.1 The Idea behind Matching

Given two ontologies O_1 and O_2 it might be asked whether theses ontologies conceptualize the same objects, i.e., whether there is a correspondence from one concept c_1 of O_1 to a similar concept c_2 of O_2. For example, two ontologies which both conceptualize anatomical entities will very likely share a set of concepts and namely entities which are part of both ontologies like *blood* or *lower extremity*. Finding a set of correspondences is achieved by identifying identical concepts or subsumption relations (one ontology may only contain *extremity* and thus *lower extremity* is subsumed under *extremity*). In practical this is an important issue as ontologies share a huge amount of heterogeneity on the syntactic level (different ontology languages), terminological level (different names for the same entity), conceptual level (different conceptualization approaches of the same domain), semiotic level (different interpretations of the same entity by different people)

[ES07, Stu11], whereby the crucial heterogeneity for this section is of semantic nature, i.e., on the terminological and conceptual level. There are a lot of ontologies which conceptualize the same domain of interest but with different concept labels. A common example, see [SE13], says that an e-commerce company acquires another one. Hence, their ontologies describing their product data has to combined in order to yield an integrated ontology.
This (semi-automatic) process of determining similar, "matching" concepts between two ontologies is called *matching* and can be defined according to [ES07, SE13] as follows (ignoring further parameters like an input alignment):

Definition 2.6. The matching process can be seen as a function $f(O_1, O_2)$ which takes two ontologies O_1, O_2 as input parameters and returns an alignment A between these ontologies.

In turn an alignment (also called mapping, especially to point out that the alignment is directed, see [ES07, 42f]) is a set of correspondences between entities of O_1 and O_2. The interesting part is the definition of a correspondence in definition 2.7 after [ES07, SE13, AR13, Gro14]. Let O be an ontology, note that $c \in O$ means that c is a concept within O.

Definition 2.7. Given two ontologies O_1 (source ontology) and O_2 (target ontology), a set of alignment relations P, a confidence structure over Ξ, a set of methods M, and set of status types S. Then a *correspondence* a is a 7-tuple

$$\langle id, c_1, c_2, r, \xi, m, s \rangle$$

such that

- id is an identifier for a,
- source concept $c_1 \in O_1$, target concept $c_2 \in O_2$,
- $r \in P$,
- $\xi \in \Xi$ is the strength of a,
- m is the method of determining ξ,
- s is the status of a,
- relation r holds between c_1, c_2 with confidence ξ.

In the following, the confidence value ξ is a value from $[0, 1]$ where a value of 1 (0) indicates a true match (false match); and P contains exactly *equal, is-a, inverse is-a, has-a, part-of, related-to*, although in most accounts only '$\equiv$' (equivalence), '$\sqsubseteq$' (less general), and '$\sqsupseteq$' (more general) are considered as relations [SHB+09]. If the correspondence was created manually (m = manual), normally $\xi = 1$. S consists of *handled*, *to verify*, i.e., $s \in S$ specifies whether an automatically created correspondence has already been checked by an expert (handled), or not (to verify). An alignment where each correspondence has been verified is called *reference alignment* (or perfect mapping, benchmark, or gold standard). Depending on the context and what information is important to consider, a correspondence a is given only as 5-tuple $\langle id, c_1, c_2, r, \xi \rangle$, or triple $\langle c_1, r, c_2 \rangle$ (with changed sequence of elements) – or in some other form.
The strength ξ of a correspondence a is calculated by means of the similarity function σ between

the two concepts c_1, c_2 of a. The definition 2.8 captures the above described characteristics of ξ, slightly modified with respect to [ES07, 74]. ξ is considered as a value of the image of σ and thus, ranges from 0 (positiveness) up to 1 (maximality).

Definition 2.8. Given a set of concepts O, a *similarity* $\sigma : O \times O \to \mathbb{R}$ is a function from a pair of concepts to a real number expressing the similarity between two objects such that

$$\begin{aligned} \forall c, c' \in O \,:\, \sigma(c, c') \geq 0 && \textit{(positiveness)} \\ \forall c, c', \bar{c} \in O \,:\, \sigma(c, c') \leq \sigma(\bar{c}, \bar{c}) = 1 && \textit{(maximality)} \\ \forall c, c' \in O \,:\, \sigma(c, c') = \sigma(c', c) && \textit{(symmetry)} \end{aligned}$$

[ES07]

Nevertheless, it might also be possible to determine the dissimilarity δ as a measure of the difference between two concepts, and then define ξ as $1 - \delta$, given that δ is normalized, i.e., $0 \leq \delta \leq 1$. Dissimilarity as well as distance, as a stricter notion of dissimilarity, are defined in the following way (after [ES07, 73f]):

Definition 2.9. Given a set of concepts O, a *distance* $\delta : O \times O \to \mathbb{R}$ is a function from a pair of concepts to a real number such that

$$\begin{aligned} \forall c, c' \in O \,:\, \delta(c, c') \geq 0 && \textit{(positiveness)} \\ \forall c \in O \,:\, \delta(c, c) = 0 && \textit{(minimality)} \\ \forall c, c' \in O \,:\, \delta(c, c') = \delta(c', c) && \textit{(symmetry)} \\ \forall c, c' \in O \,:\, \delta(c, c') = 0 \text{ iff } c = c' && \textit{(definiteness)} \\ \forall c, c', \bar{c} \in O \,:\, \delta(c, c') + \delta(c', \bar{c}) \geq \delta(c, \bar{c}) && \textit{(triangular inequality)} \end{aligned}$$

If at least the function satisfies positiveness, minimality, and symmetry it is called *dissimilarity.* [ES07]

An alignment is total if each concept of the source ontology is mapped to at least one target concept, as defined in definition 2.10 after [ES07, 48f]. This is important in all cases where one ontology is translated into another. A total alignment makes sure that there is a translation for each concept in the source ontology.

Definition 2.10. Given two ontologies O_1 and O_2, an alignment A over O_1 and O_2 is called *total alignment* from O_1 to O_2 iff

$$\forall c_1 \in O_1 \exists c_2 \in O_2 \exists r \in P \,:\, \langle c_1, r, c_2 \rangle \in A$$

[ES07]

Furthermore, it is not necessarily the case that the alignment is a one-to-one alignment where a source concept cannot have more than one target concept, and vice versa. There are cases of "complex matches" [DH05] which means that the resulting alignment is one-to-many and many-to-one, respectively.[19] For instance, O_1 contains the concepts *upper extremity* and *lower*

[19] A many-to-many alignment is very untypical and occurs extremely seldom [RB01].

extremity whereas O_2 contains the concept *extremity* and no further specification of it. Hence, $\langle$*lower extremity*, *is-a*, *extremity*$\rangle$ and $\langle$*upper extremity*, *is-a*, *extremity*$\rangle$ are part of the (many-to-one) alignment. Normally, complex matches only occur for relations which are not *equal*. Assume that $\langle c_1, equal, c_2\rangle$ and $\langle c_1', equal, c_2\rangle$ are part of the alignment between O_1, O_2 such that $c_1, c_1' \in O_1, c_1 \neq c_1', c_2 \in O_2$. Due to commutativity and transitivity of *equal* it can be inferred that $\langle c_1, equal, c_1'\rangle$ which means that O_1 is redundant and contains the same concept twice.
An example for a matching graph is given later in this thesis in figure 5.5. It is a mapping between a flat ontology as the source and an extract of the Wikipedia category tree as the target.

2.4.2 Techniques for Matching

The following lines introduce important techniques and measurements in order to determine the similarity or dissimilarity of two concepts. These are essential part of varying matching processes. The classification of these techniques follows [ES07, 74-116]. In most cases, different strategies are applied during the matching in order to yield best results.

Name-based Techniques The basic assumption of name-based techniques is that the more similar two concept names or labels are, the more similar the underlying concepts are. This assumption is challenged by i) synonyms, that means two concepts are very similar or constitute the same concept but are named with very dissimilar labels, e.g. *breasts/ mammary glands*; ii) polysemy/ ambiguity, i.e., two concepts are named the same but denote very different things, like *mouse* (pointing device of computers vs. a small rodent). Acronyms for instance may challenge name-based techniques as they combine the synonym as well as the polysemy problem, e.g. *ICD* may be the acronym of *international statistical classification of diseases and related health problems* (synonyms) but it may also stand for *implantable cardioverter-defibrillator* (polysemy).
After normalization of the concept labels (e.g., converting each alphabetic character to lower case) different techniques are applied to a pair of such labels $\langle l_1, l_2\rangle$. In the simplest case one defines the similarity as 1 if l_1 and l_2 are identical, else as 0. This can be expanded by considering substrings, i.e., the longer a common substring of l_1, l_2 is, the more similar the concepts are. For instance, $l_1 = dna$ and $l_2 = rna$ share the substring *na* which is 67% of each label. Hence, l_1 and l_2 are assumed to denote similar concepts. This technique can be refined by n-gram similarity which compares the shared n-grams of l_1, l_2[20]
More complex measures are the edit distance, which counts the costs for the operations that are necessary to apply to label l_1 in order to obtain l_2, and cosine similarity, where l_1 and l_2 are represented as vectors (for example, over the containing morphems) and then the cosine between these vectors defines the similarity measure. Furthermore, path information may be added to a label l, i.e., each label of a concept which is a node on the path from the root to the current concept of l is concatenated to l. The idea is that the path contains information which might be helpful for determining the similarity of the current treated concepts.
In addition, linguistic methods may positively influence the matching task. Thus, tokenization, lemmatization, and stopword elimination clean up the labels. External resources as lexicons or thesauri deliver further information on which the matching will be grounded.

[20] An n-gram is a substring of a word w of length n. For instance, given w = *$$dna$$* where $ marks beginning and end of the word, w consists of the following 3-grams (trigram): *$$d*, *$dn*, *dna*, *na$*, *a$$*.

Structure-based Techniques Structure-based techniques cover the processing of the internal and the relational structure of a concept [ES07, 92ff]. Internal structure-relevant aspects are the set of properties which may be assigned to the concept, the range and values, respectively, cardinality and multiplicity as well as such characteristics like transitivity or symmetry of these properties. For instance, consider data types. Ontology O_1 terms a concept *title* whereby the same concept is labelled *name* in ontology O_2. Although, these two terms are not very similar from a name-based technique, they both share the same data type, namely 'string'. Hence, this can be a (potential) further hint that they denote the same concept. [ES07] point out that internal structure-based techniques (except name-based techniques) are mostly used to eliminate correspondences which are incorrect, or they are applied in combination with other techniques. Nevertheless, the advantage of these techniques is that they are easy to implement and efficient.
Relational structure-based techniques take into account other concepts to which the current considered concept is related. Common and well-studied relations are taxonomic (*is-a*) and mereological (*part-of*) ones.

Extensional Techniques If the ontology contains instances, i.e. concrete objects, extensional techniques can be applied. For example, if a set of instances is subsumed in ontology O_1 under concept c_1 and under concept c_2 in ontology O_2, then it can be assumed that c_1 and c_2 correspond to each other.

Semantic Techniques The last class of mapping techniqunes categorised by [ES07] are semantic ones. Matching with an external ontology, i.e. background knowledge, is subsumed under theses approaches. Thereby, the domain ontology O_1 as well as the target ontology O_2 are mapped ("anchored") to an external ontology O_{ex}. Then concepts of O_1, O_2 are related via their anchor in O_{ex}. For example, $brain_{O_1}$ is mapped to $brain_{O_{\text{ex}}}$, and $head_{O_2}$ is anchored to $head_{O_{\text{ex}}}$. Since $\langle brain_{O_{\text{ex}}}, \textit{part-of}, head_{O_{\text{ex}}} \rangle$ holds in O_{ex}, it is concluded that $\langle brain_{O_1}, \textit{part-of}, head_{O_2} \rangle$ holds.
Deductive techniques, for example, check whether a correspondence leads to an inconsistent alignment. This requires a set of axioms which already contains some (true) correspondences. If a correspondence can be deduced from these axioms, this correspondence is valid concerning the axioms. Otherwise, the correspondence leads to inconsistency.

3 Mapping and Enrichment Tools

The previous chapter sets up the background of the ontology matching system GOMMA (Generic Ontology Matching and Mapping Management). Furthermore, STROMA (Semantic Refinement of Ontology Mappings) is introduced, which refines a given mapping by adding a semantic relation type to each correspondence.

3.1 Short Introduction to GOMMA

It is rarely the case, especially within the life sciences, that a designed ontology is accepted without modifications for a long period of time. Rather, ontologies are updated frequently, e.g. the gene ontology consortium[1] daily publishes a new ontology version, whereas a new version of NCIt is created every month. Nevertheless, the new version only partly differs from the previous one. For instance, a concept is added or deleted, two concepts are merged to a new one, or a concept is split up. In order to model the version management of ontologies, a timestamp t is added to the definition of an ontology O of version v – resulting in a quadruplet $O^v = \langle C^v, A^v, R^v, t \rangle$ with C^v/ A^v/ R^v as the set of concepts/ attributes/ relations of version v at time t [Gro14, 59].

3.1.1 Basic Architecture

That ontologies change from version to version is called ontology evolution. GOMMA [KGHR11] is an infrastructure which analyzes such evolutions.[2] Its structure is depicted in figure 3.1. GOMMA contains three layers: the repository level, the functional component level, and the tool level. The first layer manages the data, i.e. the ontology, mapping, and entity source versions, respectively. The key assumption is that versioning is linear, which means that a specific version is preceded by at most one version and succeeded by at most one version, resulting in a "chain" of versions. Hence, GOMMA stores a concept (or entity) and additionally its life time, i.e. at which time t_{start} it was added to the ontology and since which time t_{end} it has been no longer valid. Due to this information, it is easily calculate which information is relevant and valid for building an ontology version at time t.
The second level represents the three functionalities of GOMMA and their APIs. First, the match function determines an alignment between two given ontologies. Various similarity and distance measures are taken into account, see "Techniques behind Matching" in subsection 2.4.2. Second, the DIFF function determines an evolution mapping according to which it can be inferred which elements have changed (addition, deletion, split, merge) from one version to another. Third, the evolution function analyzes the whole history of an ontology (not only two succeeding ontologies as it is be done by DIFF) and may allow statements about which parts of an ontology are stable over time, i.e., which subgraph of the ontology seldom changes, and which parts are subject to frequent changes.

[1] http://geneontology.org/
[2] http://dbs.uni-leipzig.de/de/gomma

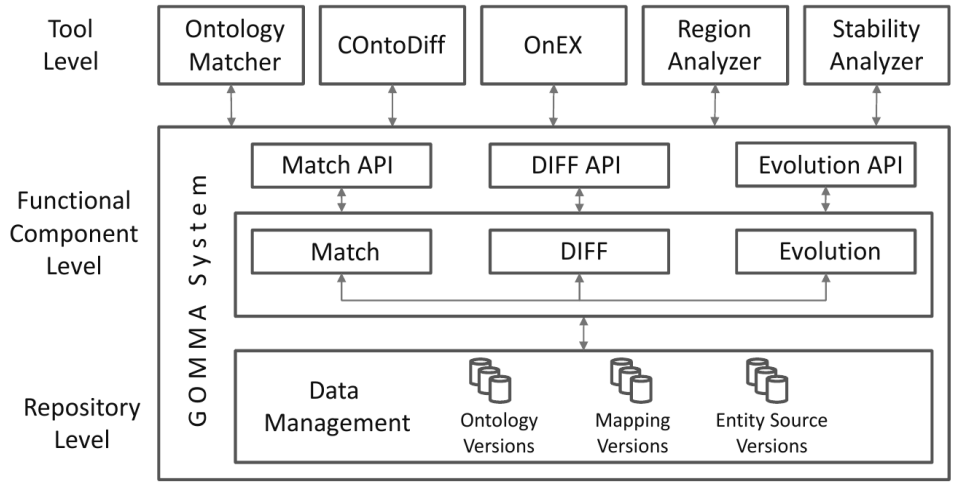

Figure 3.1: Overview of the GOMMA infrastructure with its three levels, its functional components on the middle level, and the data management system [KGHR11, 5].

At the third layer, the top level, tools are embedded which enable an elaborate access to GOMMA. The ontology matcher, for instance, determines correspondences between two ontologies, and OnEX visualises the changes within one ontology.

3.1.2 The GOMMA Format of Mappings

GOMMA returns a mapping in XML format. An example is given in listing 3.1. The opening mapping tag (see line 2) contains attributes which specify the mapping such as its name and its class (e.g. ontology mapping or annotation mapping). The threshold θ is stated in line 3 as the value of *minConfidence*. Further meta information is given in line 4 to 6 regarding the source ontology and in line 7 to 9 regarding the target ontology (e.g., their name and version). After that the correspondence set is indicated. A correspondence is given in line 12 to 19. The confidence vale ξ is given as *confidence* the attribute of the opening tag. The attribute *corr_type* will be setted with the semantic relation type. As a correspondence may be one-to-many or many-to-one, there are tags for introducing the set of source and target concepts, respectively (line 13/ 15 and 16/ 18). Between these tags the concept, called *object*, is stated (line 14 and 17, respectively). The ID of the object is given as the *accession*. Further correspondences are omitted.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <mapping baseName="AnatomicalEntity@AdultMouseAnatomyOntology-HealthEntity@
      NCIThesaurus_MA_NCIT_gomma_NameSyn_0.8"
      versionName="AnatomicalEntity@AdultMouseAnatomyOntology[2007-01]-HealthEntity@
      NCIThesaurus[2006-02]_MA_NCIT_gomma_NameSyn_0.8" timestamp="2007-1-1"
      is_instance_map="false" mapping_class="ontology_mapping"
      mapping_type="corresponds_to" mapping_tool="GOMMA"
      mapping_method="MA_NCIT_gomma_NameSyn_0.8">
3 <metadata minConfidence="0.8" minSupport="1">
4 <domain_sources>
5 <source objecttype="AnatomicalEntity" name="AdultMouseAnatomyOntology"
      timestamp="2007-01-01" version="2007-01" is_ontology="yes"
      structural_type="directed_acyclic" url="" />
6 </domain_sources>
```

```
<range_sources>
<source objecttype="HealthEntity" name="NCIThesaurus" timestamp="2006-02-01"
    version="2006-02" is_ontology="yes" structural_type="directed_acyclic" url=""
    />
</range_sources>
</metadata>
<correspondences>
<correspondence support="2" confidence="1.0" user_checked="0" corr_type="N/A">
<domain_objects>
<object accession="MA:0000280" objecttype="AnatomicalEntity"
    source_name="AdultMouseAnatomyOntology" />
</domain_objects>
<range_objects>
<object accession="http://human.owl#NCI_C12784" objecttype="HealthEntity"
    source_name="NCIThesaurus" />
</range_objects>
</correspondence>
[...]
</correspondences>
</mapping>
```

Listing 3.1: Extract of a GOMMA mapping file.

3.2 Short Introduction to STROMA

The following lines are an introduction to STROMA as it is presented in [Arn15, AR14, AR13]. STROMA is a tool for determining semantic relation types within an a priori given ontology mapping. For each correspondence of the mapping it is calculated due to linguistic methods and background knowledge which type (*is-a*, for instance) this correspondence has. STROMA evaluates its type denotation by means of a gold standard which has been committed as further input.

3.2.1 Basic Architecture

STROMA pursues a two level approach, i.e., in the first and previous step a common match tool, like GOMMA, determines a set of correspondences as the match result, possibly by means of background knowledge. This set is input for STROMA, which computes in the second step the enriched mapping. The overall workflow is depicted in figure 3.2.

type	abbreviation	explanation	example
equal	0	c_1, c_2 are synonyms	$\langle$*cardiac chamber*, *chamber of heart*$\rangle$
is-a	1	c_1 is a hyponym of c_2	$\langle$*heart*, *muscular organ*$\rangle$
inverse is-a	2	c_1 is a hyperonym of c_2	$\langle$*muscular organ*, *heart*$\rangle$
has-a	3	c_1 is holonym c_2	$\langle$*heart*, *cardiac chamber*$\rangle$
part-of	4	c_1 is meronym of c_1	$\langle$*cardiac chamber*, *heart*$\rangle$
related	5	c_1 is a cohyponym of c_2	$\langle$*leucocyte*, *erythrocytes*$\rangle$

Table 3.1: The possibly semantic relation types between concept c_1 and c_2 in STROMA.

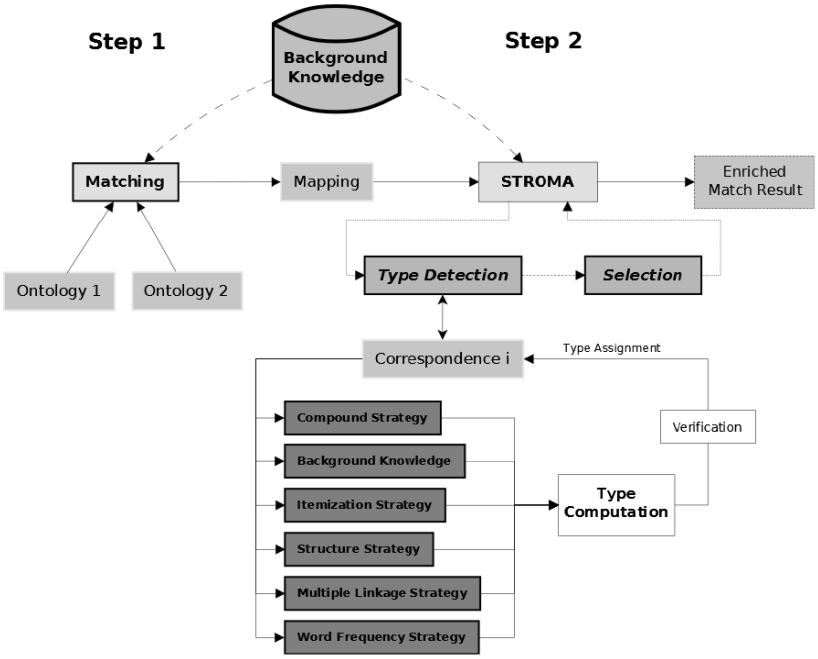

Figure 3.2: Basic workflow and components of STROMA, after [Arn15, 11].

The second step is divided into two parts: i) type detection, and ii) selection. Within type detection at least one of six type detection strategies (compound strategy, background knowledge, itemization strategy, structure strategy, multiple linkage strategy, word frequency strategy) is applied to each correspondence. If more than one strategy is used, the single results have to be aggregated in order to yield exactly one type (type computation), see below.
Furthermore, there are two post-processing control modules (verification, selection) which check whether a correspondence with its assigned type is valid.
STROMA assigns a relation type t to a mapping correspondence out of five different types plus the label *undecided*. Each type is abbreviated by a natural number. The types of a correspondence $\langle c_1, c_2 \rangle$ are depicted in table 3.1. The types are defined by common linguistic relations of hyperonymy, meronymy, and synonymy. Each non-symmetric relation type has an inverse type: *is-a* and *inverse is-a* as well as *has-a* and *part-of*.

3.2.2 Strategies for Semantic Type Detection

STROMA uses six strategies [Arn15, 13ff][AR15, 10ff] in order to determine the semantic type of a correspondence. The strategies and the types which they are able to detect are listed in table 3.2 and described briefly in the following. Note that since *is-a* and *part-of* are no symmetric relations, their inverse counterpart is named *inverse is-a* and *has-a* and is subsumed in the table under *is-a* reps. *part-of*. If a strategy cannot return a type of the correspondence because e.g. the strategy cannot be applied to this correspondence, a strategy returns *undecided*. STROMA enables to set a default value instead, namely *equal*, because matching tools normally connect concepts which are equal. This setting is called *undecided-as-default*. Contrarily, if *undecided* is returned as it is, the setting is called *undecided-as-false*.

		denoted types			
strategy	weight	equal	is-a	part-of	related
compound strategy	1.0	–	✓	–	–
background knowledge	0.9	✓	✓	✓	✓
itemization strategy	1.0	✓	✓	–	–
structure strategy	0.7	–	✓	✓	–
multiple linkage strategy	0.8	✓	✓	–	–
word frequency strategy	0.6	✓	✓	–	–

Table 3.2: STROMA strategies, their weight and denoted types.[AR14, 11].

Each strategy has a specific weight, see table 3.2, which is important for determining the overall result type. After all strategies have returned a type, for each returned type all weights of those strategies are summed up which have "voted" for this type. The type with the highest weight is stored as the semantic type of the current correspondence.[3]

Compound Strategy A compound consists of a head h, which carries the main meaning, and some modifier m, which slightly "restricts" the denotation of h.[4] For example, *hand joint* is a compound with $h = joint$ and $m = hand$, hence it is a specific joint, namely that of a hand. If a correspondence is given which relates the compound to its head (or the head to the compound), the compound strategy returns *is-a* and *inverse is-a*, respectively, as the semantic type of this correspondence, e.g. $\langle hand\ joint, is\text{-}a, joint\rangle$.[5]

Background Knowledge The background knowledge resources are stored in SemRep (Semantic Repository) [AR15]. Given a correspondence from STROMA SemRep calculates the semantic type of this correspondence referring to varying knowledge sources like WordNet, UMLS, or information extracted from Wikipedia. The background knowledge is modelled as a graph where nodes represent a concept(-label) and edges symbolize that there is a relation of a specific type (*is-a*, ...) between the nodes of the edge. Since background knowledge is a powerful strategy, each relation type is a possible outcome. Furthermore, background knowledge helps to determine relation types between concepts where linguistic or structural strategies fails, e.g. $\langle serum, part\text{-}of, blood\rangle$. Nevertheless, background knowledge fails if the concept is not entailed in SemRep or if there is no edge between the two concepts under consideration. The SemRep result of querying for the relation between *serum* and *blood* is depicted in figure 3.3. Three paths with one or two edges between the concepts are found. A path is represented by a relation and the source of this relation, which is WordNet in all cases of the example. Besides the path, the type as well as the confidence value for this relation is shown.

[3]Note that i) *undecided* is returned iff no strategy has determined a type other than *undecided*; ii) if two types have the same weight, the most prominent relation type (according to the hierarchy *equal* > *is-a* > *inverse is-a* > *part-of* > *has-a* > *related*) is returned.

[4]There are exceptions to that generalisation, e.g. *bitter-sweet*.

[5]Relating the compound to its modifier the resulting type is not deterministic predictable: $\langle hand\ joint, part\text{-}of, hand\rangle$ vs. $\langle bookstore, has\text{-}a, book\rangle$ vs. $\langle headline, ?, head\rangle$, see discussion in [Arn13, 17f].

```
Path:  serum [IS_A | WORDNET] body fluid [INVERSE_IS_A | WORDNET] blood
Type:  RELATED
Conf:  0.74916225
-----------------------------------------------------------------------
Path:  serum [EQUAL | WORDNET] blood serum [PART_OF | WORDNET] blood
Type:  PART_OF
Conf:  0.8848987799999999
-----------------------------------------------------------------------
Path:  serum [PART_OF | WORDNET] blood
Type:  PART_OF
Conf:  0.9341999999999999
```

Figure 3.3: Example of a SemRep query to the relation type between *serum* and *blood*.

Itemization Strategy Itemization is understood as a list of terms, e.g. *body fluid or substance* (MA:0002450), *head and neck* (MA:0000006), or *Fever, Sweat, and Hot Flashes* (NCIt:C115213). The itemization strategy maps the itemization to its item set, i.e., a set of the terms which make up the itemization. In the NCIt example the corresponding item set is $\{Fever, Sweat, Hot\ Flashes\}$. If at least one concept is an itemization, the following reducing steps are applied in exactly that order (note that a single-term concept is an item set with exactly one element) [AR15, 14f]:

1. Intra-Synonym Removal:
   In each item set I replace the items $i_1, i_2 \in I$ by i_1 if i_1 and i_2 are synonyms.
2. Intra-Hyponym Removal:
   In each item set I remove an item $i_1 \in I$ for which there exists a hypernym[6] $i_2 \in I$.
3. Inter-Synonym Removal:
   Remove each item $i_1 \in I_1$ and $i_2 \in I_2$ if i_1 and i_2 are synonyms.
4. Inter-Hyponym Removal:
   Remove each item $i_2 \in I_2$ if there exists a hypernym $i_1 \in I_1$ and vice versa.

For instance, let I_1 be the set $\{temperature, fever, cold\ sweat\}$, I_2 the set $\{fever, sweat, hot\ flashes\}$. Since *temperature* and *fever* are synonyms, I_1 is reduced to $I_1 = \{temperature, cold\ sweat\}$ (intra-synonym removal). The second removal rule is not applied as there are no intra-item set hyponyms. Inter-synonym removal leads to $I_1 = \{cold\ sweat\}$ and $I_2 = \{sweat, hot\ flashes\}$. Finally, inter-hyponym removal sets $I_1 = \emptyset$ because *sweat* is a hypernym of *cold sweat*.
After reducing the item sets the semantic type is determined along the following rules – the fourth case returns the type which is determined by the background knowledge strategy[7]:

- $I_1 = \emptyset \wedge I_2 = \emptyset$ $\Rightarrow$ type: *equal*
- $I_1 = \emptyset \wedge I_2 \neq \emptyset$ $\Rightarrow$ type: *is-a*
- $I_1 \neq \emptyset \wedge I_2 = \emptyset$ $\Rightarrow$ type: *inverse is-a*
- $|I_1| = 1 \wedge |I_2| = 1$ $\Rightarrow$ apply background knowledge strategy
- $|I_1| > 1 \vee |I_2| > 1$ $\Rightarrow$ type: *undecided*

[6] A word w is an hypernym of a word $\bar{w}$ if the denotation of w entails the denotation of $\bar{w}$, e.g.: *mammal* is a hypernym of *human*.

[7] No other strategy is possible as the only expected types are *part-of* and *has-a*, respectively. *(inverse) is-a* is excluded because that would presuppose a hypernym but all hypernym relations have already been eliminated.

Structure Strategy The structure strategy takes account of the path of two concepts. Assume that c_i and $\bar{c}_n$ are the relevant concepts. Then the path from their parent node to theirselves is $c_{i-1}.c_i$ and $\bar{c}_{n-1}.\bar{c}_n$, respectively. Knowing the type of the relation from one node to the parent of the other one allows inference to the type of the correspondence under consideration. For example, if c_i equals $\bar{c}_{i-1}$, the conclusion is that $\bar{c}_i$ is a c_i. The argument is that child and parent node are in an *is-a* relation and hence $\langle \bar{c}_i, \textit{is-a}, \bar{c}_{i-1} \rangle$. As $\langle c_i, \textit{equals}, \bar{c}_{i-1} \rangle$, it holds that $\langle \bar{c}_i, \textit{is-a}, c_i \rangle$.

Multiple Linkage Strategy If one concept $c \in O_1$ is mapped to multiple concepts $k_1, k_2, \ldots k_n \in O_2$, it can be (heuristically) assumed that c is more general than $k_1, k_2, \ldots k_n \in O_2$ and hence *is-a* (e.g. $\langle k_1, \textit{is-a}, c \rangle$) or *inverse is-a* (e.g. $\langle c, \textit{inverse is-a}, k_1 \rangle$) is returned.

Word Frequency Strategy Given a corpus a word can be assigned a frequency, which means the absolute number of occurrences within the corpus. Heuristically it is assumed that the more frequent a word is the more general is its meaning. Be $w_<$ a less frequent word and $w_>$ a more frequent one. If a correspondence between $w_<$ and $w_>$ is given the word frequency strategy returns *is-a* and *inverse is-a* for a correspondence between $w_>$ and $w_<$, respectively.

3.2.3 Post-processing: Control Type Computation

As mentioned earlier there are two modules which check whether the determined type is valid, or not. Type verification applies immediately after each computation of a correspondence. As the strategies above (except structure strategy) do not take the concept path into account, this verificator reconsiders whether the type assignment is still correct if path information is available. Otherwise the verificator changes the type. For instance, let p_1 be *arm.elbow.joint* the first concept path and p_2 be *arm.elbow joint* the second one. Considering only the leafs *joint*, *elbow joint* the type *inverse is-a* is computed. But obviously the leaf *joint* denote the joint of the elbow. Hence, the verificator would change the type to *equal*.
Within the selection module, which is executed after type detection for all correspondences is finished, correspondences are filtered out for which there is not enough evidence that they might be correct. In order to achieve this a threshold θ is implement such that all correspondences with a confidence value ξ equal or greater θ are surely maintained within the mapping. All other correspondences with a value lower θ are removed if and only if their relation type is *undecided*, or their mapping lack linguistically motivated evidence.

4 Semantic Enrichment of GOMMA Mappings

This chapter deals with the aspects of implementing a tool, called GOMET (Gomma Mapping Enrichment Tool), which enables the semantic enrichment of a GOMMA mapping by means of STROMA. Firstly, since STROMA has been applied to generic real life ontologies like ontologies on clothing, its biomedical background knowledge stored in SemRep contains only a small number of specific concepts and their relations within the biomedical domain. Hence, it was necessary to integrate further background knowledge to SemRep, in concrete words: SemRep was expanded by UMLS. The expansion enabled STROMA to apply the background knowledge strategy for semantic type denotation for more valid results. Secondly, the actual implementation of GOMET is introduced.

4.1 UMLS as Background Knowledge Source

The following paragraphs outline the integration of UMLS concepts and relations into SemRep. Therefore, subsection 4.1.1 gives an overview about the structure and intention of UMLS. In subsection 4.1.2 the extraction of relevant data from UMLS is described. Finally, a brief evaluation of the expanded SemRep system is presented, subsection 4.1.3.

4.1.1 Overview about UMLS

UMLS is a project of the U.S. National Library of Medicine which seeks to unify different sources of biomedical and health data in order to achieve interoperability between computer systems. UMLS consists of three knowledge sources: i) Metathesaurus: a database which combines the muli-lingual vocabularies of different ontologies, e.g. MeSH (Medical Subject Headings), SNOMED CT. Hence, it stores concepts and the relations among them. ii) Semantic Network[1]: as each concept within

concepts (CUIs)	2,930,638
concept names (AUIs)	11,399,740
distinct concept names (SUIs)	9,487,373
distinct normalized concept names (LUIs)	8,487,373
sources	168
languages	21

Table 4.1: General statistics of the UMLS 2013AB. The numbers denote the count.

[1] http://semanticnetwork.nlm.nih.gov/

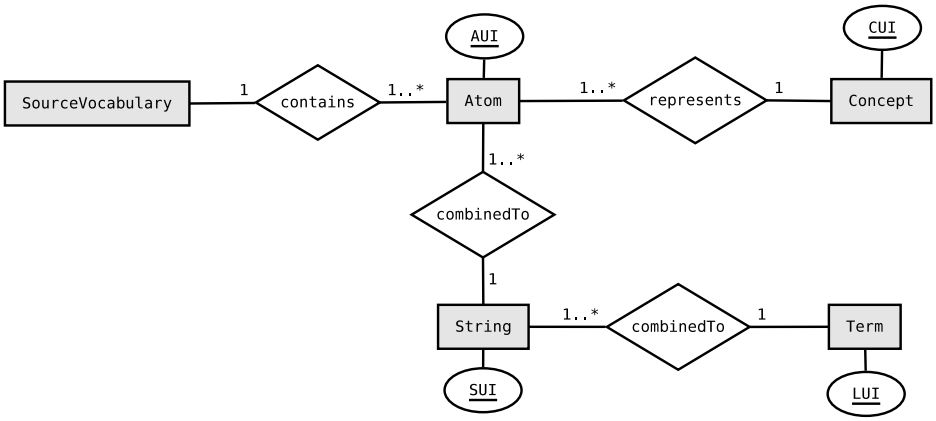

Figure 4.1: ER model of the basic notions of the UMLS metathesaurus. Only the identifier attributes are depicted to ensure a good readability.

the metathesaurus is categorized by a semantic type, the semantic network consists of a set of all such types as well as a set of relationships between these types. iii) SPECIALIST Lexicon and Lexical Tools[2]: a tool set for natural language processing, especially developed for mediating between everyday natural language and biomedical terminology. The metathsaurus is the core of UMLS and generated by use of the semantic network and the lexical tools.

UMLS is updated twice a year. The release version which underlies this thesis is the second one of 2013 (2013AB). Table 4.1 gives a short overview about the UMLS data: There are approximately 3 million concepts in 10 million different spellings extracted from 168 sources of 21 languages. The main part of UMLS is English (75% of all names are English ones).

The most basic unit of UMLS is an "atom", i.e. the ouccurence o of a particular string s of a particular concept c within a particular source vocabulary v. A unique identifier AUI (Atom Unique Identifier) is assigned to that occurence. Each string s is assigned a unique identifier SUI (String Unique Identifier). Two concept strings are different if they vary in their character sets, thus respecting lower vs. upper case, punctuation etc. Note that s can be ambigous – cf. the example of *ICD* in subsection 2.4.2; whereas it is always clear which concept is represented by o since o is a contextualized string. The concept which is denoted by o is uniquely identified by its CUI (Concept Unique Identifier). Obviously, a CUI can be associated with more than one AUI but one AUI is associated with exactly one CUI. An LUI (Lexical Unique Identifier) captures different linguistic variants of two or more strings. Similar to an SUI it can be associated with more than one CUI. Figure 4.1 shows an ER model of these circumstances. Note that only CUIs are language independent. AUIs, SUIs, and LUIs are language sensitive due to the source they are extracted from.

In figure 4.2 an example of the UMLS structure is depicted. Two concepts are shown with a set of LUIs. The LUI L0003792 is part of both concepts. Thus, that LUI denotes an ambiguous word, namely *arm*. The first reading is the default reading. It denotes a body part. The second reading describes a treatment plan within a clinical study. In this context, *arm* is a shortening of *protocol treatment arm*. As each AUI is contextualized, the AUI sets of LUI L0003792 for SUI S00155710 are distinct. The sets of SUIs for the same LUI only have to differ. It is possible that their intersection is not empty – like in this case for S00155710.

[2]http://lexsrv3.nlm.nih.gov/Specialist/Home/index.html

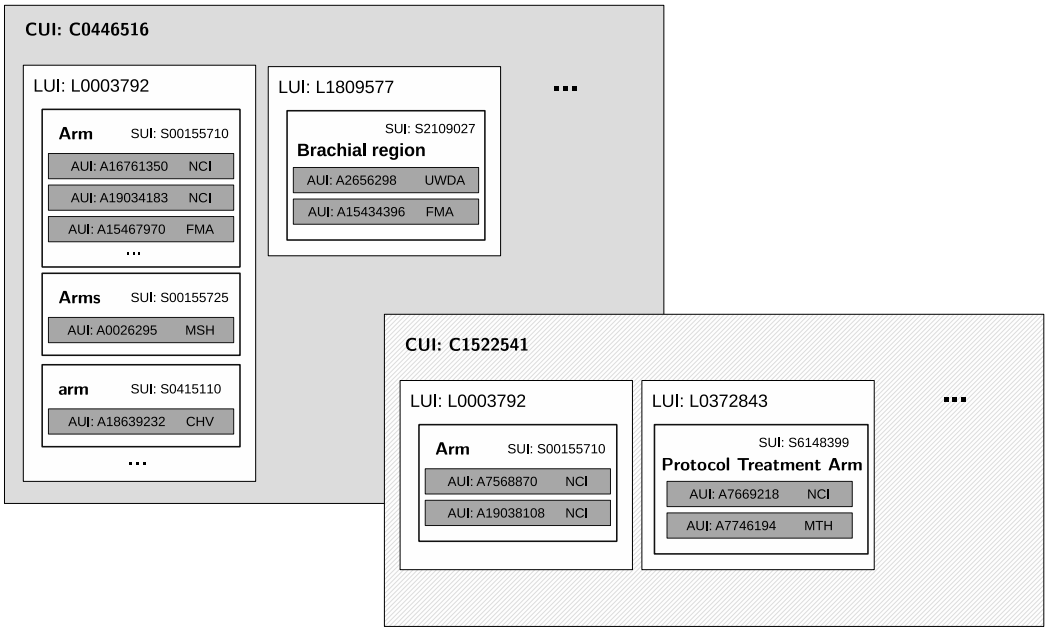

Figure 4.2: UMLS example: LUI L0003792 occurs in both concepts. It denotes an ambiguous word.

REL	count	%	definition
SIB	22,624,718	39.0%	has sibling
RO	15,065,527	26.0%	has relation other than synonymous to
SY	5,320,366	9.2%	source asserted synonymy
CHD	4,392,763	7.6%	has child
PAR	4,392,763	7.6%	has parent
RB	1,700,628	2.9%	has broader relation to
RN	1,700,628	2.9%	has narrower relation to
RQ	1,615,330	2.8%	related (possibly synonymous) to
AQ	606,008	1.0%	allowed qualifier
QB	606,008	1.0%	can be qualified by
Σ	58,024,739	100%	

Table 4.2: The relation types of REL and their count.

Two AUIs can be connected via an explicit mentioned relationship or via an implicit (synonym) relation. The last one can be inferred as two AUIs are connected to the same CUI. The former one is explicitly stored in the database and may be non-synonym. Two CUIs are always explicitly related. The most relevant tables within the UMLS database for the current purpose are MRCONSO and MRREL. MRCONSO contains a record for each AUI. It specifies among others the language, CUI, LUI, SUI, the vocabulary source, and the actual string of an atom. MRREL stores the relations between two CUIs and AUI, respectively. Besides the identifiers and other attributes it contains the name of the relationship (REL) which describes the basic nature of a relation, see table 4.2. Most of the 58 million relations (65%) are either the sibling relation or an other (not specified relation) than synonymy. Note that symmetric relations are counted twice, e.g., $\langle a, SY, b\rangle$ as well as $\langle b, SY, a\rangle$ are stored. Furthermore, MRREL contains a more specific relationship label (RELA). Exemplary values are *isa*, *inverse isa*, *translation_of*, *ingredient_of*. Out of the 58 million relations within MRREL around half (28 million) are described further by a RELA value unequal null. There are 652 different types of RELA. Most of them appear in less than 1% of all cases.
There are 684 different pairings of REL and RELA (ignoring the cases where RELA is null). As there are 10 REL and 652 RELA types, up to $10 \cdot 652 = 6520$ possible pairings could be expected. But only 10% of these pairings appears. Obviously, this is due to the meaning of REL and RELA. For instance RO (denoting a relation other than synonymy) and *same_as* (denoting something like synonymy) are not compatible to each other.

4.1.2 Relation Extraction from UMLS

For this thesis UMLS is given within a MySQL database. Thus, the data are accessed via JDBC (Java Database Connectivity) queries and are filtered in Java. Two ways are implemented how to extract the relevant relations for SemRep. First, all relations are relevant whose relation label RELA in MRREL can be reliably associated with a semantic type of STROMA. These relations are selected from the UMLS database by SQL. For instance, *used_for* matches *equal*. However, most relation types, like *biological_process_has_result_anatomy*, do not correspond to a STROMA relation type since they are too fine-grained. Exception to this first case are relations which link concepts or concept names which are taken from different languages than English. Thus, although *translation_of* is an equal relation, such relations are not integrated into SemRep since only English-language ontologies are considered. Consequently, seven relation types are chosen, as they are listed in table 4.3. Second, an equal relation between two concept strings is extracted if the concepts strings are mapped to the same CUI. Thus, the SQL query select distinct CUIs and the associated concept strings. In Java they are grouped by the CUI. The resulting *equal* relations are filtered. That procedure returns relations which have not been extracted by the first

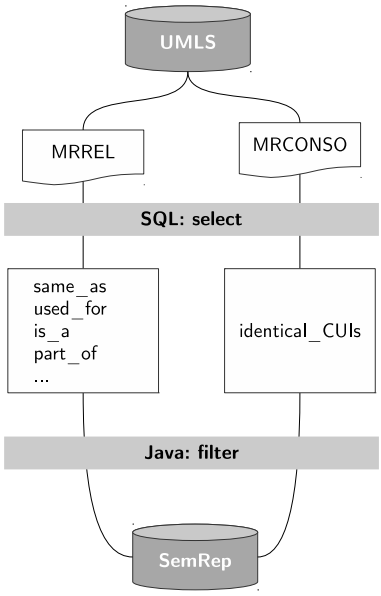

Figure 4.3: Workflow of the integration of relations from UMLS tables to SemRep.

procedure, for example ⟨*menstruation disturbances*, *equal*, *periods problems*⟩. In the following, this set of relations is abbreviated *identical_CUIs*. The workflow is depicted in figure 4.3.
The attribute REL is not used for extracting relations of UMLS for SemRep. That method has two reasons: On the one hand, the REL values are too general. It is unclear to which STROMA relation type they should be mapped. For example, given a child relation CHD between two concepts it can be mapped in some cases to *is-a* in other cases to *part-of*. Even worse is the case for RO relations, see table 4.2. As RO denotes that the relation is not synonymy, it is absolutely unclear to which STROMA type the relation has to be mapped. On the other hand, the REL values may represent unwanted relations. The SY relation consists to 45% of the RELA *translation_of* and *has_translation_of* which should not be part of the background knowledge for an English-language task. These problems with REL lead to the consequence that it seems to be more appropriate to choose the RELA column in order to specify the relations that should be integrated to SemRep. Additionally, that method can be combined – as it maximally covers a half of all UMLS relations – with the extraction of equal relations between concept strings with the same CUI.
Each relation is transformed into the format of SemRep, namely $s :: o :: t$ where s is the subject, o the object, and t the type of the relation. t is a natural number taken from table 3.1. For example, *fever* :: *disease* :: 1 which says that fever is a disease.
In order to achieve a good performance of SemRep it is recommended that it contains as much relations as possible but not more relations than necessary. Hence, the extracted relations are filtered which results in more "useful" and "cleaned" data. The filter techniques are described in the following. These techniques are rules whose order of application is important. The result of the filter is shown in table 4.3. At first, the first part of techniques are listed which lead to the deletion of the whole relation:

ENG A priori only English concept names are considered as only English-language ontologies are taken into account.

LONG "Long" relations, i.e., relations with more than 200 characters in their SemRep format, are deleted. Due to their size these relations contain complex n-word concept names and thus, cannot be effectively computed by SemRep. Example:
⟨*4-alpha-D-{(1->4)-alpha-D-glucano}trehalose trehalohydrolase activity*, *is-a*, *hydrolase activity, hydrolyzing O-glycosyl compounds*⟩

UNIT All relations are deleted which enclose at least one concept which contains a unit like *ml* or *milligram*. Units are rather part of instances than of concepts. Thus, such relations are not relevant and may be excluded. Example:
⟨*anadrol 50 50mg tablet*, *equal*, *oxymetholone 50 mg oral tablet*⟩

BRACE Relations with '{' or '}' as part of the concept name are not relevant. They are deleted. Example: see LONG.

CONJ SemRep cannot handle complex concept names containing a conjunction. Thus, such relations are deleted. Example:
⟨*vestibulocochlear nerve and its branches*, *is-a*, *vestibulocochlear nerve structure*⟩

PUNC Relations with special punctuation (';', ' + ', '<', ...) are deleted to, as they imply too complex concept names or instances of a concept. Example:
⟨*alprostadil 500mcg/mL injection*, *is-a*, *prostaglandin E>1<preparation*⟩.

PATT The last types of relations which are deleted during the postprocessing are such ones which match a further pattern. The crucial point is that the patterns are context-sensitive and it is not possible to check whether a particular substring is part of the relation. Thus, they have to be implemented as regular expressions. For instance, as ':' is part of each relation due to the STROMA format definition, the deletion of correspondences with a double dot as part of a concept name is handled by PATT. Example:
⟨*hla c*03:15 antigen*, *equal*, *hla cw*0315 antigen*⟩

Secondly, the following techniques do not cause deletion but lead to modification of the relation string. Obviously, it is more effective to apply them after the first block of deleting rules. The reason is that it is unnecessary to modify a relation which is later deleted anyway.

LOW All relations are transformed to lower case. This part of the normalization.

BRACK Brackets of sort '[', '(', ']', ')' (including their content) are deleted as they contain additional information and hamper a good performance of SemRep. Example:
⟨*biotin-[pyruvate-carboxylase] ligase activity*, *is-a*, *biotin-protein ligase activity*⟩

ADD Additional fragments, which starts with '-', are deleted. Such fragments are not part of the concept name as they contain domain information, for instance. Example:
⟨*microbiology - prostatic fluid culture mycoplasma*, *is-a*, *prostatic fluid culture*⟩

HYPH All hyphens are replaced by a blank character. This part of the normalization.

NORM All sequences of two or more blank characters are reduced to a single one. This part of the normalization and may be necessary due to one of the previous modification rules.

Now further rules are applied, which again delete whole relations. It is necessary that they are taking place after the previous modification rules because these rules establish the context for the adequate application of the following deleting rules.

X_X After the modifications, as described above, this rule applies which deletes all relations with now identical subject and object. This may be caused by deleting brackets or additions.

TRANS If the relation is extracted as *identical_CUIs* relations are excluded which can be inferred due to transitivity, e.g. given $\langle a, equal, b\rangle$, $\langle a, equal, c\rangle$, $\langle b, equal, c\rangle$ the last relation is deleted since the equality between b and c can be inferred given the first two relations.

SYM All symmetric relation types (*same_as*, *related_to*, *identical_CUIs*, ...) are checked in the following way: if both $\langle a, t, b\rangle$ and $\langle b, t, a\rangle$ are in place, then the second one is deleted.

4.1.3 Interim Evaluation of Integrating UMLS to SemRep

All in all, more than 2.5 million relations are integrated to SemRep. Nearly 45% of them are based on the RELA attribute. That are 9% of all relations in UMLS with a RELA attribute not null (assuming that symmetric relations are only counted once and non-symmetric relations, e.g. *isa*, are not counted within their inverse relation, *inverse_isa*). All other relations which are based on RELA are either too specific or useless for the given task. The second extraction way leads to a huge amount of relations. Thus, approximately 55% of all new SemRep relations link concept names due to their identical CUI (*identical_CUIs*). These results are shown in figure 4.3. For each

UMLS relation	STROMA relation	count after filter	% after filter
0. identical_CUIs	equal	1,497,789	22%
1. has_expanded_form	equal	68,882	16%
2. same_as	equal	28,627	18%
3. used_for	equal	11,895	91%
4. is_a	is-a	1,012,798	38%
5. has_physical_part_ of_anatomic_structure	part-of	1,996	26%
6. part_of	part-of	75,492	45%
7. related_to	related-to	2,463	21%
Σ		2,699,942	

Table 4.3: The extracted relations from UMLS. Line 0 represents the relations due to extraction way one, lines 1-7 due to RELA.

relation their corresponding STROMA type as well as the number of relations after filtering are depicted. The last column states the ratio between relations after and before filtering.
In average the ratio of the filter is 34.6% (standard deviation: 23.3%). Generally, the symmetric relations *identical_CUIs*, *same_as*, *related_to* show a lot of deleted relations due to the SYM technique which does not apply to other relations. Two relations step out of line: First, *has_expanded_form* has astonishingly a lot of relations which are filtered out (84%). Table 4.4 shows which deletion rules apply to how many relations. Additionally, it is depicted how many relations of *has_expanded_form* are affected by each deletion rule, and how many relations of all relations which are deleted by a specific rule are deleted within *has_expanded_form*. Most relations are deleted because of the PUNC technique but there are other relation types where much more relations (relatively and absolutely) are deleted due to this rule. As it can be seen within *has_expanded_form* the most amount of BRACE (86%), LONG (71%), and PATT (81%) application take place. That the LONG rule applies such frequently is suggested by the semantics of this relation type because a short term is mapped to a long one as the latter is the expanded form of the former.

deletion rule	count	ratio within relation	ratio of deletion type
BRACE	786	0%	86%
CONJ	77,520	22%	45%
LONG	45,871	13%	71%
PATT	50,020	14%	81%
PUNC	156,862	45%	40%
UNIT	1,304	0%	6%
X_X	15,580	4%	17%

Table 4.4: The *has_expanded_form* relation type which has been extracted for integration to SemRep with the deleting rules that apply to its relations.

deletion rule	count	ratio within relation	ratio of deletion type
CONJ	62	6%	0%
PATT	2	0%	0%
PUNC	1004	91%	0%
UNIT	21	2%	0%
X_X	13	1%	0%

Table 4.5: The *used_for* relation type which has been extracted for integration to SemRep with the deleting rules that apply to its relations.

Second, consider the relation type *used_for* which is most stable against deletion. As it is depicted in table 4.3, after filtering still 91% of the original relation set remains. Thus, only 9% of its relations are deleted. Table 4.5 show the same kind of data analogue to *has_expanded_form*. Only one deletion rule is mainly applied, namely PUNC. Furthermore, the executed deletions do not make a substantial contribution to the overall deletion number per technique (all ratios are approximately 0%). Besides this descriptive evaluation of deletion rules no further analysis has been undertaken. Some interpretation of the correlation between deletion rules and relation type might be enlightening to a certain degree. Nevertheless, we leave this open for further investigations as it goes beyond the scope of this work.

Problematic for a reasoning task is the fact that language is ambiguous, for example due to shortening a term within a given context. This leads to relations which seems to be wrong without context. Such relations, which are not true in general, as background knowledge may mislead the SemRep algorithms when computing the relation type between two given concepts. There is no way to identify the misleading relations by means of their structure – i.e. without using SemRep or another background knowledge tool. Nevertheless, anticipating the evaluation of the mapping enrichment in chapter 5, it has not been found any evidence that such relations actually misled the type detection. Eventually, one example of such relations is taken from *used_for*. Given ⟨*vein*, *equal*, *uterine vein*⟩, ⟨*uterus*, *equal*, *uterine vein*⟩, generally it should be concluded that*vein* is equal to *uterus*. Obviously, *vein* and *uterus* should not be in an equal relation. The overall explanation of this strange behaviour is that *used_for* may not be a symmetric relation and it is an oversimplification to interpret it as an equal relation.

Figure 4.4 gives an overview about the proportion of relation types which are imported into SemRep. Most of them represent an equal relation, among other things, because a lot of a data is extracted by means of *identical_CUIs* which leads only to equal relations. One fifth is a set of *is-a* relations, and the rest constitutes *part-of*. Although *related-to* was also extracted, these relations are not imported to SemRep as SemRep do not store this kind of relation type directly but only indirectly via the relation two concepts have according to a third shared parent concept. For instance, ⟨*mammal*, *is-a*, *vertebrate*⟩ and ⟨*bird*, *is-a*, *vertebrate*⟩, hence ⟨*mammal*, *related-to*, *bird*⟩. Comparing the extracted UMLS relations with the already given relations from Wikipedia it is observed that there are no relations within the intersection of both relation sets. Hence, all relations from UMLS has not been integrated to SemRep before.

[AR15] presents a diseases mapping for his evaluation. A comparison of his results with the results after integrating the UMLS background knowledge to SemRep shows a slight improvement in the

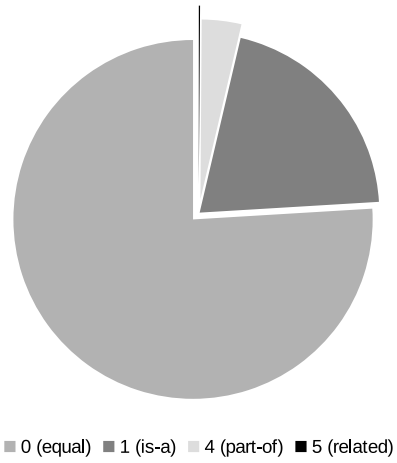

Figure 4.4: Ratio of the extracted UMLS relations. 76.0% represent equal relations, 20.4% is-a relations, 3.6% part-of relations, and 0.1% are related-to relations.

recall for non-trivial types (from 65.9 to 70.7) and hence a greater f-measure (46.4 instead of 44.2). Even better results can be achieved for deactivating the node degree heuristic and allowing a maximal path length of three, see section 5.1. In the non-trivial case this leads to a recall of 73.2, a precision of 60.0 as well as an f-measure of 65.9. For trivial correspondences holds that the recall amounts to 94.3, the precision to 96.8, and the f-measure to 95.5. Hence, most values significantly increased, except the trivial precision, which remains constant. This should be kept in mind for the discussion of mapping enrichment in the biomedical domain.

4.2 GOMET: Extending GOMMA Mappings

Semantic mapping as such and especially within the biomedical domain is rarely realized [SHB+09, SE13]. As describe above, STROMA is an enrichment tool, i.e., it takes an already generated mapping as input and enriches each correspondence with an appropriate semantic type information. Main purpose of this thesis is the development of a tool, called GOMET, which takes a GOMMA mapping as input and returns the by STROMA enriched mapping.

Figure 4.5 shows the UML model of GOMET. Only public methods and constructors are depicted. Further mapping types (instead of GOMMA) may be added by implementing a further specialisation of Parser and an own ReparsingPointX.

First, the class GometProcess manages the overall workflow. Three files are expected as being given (their location is specified within a properties file), namely the original mapping file which is the output of GOMMA and contains the correspondences as well as an ID resolution file for source and target ontology, respectively. These latter files are necessary since the mapping file encodes concepts by means of IDs and hence, the ID resolution files resolve each ID to the name of the concept plus its parent and grandparent concept (if existing). The ID resolution files are used to initialize the ID resolver, represented by class IDResolver. After initializing StromaInputGenerator the original mapping is parsed and transformed to the STROMA input format. Now, STROMA enriches the given correspondences. Finally, the semantic type information is added to each correspondence of the original mapping (ReparsingPointGomma) and a copy of the original mapping with this

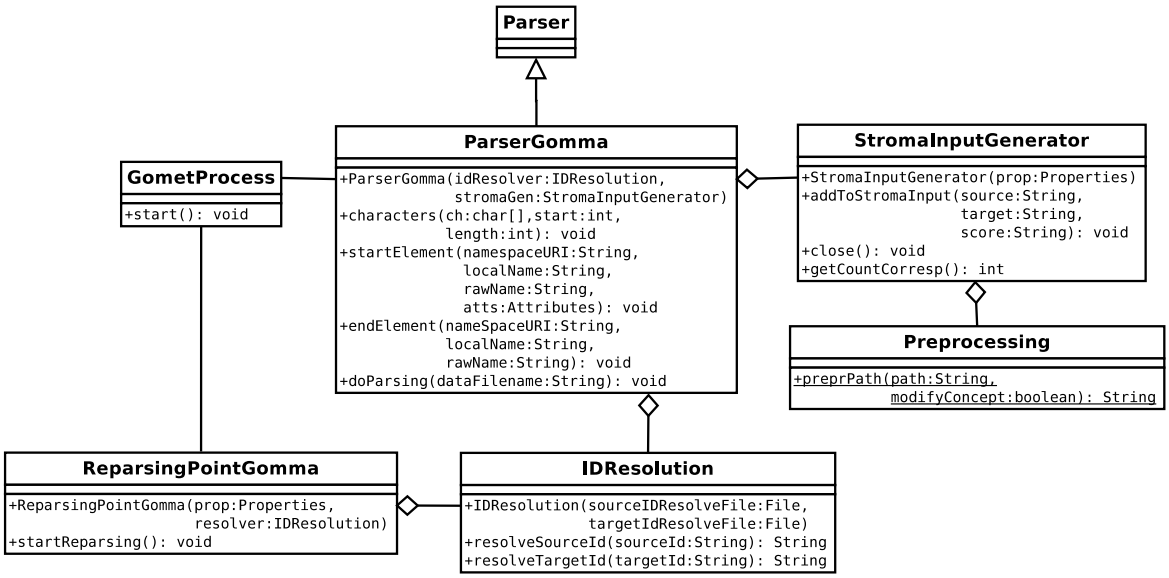

Figure 4.5: Structure of GOMET. Depicted are only public methods.

information is stored as the enriched mapping. The overall workflow of GOMET is shown in figure 4.6. The input and output format of STROMA data stands for the particular files.
Second, the IDResolution class loads the ID resolution files to working memory. Each line of the files must be in the format *ID* <tabulator> *concept (sub-)path*. Furthermore, this class provides two methods which return the associated concept path to a given ID of the source and target ontology, respectively.
Third, each correspondence, which is extracted from the original mapping file via parsing, has to

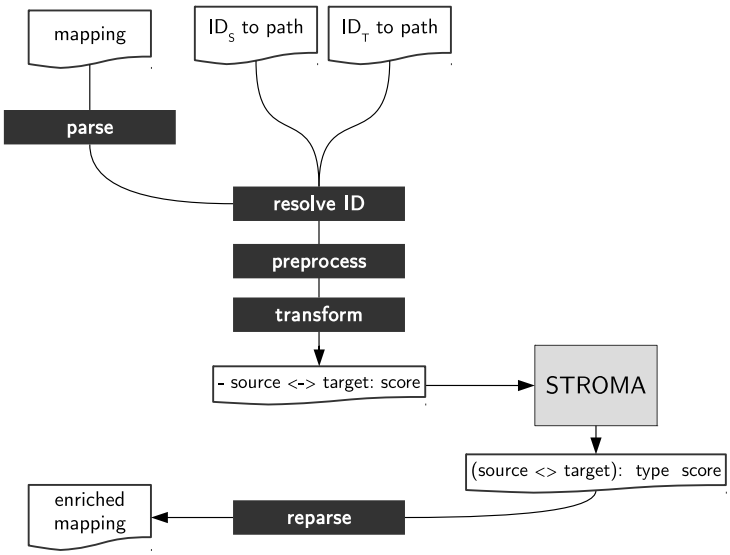

Figure 4.6: The workflow of GOMET. Depicted are input and output files as well as intermediate files of STROMA.

be represented within the STROMA input format, namely - *source path <-> target path: score*. This is realized by StromaInputGenerator, which is initialized with the above mentioned property file where it is specified whether preprocessing of source and target path should take place, if so whether preprocessing should be logged, and finally where to store the generated STROMA input file. The method addToStromaInput(.,.,.) adds a correspondence consisting of a source concept (called *subject*), a target concept (called *object*), and the score of this correspondence (i.e., the similarity value) to said file. close() tells this class all correspondences within the original mapping file are parsed and getCountCorresp() returns the number of correspondences which are added to the STROMA input file.
Fourth, the class Preprocessing provides a set of methods which are applied in order to normalize the given data for STROMA. It is a more heuristic aspect of GOMET and can be activated or deactivated on the source/ target path level but not on the level of the whole correspondence. The former level consists of the following preprocessing steps (obligatory):

'_'to' ' Replacing every occurrence of underscore ('_') by a blank character (' ').

LOW Converting all characters to lower case.

'/'to'.' Changing the node delimiter in the paths from '/' to '.'.

The concept level contains then following preprocessing steps. Note that these are the actual heuristic component and are optional:

GEN Leave out the apostrophe in Anglo-Saxon genitives, like *huntington's disease* is normalized to *huntington s disease*.[3]

XofY As *of*-construction are a challenge for STROMA it seems a good approach to normalize constructions like *X of (the) Y* to *Y X*, for instance *digit of the hand* to *hand digit*.

ROM It has been often observed in the data that Roman numbers as a numeration of anatomical parts can be deleted as they do not restrict the meaning. For example, in the correspondence - *nerve.cranial nerve.trochlear iv nerve <-> nerve.cranial nerve.trochlear nerve: 0.8108108* the Roman number 'IV' in the source concept denotes the fact that the trochlear nerve is the fourth cranial nerve, in other words, the fourth cranial nerve is called *trochlear nerve*. Consequently, the given concepts interrelate by *equal*. Deleting the Roman number simplifies the computation.

Fifth, the class ParserGomma is an extension of the abstract Parser class. ParserGomma handles the parsing of the original mapping which is given as GOMMA output. The parsing is done by using a SAX parser[4]. After successfully parsing a correspondence the ID of source and target concept is resolved and then the correspondence is added to the STROMA input file.
Sixth, ReparsingPointGomma adds the semantic type for each correspondence as the value of the attribute *corr_type* to the GOMMA mapping file. *corr_type* is *undecided* if STROMA could not calculate a type. *corr_type* is *null* if the current correspondence is not part of the STROMA output file, i.e., for instance, STROMA deleted this correspondence due to low confidence.

[3]The genitive treatment should be expanded by different possibilities, e.g. deleting the whole genitive marker, *huntington disease*, depending on the standards in the source and target ontology. This is motivated by the fact that STROMA returns *equal* if two concept names are identical.
[4]http://www.saxproject.org/

5 Evaluation and Discussion

This chapter presents an evaluation of the enrichment process of GOMMA mappings in the biomedical domain. At first, the parameters are described which span the space for the evaluation. The second part of this chapter shows the results of testing GOMET. Finally, benefits and problems of a semantic enrichment are discussed.

5.1 Parametrizing the Evaluation

This section consists of two parts. In subsection 5.1.1 the independent and dependent variables of the experiment are introduced. A short overview to the statistical evaluation measures is given in subsection 5.1.2.

5.1.1 Independent and Dependent Variables

Independent Variables The evaluation is based on six independent variables. One, as it is a parameter of GOMET, has already been introduced in section 4.2, namely that preprocessing can be activated or deactivated (prepr). Two other variables are characteristics of SemRep: the maximal path length maxPath and the measurement of node degree degree. The former one defines an upper limit of edges (relations) between two concepts in order to determine their relationship. For instance, a maximal path length of three allows only two intermediate concepts, i.e. three relations, between two given concepts. All paths with more than three edges are not considered for detection of the semantic relation type. The latter one, the node degree heuristic, specifies whether a node n_1 with more outgoing edges than a node n_2 should be interpreted as being more specific than n_2. It is used whenever there is no path between the given nodes and returns an *is-a* relation.[1] The fourth and fifth independent variables are settings of STROMA: (de-)activate *undecided-as-default* (undec), see subsection 3.2.2, and changing the weights of the strategies for semantic type detection (weightBK). The standard setting of the weights is depicted in table 3.2. Besides that setting, high and low weight of background knowledge are tested.[2] Finally, the amount of background knowledge is alternated (identCUIs). This is achieve by parametrizing whether the enrichment takes place including *identical_CUIs* relations, or not. The independent variables and their values are depicted in table 5.1. 'on'/ 'off' may be abbreviated as '✓'/ '–'. As there are five variables with two possible values and one with three values, there are $2^5 \cdot 3 = 96$ experimental conditions.

The motivation for these variables is the following. The utility of prepr is obvious: It is assumed that a reasonable preprocessing influences the results positively as the ontologies differ in less unimportant aspects like case sensitivity or printing of "gaps" between words as blank character or

[1] Actually, the node degree heuristic is not a boolean variable but declares a factor ϕ such that $\langle n_1, \textit{is-a}, n_2 \rangle$ if $g(n_1) \geq \phi \cdot g(n_2)$ where $g(x)$ return the number of outgoing edges for node x.

[2] high weight: background knowledge of weight 0.9, all others of 0.2; low weight: all like standard, except background knowledge which has weight 0.1.

independent variable	abbreviation	values
preprocessing	prepr	{on, off}
undecided-as-default	undec	{on, off}
maximal path length	maxPath	$\{2, 3\}$
node degree heuristic	degree	{on, off}
identical_CUIs are used	identCUIs	{on, off}
weight of background knowledge	weightBK	{s(tandard), h(igh), l(ow)}

Table 5.1: Independent variables of the evaluation.

underscore. Activating undec seems to be an improvement of the mapping as well since it should be a good guess to specify a relation as *undecided* if no information is available. There are two aspects for determining the value of maxPath. On the one hand, the longer the maximal path is allowed to be, the greater the probability that a relation between two concepts is detected. But on the other hand, the longer the maximal path is allowed to be, the greater the probability that an incorrect relation is determined. Furthermore, mostly "short" relations are preferred above "longer" ones, which (combined with the fact that the search space grows exponentially with the maximal path length) leads to only two reasonable settings, namely maxPath $\in \{2, 3\}$. A variable with an unknown influence on the results is degree, i.e., is the node degree heuristic, which returns *is-a* if applicable, a good approach within the biomedical domain? The last two parameters concern the role of background knowledge: How much background knowledge is needed for the best results, and which weighting is optimal? It seems reasonable that background knowledge improves the enrichment. To toggle only between activating and deactivating *identical_CUIs*, is exemplary for this issue. The question is how much weight STROMA should attribute to the background knowledge comparing to the other (e.g. linguistic) strategies of type detection.

Dependent Variables Seven dependent variables are measured. The first dependent variable is the duration (in seconds) of executing the type detection for all correspondences by STROMA, which is the most time-consuming part for GOMET. Further variables ground on the distinction between non-trivial types (all types but *equal*) and the trivial type (*equal*). For both types recall, precision, and f-measure are calculated. Their definition is given in definition 5.1, see [MKSW99]. In the following, trivial precision, recall, and f-measure are abbreviated tP, tR, tF, respectively. Their non-trivial variants are shortened to ntP, ntR, ntF. The term *reference* means a (manually created) set which contains correspondences and their real relation type. This is the set of all true non-trivial or trivial correspondences between two ontologies. Such a reference is called gold standard. However, in some cases the design of such a gold standard is very expensive as it can only be done by experts. A solution might be a silver standard which is based on the mapping and assigns the relation type only to correspondences of this mapping. Hence, d is always zero. However, silver standards might lead to a overestimated recall when the mapping is executed and evaluated to it. The term *hypothesis* refers to the set of correspondences which are assigned a trivial or non-trivial type.

Definition 5.1. Let n be the total number of correspondences within the reference mapping, m the total number of correspondences in the hypothesis, c the number of correspondences with a correct assigned relation type, s the number of correspondences with an incorrect assigned relation type, d the number of (deleted) correspondences that are contained by the reference mapping but not by the hypothesis, and i the number of (inserted) correspondences which are contained by the hypothesis but not by the reference mapping. Then the following holds:

$$\text{Precision } P = \frac{c}{m} = \frac{c}{c+s+i} \tag{5.1}$$

$$\text{Recall } R = \frac{c}{n} = \frac{c}{c+s+d} \tag{5.2}$$

$$\text{F-measure } F = \frac{2PR}{P+R} \tag{5.3}$$

[MKSW99]

5.1.2 Evaluation Measures

All dependent variables are interval scaled. For each of them the arithmetic mean as a measure of central tendency and the standard deviation as a measure of variability are calculated. Their definitions are given in definition 5.2 according to [BS10, 25, 30f]. The arithmetic mean (often only *mean*) characterizes the center of a distribution. The standard deviation states how different the values within a distribution are. It denotes a representative deviation from the center.

Definition 5.2. Let X be an attribute and n be its number of values. The arithmetic mean $\bar{x}$ as well as the standard deviation s of the distribution of X are defined as follows:

$$\bar{x} = \frac{\Sigma_{i=1}^{n} x_i}{n} \tag{5.4}$$

$$s = \sqrt{\frac{\Sigma_{i=1}^{n} (x_i - \bar{x})^2}{n-1}} \tag{5.5}$$

[BS10]

If a variable X is normally distributed, the interval $[\bar{x} - s; \bar{x} + s]$ contains approximately 68% of all values of X [BS10, 70ff]. Approximately 95% of all values are located within the interval $[\bar{x} - 2s; \bar{x} + 2s]$. Based on a short analysis of the distribution of the dependent variables, the assumption of normally distributed dependent variables has been shown to be valid. Thus, "best" results regarding a dependent variable V_{dep} are defined as all results whose value of V_{dep} is greater than the mean plus one standard deviation of this distribution.

As the effect of the independent variables on the dependent ones are evaluated, the correlation coefficient between these is pairwise computed. A correlation coefficient r describes the linear correlation between two variables. r is element of the interval $[-1; 1]$ whereby $r = 0$ denotes no correlation, $r = -1$ denotes a perfect negative correlation, and $r = 1$ describes a perfect positive correlation [BS10, 157]. For instance, age and the achieved score of students in a test are gathered. A correlation r of 0.9 would be interpreted that the older a student, the better his results. $r = -0.9$

would mean that means that the younger the student, the better his results. Furthermore, when r was 0.4 instead of 0.9, one would have to state that the correlation is less obvious, i.e., that the variables *age* and *score* were more loosely connected.
Except weightBK , all independent variables are dichotomous. weightBK represents an ordinally scaled variable. All dependent variables are interval based. The correlation coefficient between dichotomous and interval based variables is computed by the point-biserial correlation [BS10, 171ff], see definition 5.3. It basically divides the dependent variable values into two groups according to the independent variable and compares their means. The correlation coefficient between maxPath and the dependent variables is calculated according to Spearman's rank correlation, see [BS10, 178ff]. Another coefficient is used in this case because maxPath is no dichotomous variable but ordinally scaled. The dependent variables are mapped to a ranking as proposed in [BS10, 174].

Definition 5.3. Let n_{on}, n_{off} be the number of results for the dichotomous independent variable V_{indep} with value 'on' and 'off', respectively. Let $\bar{x}_{\text{on}}$ and $\bar{x}_{\text{off}}$ be the mean of the interval based dependent variable V_{dep} grouped by V_{indep}, and $s_{V_{\text{dep}}}$ be the standard deviation for all values of V_{dep}. It follows that the total sample size n is the sum of n_{on} and n_{off}. Then, the point-biserial correlation coefficient r_{pb} is defined as follows:

$$r_{pb} = \frac{\bar{x}_{\text{on}} - \bar{x}_{\text{off}}}{s_{V_{\text{dep}}}} \cdot \sqrt{\frac{n_{\text{on}} \cdot n_{\text{off}}}{n \cdot (n-1)}} \tag{5.6}$$

[BS10]

Finally, the semantic relation types *is-a*, *inverse is-a*, *part-of*, *has-a*, and *related* are classified as non-trivial ones. A correspondences which is denoted with such a type is called a non-trivial correspondence. Contrarily, a correspondence denoted with *equal* is named a trivial correspondence.

5.2 Testing GOMET

This section deals with three test cases of GOMET. First, in subsection 5.2.1 a manually designed mapping is introduced and evaluated. It contains a lot of non-trivial correspondences and maps MA-to-Wikipedia categories. Second, a real world case from the OAEI[3] is enriched, namely MA-to-NCIt. This mapping contains only less non-trivial correspondences, see subsection 5.2.2. Third, subsection 5.2.3 gives an overview of further mappings while pointing out the difficulty to identify an appropriate test case for GOMET.

5.2.1 A Small but Rich Mapping: MA-to-Wikipedia

The Mapping The source ontology is of a flat type, i.e., it represents a list of 64 MA concepts. These concepts, for example *cardiovascular system endothelium* (MA:0000717), are more specific than the concepts of the target ontology.
The target ontology, an extract of the Wikipedia category tree, consists of 50 concepts which are subconcepts of *Human anatomy*. Note that the category tree is actually not a tree within the graph-theoretic sense as more than one path between two nodes is possible. The Wikipedia subgraph has been manually revised, i.e. some concepts, like *Neckwear*, are deleted as they represent neither

[3] http://oaei.ontologymatching.org/

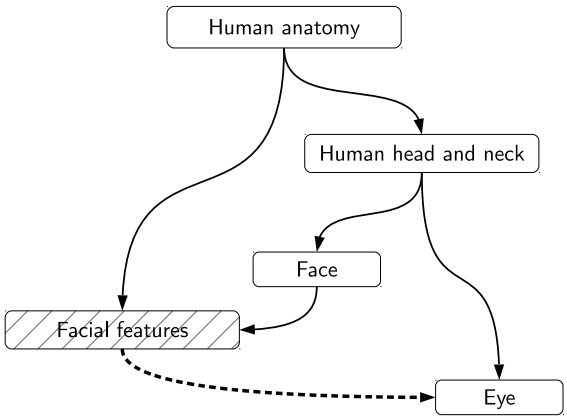

Figure 5.1: Subgraph of the extracted Wikipedia category graph. Arrows pointing from a category to its subcategory. Dotted arrows symbolize *is-a* relations, solid arrows *part-of*.

an *is-a* or *part-of* relation to the superior concept *Human anatomy*. Additionally, concepts are deleted which lead to multiple paths between the root and a concept node. This is shown in figure 5.1 by the concept *Facial features*. The reason for this latter procedure is that STROMA cannot handle multiple paths (multiple inheritance has to be resolved).

As the matching is executed between a highly specific and more general ontology, it is expected that more than one source concept is aligned with the same target concept, like *brainstem* and *hindbrain* are both mapped to *brain*. Hence, a complex mapping is required. This is one reason for the chosen selection strategy MaxDelta [Gro14, 43] [DR02]. Main advantage of the MaxDelta selection is that it allows more than one correspondence to be detected between two concepts, namely all correspondence which have a confidence value ξ greater equal the greatest value for this correspondence minus δ. For $\delta = 0$ all correspondences of a particular concept pair have the same confidence value, which is the maximum. Furthermore, three alignment conditions for the MaxDelta selection are tested: $1:n$, $n:1$, $1:1$. The condition $1:1$ is the most restricted one where a correspondence must contain the best mapping partner for both sides, source and target. For the other parameters a calculated match has to be the best match only for one side, source or target. This yields complex mappings.

In addition, GOMMA creates mappings for a varying (overall) confidence value θ, i.e., correspondences with a confidence value less than θ are excluded. Figure 5.2 shows the size of the mappings depending on θ and parameters for MaxDelta. For $\theta \geq 0.8$ all mappings share the same eight correspondences with $\xi = 1.0$. All of these correspondences relate identical concept names, except $\langle$*cardiovascular system*, *circulatory system*$\rangle$. Considering the overlap between correspondence sets of a particular threshold group, it is mentionable that for $\theta = 0.2$ in alignment condition $1:n$ about 33% of its correspondences are unique which means that they are not detected by one of the others; and 30% are unique for $n:1$. For $\theta = 0.4$, approximately 28% (11%, respectively) of the detected correspondences are unique within $1:n$ ($n:1$, respectively). In $1:1$ there are no unique correspondences in any group.

The creation of mappings varying in threshold θ aims to figure out how many reasonable non-trivial correspondences can be detected with small θ. The idea behind this method is that non-trivial correspondences are more likely to occur in mappings with small θ since concepts of such cor-

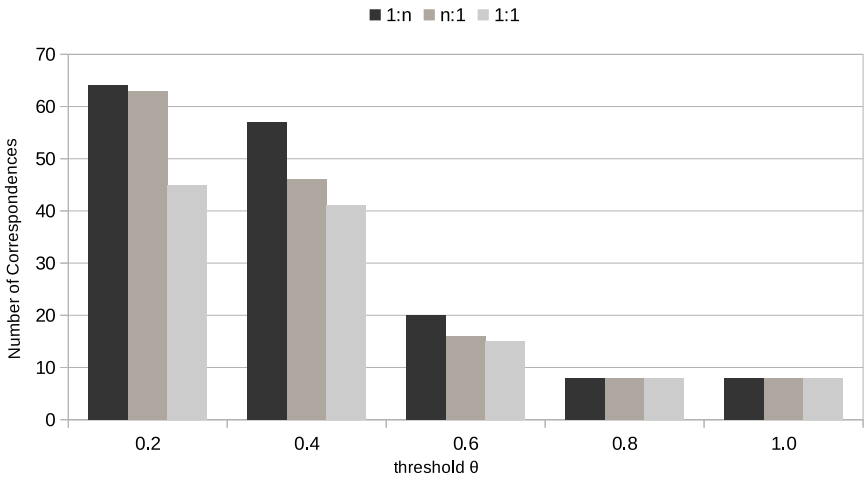

Figure 5.2: Number of correspondences for different MaxDelta strategies ($\delta = 0.05$) for MA-to-Wikipedia. The values are grouped by different thresholds.

respondences have more non-identical strings and therefore these correspondences have a lower confidence value than trivial ones. Nevertheless, the lower θ the more unreasonable and invalid concept alignments are represented within the mapping. A powerful STROMA would then declare the type of this invalid correspondences as *undecided*, and assign a true relation type to the correspondences which are actual ones.

Duration The maxPath influences the execution time of STROMA. For a maximal path length of 2 STROMA calculates the relation types within approximately one second. A maximal path length of 3 leads to a mean execuation of four seconds.

Standard Setting As a baseline of the evaluation the standard settings of STROMA are presumed, i.e., undec=on, maxPath=2, degree=on, identCUIs=on, weightBK=s. Note that there is no preprocessing since the optional preprocessing rules apply for no concept name in this case. For this baseline, each mapping which results from different $\theta \in \{0.2, 0.4, 0.6, 0.8, 1.0\}$ and different MaxDelta strategies (with $\delta = 0.05$) is evaluated. In all 15 cases[4] all trivial correspondences are denoted (tR: 100.0) and the precision tP increases with θ which means that less non-trivial correspondences are erroneously assigned *equal* ($73.3 \leq tP \leq 100$). For $\theta = 0.8$ and greater, only trivial correspondences appear in the mapping which are the same for each MaxDelta strategy, see figure 5.2. Table 5.2 shows the results of the $1:n$ mapping for different thresholds. Effective measures take as population those correspondences which are detected by GOMMA for a specific θ. Overall measures define the population as the set of all possible correspondences (hence including those which are not part of the mapping generated by GOMMA). The mapping size is depicted as 'count'. The results are similar for $1:1$ and $n:1$. For $\theta = 0.4$ the effective values (i.e. the values with respect to mapping generated by GOMMA) are at the lowest rate. The reason might be that at this point, a significant set of valid correspondences are filtered out while wrongly-typed correspondences are still part of the mapping. The last set of correspondence is reduced if θ increases

[4]Five values for θ, three MaxDelta strategies and exactly one setting (the standard setting) of the independent variables lead to 15 cases.

	recall		precision		f		
θ	effective	overall	effective	overall	effective	overall	count
0.2	50.0	43.1	55.0	45.8	52.4	44.4	64
0.4	47.5	37.2	52.8	46.3	50.0	41.2	57
0.6	60.0	11.7	75.0	75.0	66.7	20.2	20

Table 5.2: Recall, precision, and f-measure as well as the size of the mapping for non-trivial correspondences for $1:n$ and different thresholds θ.

to 0.6. However, the overall values, especially the recall, show that the greater θ, the more valid correspondence are a priori filtered out. Thus, it seems to be the best approach to look at the set of correspondences for $\theta = 0.2$ and determine how the independent variables have to be set in order to gain higher recall and precision.

Optimization In the following, independent variable values are changed with respect to the above defined baseline, see figure 5.3. It can be concluded that undec and identCUIs only affect the trivial recall and precision. Deactivating undec and enriching the mapping with the additional knowledge due to identical UMLS CUIs leads to more valid types which are calculated for the correspondences. Contrarily, weightBK only influences the results for non-trivial correspondences. For a high weight the results become better. maxPath=3 has a positive impact on all independent variables. Deactivating the node degree heuristic has a complex influence. It significantly decreases the trivial precision but increases the non-trivial one. The reason is that there are less correspondences which are assigned an *is-a* relation due to different node degrees of the concepts. Since no path between the concepts exists in SemRep, those relations are now returned as *undecided* and as undec is activated in STROMA, they are denoted as *equal* relations. However, the correspondences are wrongly-typed *equal* relations and thus the precision of the trivial correspondences decreases.
As a consequence of these observations the best results might be expected for undec=off, maxPath=3, degree=off, identCUIs=on, weightBK=h. These settings results in a very good enrichment with the following values:[5]

trivial recall = 100.0	non-trivial recall = 70.7
trivial precision = 92.3	non-trivial precision = 80.6
trivial f-measure = 96.0	non-trivial f-measure = 75.3

In particular, more non-trivial correspondences with a higher precision are denoted as it can be seen in figure 5.4. But also the relation type detection of trivial types is improved.
What about the hypothesis that all correspondences with relation type *undecided* should be interpreted as wrong correspondences? On the one side, there are 11 correspondences in the mapping which are wrong correspondences, like $\langle$*hand joint*, *head and neck joints*$\rangle$. Only 3 of these correspondences are typed as *undecided*. On the other side, there are 7 correspondence typed as *undecided*. Besides the 3 wrong ones, all other correspondences are expected to be *part-of* rela-

[5]Similar results are gained for $1:1$ (tR=100, tP=91.7, tF=95.7, ntR=72.4, ntP=84.0, ntF=77.8) and $n:1$ (tR=100, tP=91.7, tF=95.7, ntR=69.7, ntP=82.1, ntF=75.4) mapping with $\theta = 0.2$.

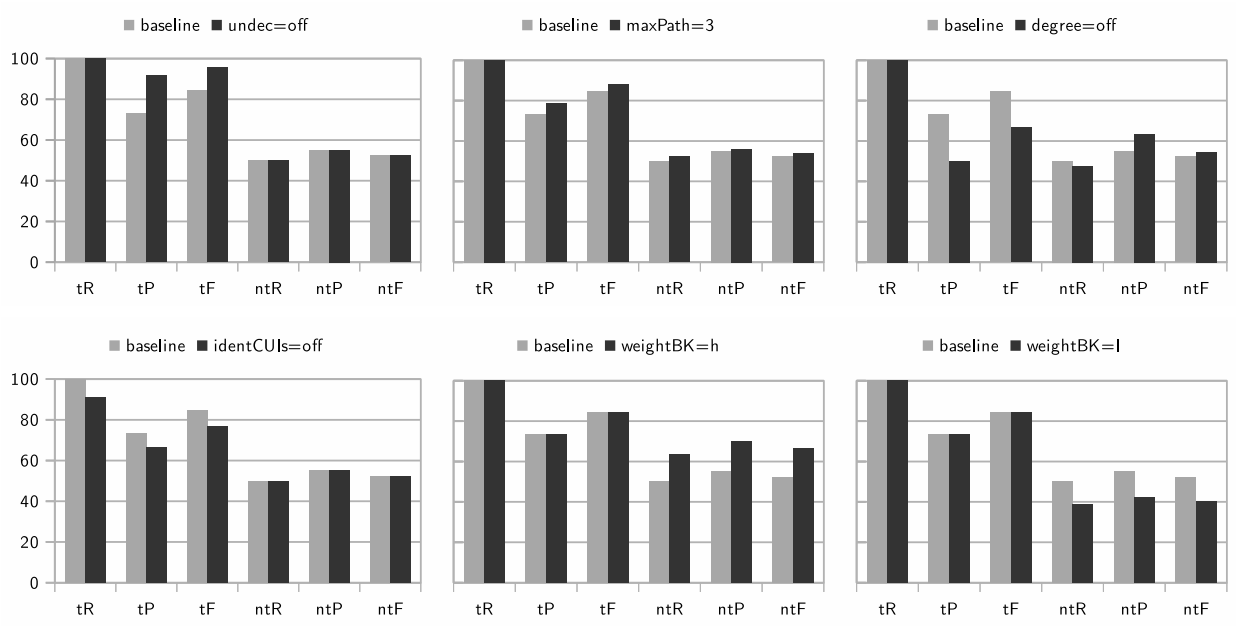

Figure 5.3: One-factorial optimization of the baseline. The baseline results, depicted in light gray, can be improved or worsen when changing exactly one parameter (dark gray bars).

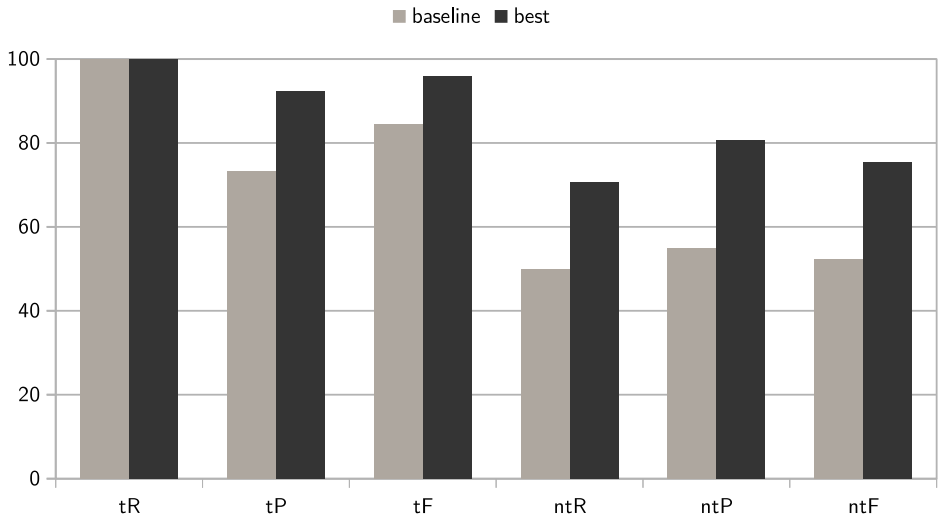

Figure 5.4: Measure for MA-to-Wikipedia mapping ($\theta = 0.2$, $1 : n$). The baseline results are depicted in light gray, the optimal setting, called "best", as dark gray bars.

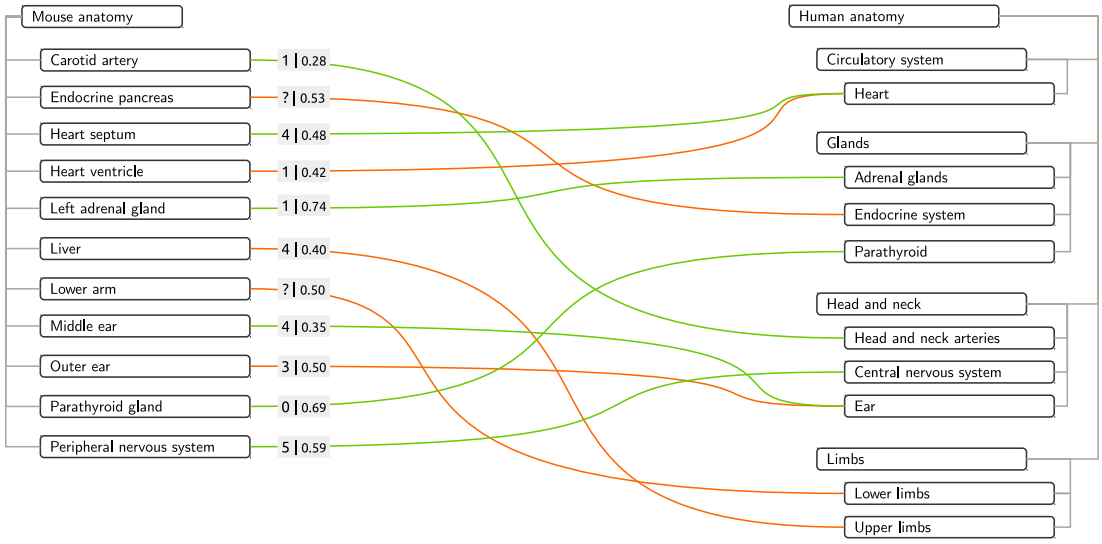

Figure 5.5: Enriched mapping of the correspondence subset from MA-to-Wikipedia. Green lines represent correct correspondences, orange lines represent wrong or falsely-typed correspondences. The first number in the gray boxes denotes the relation type (a question mark stands for *undecided*) and the second number the confidence value. The selection does not reflect the proportions in the whole correspondence set.

tions. Consequently, no generalisation can be made about *undecided* correspondences. They might be correct or they might be wrong correspondences.

Finally, looking at the confidence value of all falsely and correctly-typed correspondences, respectively, yields a "low" significance difference. The confidence value mean of the falsely type correspondences amounts to 0.46 (standard deviation: 0.12). The mean for the correctly assigned correspondence values is 0.63 (standard deviation: 0.21). Nevertheless, the difference is named "low" as the maximum of the confidence values for all wrongly-typed correspondences is quite high ($\xi_{max} = 0.76$), and the minimum of all confidence values for the correctly-typed correspondences is quite low ($\xi'_{min} = 0.24$). Consequently, the confidence value is no exact measurement for wrongly and correctly-typed correspondences as well.

Finally, a subset of the enriched correspondence set is given in figure 5.5. Not that this subset is not representative for the whole set. Rather the correspondences are selected in order to view multifaceted correspondences. For example, there are two *undecided* correspondences. The first one, $\langle$*Endocrine pancreas*, *Endocrine system*$\rangle$ is a correct correspondence. But the second one is not a true one as there is no semantic relation between *lower arm* and *lower limbs*. Furthermore, $\langle$*carotid artery*, *head and neck arteries*$\rangle$ is a correspondence with a very low ξ but nevertheless it is a true correspondence. Additionally, the alignment of *Liver* and *Upper limbs* is wrong, there is no relation at all. However, mapping *Outer ear* to *Ear* is correct but the calculated typed (*has-a*) is wrong. All in all, it can be seen that one concept of the target domain may be aligned to two (or more) concepts in the source domain. The inverse case occurred only once (not depicted) when *Liver* is mapped to *Upper limbs* as well as to *Lower limbs* (both correspondences are wrong).

5.2.2 A Real World Case: MA-to-NCIt

The Mapping This test case is taken from the OAEI 2013[6] and enriches an alignment between MA and the human anatomy part of NCIt. The source ontology (MA) consists of 2,737 concepts, the target ontology (part of the NCIt) of 3,306 concepts. With strong settings regarding the acceptance of a correspondence ($1:1$, $\theta = 0.8$, MaxDelta with $\delta = 0$), GOMMA detected 1,264 correspondences. These are the input of GOMET and STROMA, varying the independent variables as outlined in section 5.1. The calculation of precision, recall, and f-measure is based on a silver standard, as the manual creation of a gold standard without expert knowledge would have been a difficult undertaking.[7] Furthermore, the evaluation is focused on GOMET and not on GOMMA which means that a reliable mapping is presumed to be present.
Although the mapping is imbalanced concerning the distribution of trivial and non-trivial relation types (there are only 21 non-trivial correspondence in the silver standard), the evaluation took place on this mapping in order to see how STROMA copes with such a huge mapping of the biomedical domain (each trivial correspondence has to be assigned *equal* independent of the number of non-trivial relations). Subsequently, the settings are relaxed and samples are taken to see whether the results for the strong settings can be generalised. Thus, the evaluation of the real world case starts with the presentation of the results for the strong setting.

Duration The average duration of processing the correspondence set with STROMA amounts 35.96 s (standard deviation: 23.25 s) over all conditions. There is a strong positive correlation of 0.94 with maxPath. As expected the processing with maxPath = 2 is faster (mean: 14.20 s, deviation: 1.07 s) than with maxPath = 3 (mean: 57.71 s, deviation: 11.16 s).

Mean and Correlations The average recall, precision, and f-measure are given in table 5.3. Trivial relations types are denoted mostly correctly. More than one half of all non-trivial types are detected. However, a non-trivial relation type is assigned to a correspondence with a low precision. As will be seen later, the results are much better in the optimal condition, i.e., the best setting for the independent variables.
The correlations between independent and dependent variables are shown in table 5.4. A positive correlation between a dependent and a dichotomous variable means that activating the dichotomous one improves the result for the dependent one. Additionally, setting the dichotomous variable to 'off' increases the value of the dependent variable, if a negative correlation is notated in the table. The results show that a high value for background knowledge positively influences the results on the trivial as well as on the non-trivial side. An unexpected result is that the non-trivial

	recall	precision	f
non-trivial	55.37 (9.36)	14.02 (3.85)	22.08 (5.07)
trivial	91.14 (4.12)	99.63 (0.48)	95.16 (2.20)

Table 5.3: Average recall, precision, and f-measure for non-trivial and trivial correspondence types from the MA-to-NCIt mapping. A number in brackets stands for the standard deviation.

[6] http://oaei.ontologymatching.org/2013/anatomy/index.html
[7] The same holds for the manually assignment of relation type to each correspondence. Hence, the matching task is executed with those strong settings in order to hold the correspondence set small.

	prepr	undec	maxPath	degree	identCUIs	weightBK
tR	↑	⇈	–	–	↑	↑
tP	–	↓	–	–	↓	–
tF	↑	⇈	–	–	↑	–
ntR	–	–	–	–	⇊	↑
ntP	–	–	–	⇊	–	⇈
ntF	–	–	–	⇊	–	⇈

Table 5.4: Correlation matrix for the MA-to-NCIt mapping. Single arrow: weak correlation, double arrow: strong correlation. Upward arrow: positive correlation, downward arrow: negative correlation.

recall decreases if further background knowledge, namely the *identical_CUI* set, is integrated to SemRep. The same holds for the trivial precision. But further background knowledge positively affects the trivial recall. undec only influences the results of trivial relations since this parameter, if activated, changes each *undecided* to *equal*. As a result of this, it increases the number of denoted *equal* correspondences (recall), but simultaneously converts actual non-trivial *undecided* relations to *equal* and hence decreases the precision. prepr as another independent variable which only affects the trivial side leads to a better recall (if activated). This is the motivation of the preprocessing step: normalising the concept names such that only conventional differences, like gaps between words as underscore or as blank character, disappear. Contrarily, the node degree heuristic only influences the non-trivial precision because it returns *is-a*, if no relation type could be determined by SemRep and both concepts are part of SemRep but with a different number of outgoing nodes. Since the deactivation of degree results in a higher precision, this heuristic seems to be wrong for the biomedical domain. Finally, maxPath does not correlate with any precision or recall.

Best Results There are 22 cases in which the non-trivial f-measure is greater than the mean plus one standard deviation. In 14 of these cases preprocessing, a maximal path length of 2, and a height weight of background knowledge are used (independently of each other). In all of the 22 best cases, degree is off. One half uses the *undecided-as-default* strategy. In most cases (14 of 22) the additional knowledge from *identical_CUIs* is deactivated. A low weight of background knowledge never appears for these best cases regarding non-trivial f-measure.
In 16 cases the trivial f-measure is greater than its mean plus one standard deviation. These cases yield similar results like those described above – with three exceptions: In all cases undec is activated, and there are significantly more cases where i) the node degree heuristic and ii) *identical_CUI* knowledge is used.
The best test cases regarding trivial and non-trivial f-measure share exactly seven conditions. Only one of them (condition 2) is one of the best four results in both charts. For illustration, these values are depicted in table 5.5. The part above shows the best results concerning ntF, the part below shows the best results concerning tF. The last columns show which is the setting of the independent variables to achieve these results.
To summarize, the following setting of independent variables gain the best results: prepr=on, undec=on, maxPath=2, degree=off, identCUIs=on, weightBK=h.

	non-trivial			independent variables					
id	ntF	ntR	ntP	prepr	undec	maxPath	degree	identCUIs	weightBK
2	31.3	47.6	23.3	✓	✓	2	–	✓	h
72	31.3	47.6	23.3	✓	–	2	–	✓	h
37	31.1	66.7	20.3	✓	✓	2	–	–	h
55	31.1	66.7	20.3	✓	–	2	–	–	h

	trivial			independent variables					
id	tF	tR	tP	prepr	undec	maxPath	degree	identCUIs	weightBK
2	98.3	97.4	99.2	✓	✓	2	–	✓	h
97	98.3	97.9	98.8	✓	✓	2	–	✓	h
5	98.1	96.9	99.3	✓	✓	3	–	✓	h
20	98.1	97	99.2	–	✓	2	–	✓	h

Table 5.5: The best four results regarding non-trivial (above) and trivial (below) f-measure. Column 'id' refers to the evaluation of this mapping within the depicted condition.

Relax the Settings As explained in the preliminary lines of this test case, the attempt has been made to relax the strong settings which lead to the correspondence set evaluated above. Thus, the threshold θ is reduced to 0.6. This yields a set $S_{0.6}$ of 163 correspondences.
The determined types by STROMA for the strong and the weakened setting are shown in figure 5.6. In the stronger case approximately 85% of the correspondences are assigned *equal*, 10% are denoted as *undecided*. Other relation types represent only 5%. These circumstances change when looking at the correspondences with a confidence value ξ between 0.6 and 0.8. 62% of the correspondences are returned as *undecided*, 22% as *is-a*, 10% as *inverse is-a* and 6% are assigned to another type. There are no denoted *equal* relations at all. The question arises what are the benefits of such a weakened GOMMA setting. A sample of 36 correspondences is extracted from the STROMA output of $S_{0.6}$ and checked whether the assigned relation types are correct. The result is given in table 5.6. It is shown which STROMA assigned type corresponds to which actual type and how often this is

STROMA type	actual type	number of occurrence	
undecided	–	11	(28%)
undecided	equal	9	(25%)
undecided	*other*	6	(17%)
is-a	is-a	5	(14%)
is-a	equal	3	(8%)
inverse is-a	–	1	(3%)
has-a	has-a	1	(3%)

Table 5.6: Evaluation of the sample from $S_{0.6}$.

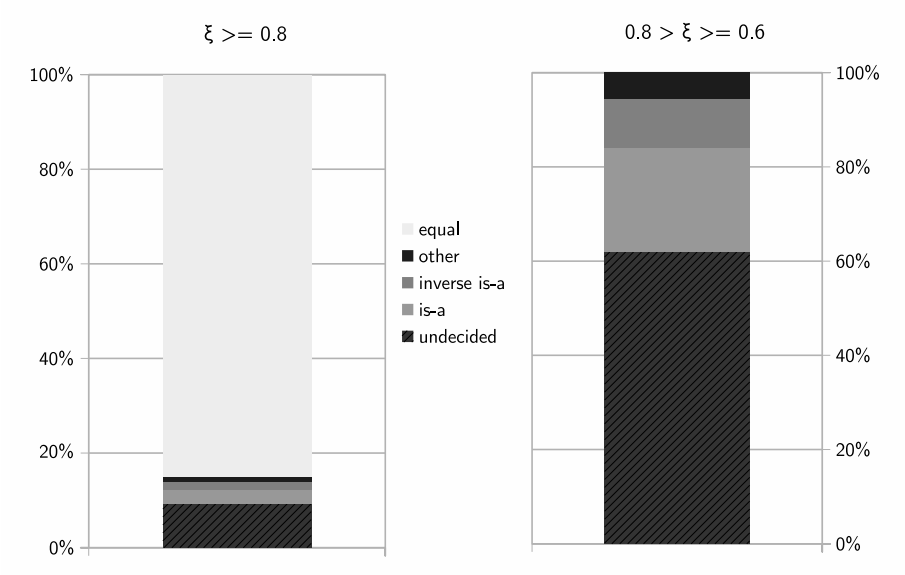

Figure 5.6: Assigned types by STROMA for MA-to-NCIt mapping ($\delta = 0$). Left: denoted types for $\theta = 0.8$. Right: denoted types for ξ between 0.6 and 0.8.

the case. '–' means that there is no relation between the two correspondence concepts at all, thus the correspondence should be deleted. '*other*' subsumes all types except those that are explicitly mentioned for this STROMA type. There 12 wrong correspondences and 6 correspondences with a correct assigned type. All other relations are *undecided* or have a wrongly assigned type. Because one of the most important questions concerns the treatment of the correspondences whose relation type is determined as *undecided*, two possibilities for that treatment are being focussed on. On the one hand, all *undecided* correspondences could be deleted. As a consequence, 16 actual and 11 wrong correspondences are thrown away. Only one wrong and 3 wrongly-typed but 10 correctly-typed correspondences are kept. On the other hand, *undecided* could be interpreted as *equal*. This leads to the case that 12 wrong and 9 wrongly-typed but 15 correctly-typed correspondences are kept. Consequently, the first option is characterised by a better precision but a lower recall than the second one. Such a cautious procedure might be the better way because it yields a smaller but more meaningful data set.

5.2.3 Further Mappings

The OAEI provides further mapping task within the biomedical domain, namely the large biomedical ontology track[8]. It contains mappings between FMA, NCIt, and SNOMED CT. The gold standards for the mappings are already given such that it can easily be seen that there are only a few non-trivial correspondences which are expected after the matching task.[9] An analysis of the reference mapping of FMA to SNOMED, which consists of the greatest ratio of non-trivial correspondences compared to the other mappings, revealed that the non-trivial correspondences are mostly of the same structural type: $\langle X, \textit{is-a}, X\ \textit{structure}\rangle$ or $\langle X, \textit{is-a}, \textit{structure of}\ X\rangle$. Nevertheless, there are also occurrences of correspondences like $\langle X, \textit{equal}, X\ \textit{structure}\rangle$ and $\langle X, \textit{equal}, \textit{structure of}\ X\rangle$.

[8] http://oaei.ontologymatching.org/2013/largebio/index.html

[9] FMA-to-NCIt contains 1.4% non-trivial correspondences, SNOMED CT to NCIt 3.0%, and FMA to SNOMED CT 7.5% in the reference mapping. The specific type (*is-a* vs. *has-a*) is not given.

Similar to *structure* are *set* and *group*. Testing STROMA for these mappings revealed that such constructions are very hard to handle for STROMA. The reason is that the concept names can become large, as *Structure of deep branch of ulnar nerve*, contain one or more *of*'s, which are tricky for STROMA, and a preprocessing like inverse *structure of X* by omitting *of* and deleting *structure* fails. Such a preprocessing destroys the clue for determining the relation type between the concepts as the relation type seems to be arbitrary. Other reference mappings show analogue problems.
These problems (complex constructions and only a few non-trivial relation types) lead to the assessment that none of those mappings constitutes a good test case for GOMET. All in all, the requirements for a reasonable test mapping for GOMET are very strict since they are the combination of the requirements for an appropriate mapping for GOMMA as well as for STROMA. This means that a given mapping M should consists of ontologies where a correspondence between two concepts can be identified via string-based techniques. Hence, the concept labels under consideration (or their given synonyms within one ontology) should be similar. Furthermore, such corresponding concepts should be not only synonyms but should also be related via non-trivial relations. However, STROMA should be able to enrich the correspondences, i.e. the concept labels have to be more simple than complex and the concepts should be part of SemRep. Besides these challenges, the evaluation requires a gold or silver standard. In the optimal way such a standard is a prior given. Its manual design would take much time as a specialist knowledge in the biomedical domain is required.

5.3 Focussing Benefits and Problems

Aim of this thesis is to establish a connection between GOMMA, a tool which generates ontology mappings within the biomedical domain, and STROMA, a tool which enriches a given mapping concerning the relation type of each single correspondence. This connection is programmatically realized by GOMET. Theoretical issues which should be discussed in the context of that connection focus on the prerequisites of a valid type detection. In the previous section the experimental results are presented. This section poses an interpretation of these results.
First, the question arises whether a general pattern can be identified which leads to a wrong type detection. This can be affirmed. For example, there are "universal concepts", like they are discussed by [Arn15], which interfere the type computation. Such a concept is for example *thing*. Especially in the case of a biomedical mapping, *part* constitutes also a universal concept. For instance, as *thorax skin* is a *part*, namely of the body, and *thorax* is a *part* of the body as well, SemRep calculates a *related* relation instead of the expected *part-of* relation. Additionally, background knowledge can be misleading. Since the relation $\langle outer\ ear, has\text{-}a, ear\rangle$ has been extracted from WordNet, the relation type *has-a* is returned to STROMA for the concepts *outer ear* and *ear* (confidence value: 0.93). However, the correct relation type (*part-of*) only achieves a confidence value of 0.80 and is ignored. As a consequence, semantic enrichment in the biomedical domain requires a detection of specific universal concepts (*part*, *anatomy*, ...) and potentially the exclusion of very common knowledge sources. It is an interesting starting point for future work to analyze knowledge sources for their suitability within a specific domain. WordNet, for example, may contain "sloppy" relations.
Another issue which leads to errors by the type detection goes back to the pragmatical usage of language. When somebody is talking about the heart and says that he has seen a human

with an unusual small chamber last week, it is clear that in this context *chamber* refers to the heart chamber; thus, ⟨*chamber*, *equals*, *heart chamber*⟩. Such relations are part of the background knowledge. Since UMLS contains the equality of *urinary bladder* and *bladder*, and WordNet contains the hypernym-relation between *bladder* and *gall bladder* it is returned that *gall bladder* is a *urinary bladder*, ⟨*gall bladder*, *is-a*, *urinary bladder*⟩. Such wrongly-typed correspondences can be excluded by looking at the structure of the involved words. Let X, Y be a string. It might be rarely the case that an X *bladder* is a Y *bladder* if Y is not a substring of X. However, for some cases it might be useful to have such pragmatic relations. Thus, trying to delete them would be a bad approach. Referring to a part of the eye, *lens* and *crystalline lens* are synonyms. In the case of the correspondence ⟨*lens*, *crystalline lens*⟩ the compound as well as the word frequency strategy of STROMA vote for a *inverse is-a* relation. Only a high weight of background knowledge (assuming that the *equal* relation is part of SemRep) can lead to a correctly-typed correspondence. Summarizing, pragmatic relations are necessary and useful when the concepts under consideration are subject and object of such a relations. But as soon as they are involved in a complex inference, they will lead to a wrongly-typed correspondence probably.

Similar to those pragmatic relations are correspondences where one concept name is only a shortened form of the other one, like ⟨*trapezium*, *trapezial bone*⟩ or, more like a modifier-head construction, ⟨*arteriole smooth muscle*, *arteriole smooth muscle tissue*⟩. In such a case the two concepts stand in an *equal* relation to each other. Specific pattern like *muscle* vs. *muscle tissue* may be recognized independently of STROMA such that there is another heuristic which may be useful if at least one of the concepts is not part of SemRep. Considering such heuristic which can be applied even if one concept is not part of the background knowledge, is a reasonable thought as background knowledge always tends to be incomplete.

Another more linguistic heuristic concerns the equivalence of suffixes. The concepts *large intestine muscularis mucosa* and *large intestinal muscularis mucosa* denote the same thing. However, they differ only in a small substring. The second concept label contains *intestinal* instead of *intestine* like the first one. Further (reduced) examples are ⟨*ovary X*, *ovarian X*⟩ and ⟨*X interosseus Y*, *X interosseous Y*⟩. Establishing such a heuristic in STROMA may further improve the results in the absence of these concepts in SemRep.

A last linguistic heuristic which is proposed for future preinvestigations handles permutations. For example, ⟨*eye anterior segment*, *anterior eye segment*⟩ consists of the concept but with slightly different labels: the single words are permuted. In such a case, *equal* should be returned by this method.

Some further heuristics for STROMA were introduced. Their scope has still to be evaluated. However, if it becomes clear that the heuristics are valid, they might be integrated into STROMA. The integration of different strategies is the core of STROMA which makes it as powerful and flexible as it is.

Second, in order to summarize the influence of the independent variables to the quality of the enriched mapping, it can be stated that although maxPath=3 returned better results it is recommended to use this setting only for small mappings. The reason lies in the correlation of maxPath with the execution time of STROMA. As the number of paths which have to be checked in the background knowledge repository exponentially increases with the maximal path length, the execution time increases considerably. Furthermore, the node degree heuristic seems to fail within the biomedical domain. Mainly, that is because of the huge amount of *part-of* relations instead of *is-a* in the domain of interest. All in all, background knowledge seems to be more important

than the other strategies of STROMA. The reason might be that the other strategies are not sensitive enough for the special structure of specialized terminology of the biomedical domain. For example, in modifier-head constructions like *thorax skin* no relation can be determined between compound and modifier in the common language. However, there are hints in the data that this might be different in the biomedical domain. In this domain language a *part-of* relation often occurs, $\langle$*thorax skin*, *part-of*, *thorax*$\rangle$.

Third, regarding to *undecided* typed correspondences, the evaluation has shown the following. In the MA-to-Wiki as well as in the MA-to-NCIt mapping, at least 25% of all *undecided* correspondences are wrong correspondences, i.e., they are erroneously determined by GOMMA. Although not all wrong correspondences are assigned *undecided*, it is reasonable to exclude the *undecided* ones from the mapping. First, this eliminates some wrong correspondences. Second, it is not possible to assign a default type to them because there is no default type at all. If the matching is executed with a low threshold more and more non-trivial correspondences appear in the mapping and thus *undecided* may be a true *is-a* or *part-of*. In MA-to-Wiki, either *undecided* identifies a wrong correspondence or a *part-of* relation.

Fourth, it was decided to test different thresholds. A complete evaluation is only possible for mapping MA-to-Wiki as the MA-to-NCIt lacks an overall reference mapping containing semantic relation type information. Thus, it holds that the higher the threshold, the more correctly-typed correspondences are returned by STROMA, and the number of wrong and wrongly-typed correspondences decreases. The best threshold seems to be $\theta = 0.6$ as for $0.6 \leq \xi < 0.8$ a lot of correspondences are typed correctly (83%) and only a few false correspondences stay in the resulting correspondence set (assuming that *undecided* correspondences are deleted). There are no correspondences with $0.8 \leq \xi < 1.0$. For ξ between 0.4 and 0.6 there are already 35% of wrongly-typed correspondences. Further tests should be run in order to provide more evidence for this threshold account.

Summarizing the results for GOMMA, lowering θ leads to more non-trivial correspondences within the mapping. Although this procedure increases the recall for true correspondences and especially, the recall for true non-trivial ones, the precision decreases since more and more wrong correspondences are determined. It would be a preferable result to state that typically, no relation type, i.e., *undecided*, is assigned to wrong correspondences. But the evaluation shows that the currently used STROMA strategies do not provide such a result. However, $\theta = 0.6$ seems to be the best setting for a mapping which contains many non-trivial types while maintaining a high precision. In this case, *undecided* correspondences have to be deleted. A lower threshold may be tested as soon as STROMA will have been successfully optimized for biomedical data sets.

6 Conclusion

This thesis establishes a connection between GOMMA and STROMA – both are tools of ontology processing. Consequently, a new workflow of denoting a set of correspondences with five semantic relation types has been implemented. Such a rich denotation is scarcely discussed within the literature. The evaluation of the denotation shows that trivial correspondences are easy to recognize (tF $>$ 90). The challenge is the denotation of non-trivial types (30 $<$ ntF $<$ 70).
A prerequisite of the implemented workflow is the extraction of semantic relations between concepts. These relations represent additional background knowledge for the enrichment tool STROMA and are integrated to the repository SemRep which is accessed by this tool. Thus, STROMA is able to calculate a semantic type more precisely. UMLS was chosen as a biomedical knowledge source because it subsumes many different ontologies of this domain and thus, it represents a rich resource. Nevertheless, only a small set of relations met the requirements which are imposed to SemRep relations. Further studies may analyze whether there is an appropriate way to integrate the missing relations as well.
The connection of GOMMA with STROMA allows the semantic enrichment of a biomedical mapping. As a consequence, this thesis enlightens two subjects of research. First, STROMA had been tested with general ontologies, which models common sense knowledge. Within this thesis, STROMA was applied to domain ontologies. Studies have shown that overall, STROMA was able to treat such ontologies as well. However, some strategies for the enrichment process are based on assumption which are misleading in the biomedical domain. Consequently, further strategies are suggested in this thesis which might improve the type denotation. These strategies may lead to an optimization of STROMA for biomedical data sets. A more thorough analysis will review their scope, also beyond the biomedical domain. Second, the established connection may lead to deeper investigations about advantages of semantic enrichment in the biomedical domain as an enriched mapping is returned. Despite heterogeneity of source and target ontology, such a mapping results in an improved interoperability at a finer level of granularity. The utilization of semantically rich correspondences in the biomedical domain is a worthwhile focus for future research.

Bibliography

[AR13] Patrick Arnold and Erhard Rahm. Semantic enrichment of ontology mappings: A linguistic-based approach. In *Proceedings of the 17th East-European Conference on Advances in Databases and Information Systems (ADBIS)*, 2013.

[AR14] Patrick Arnold and Erhard Rahm. Enriching ontology mappings with semantic relations. *Data and Knowledge Engineering*, 93:1–18, 2014.

[AR15] Patrick Arnold and Erhard Rahm. SemRep: A repository for semantic mapping. In *16. Fachtagung "Datenbanksystem für Business, Technologie und Web" (BTW)*, 2015.

[Arn13] Patrick Arnold. Semantic enrichment of ontology mappings: Advances, insights and ideas for improvement. *Talk at Seminar Zingst*, 2013.

[Arn15] Patrick Arnold. Semantic enrichment of ontology mappings. Insights and outlook. *Talk at Seminar Zingst*, 2015.

[Ben06] Michael J. Benton. *Vertebrate palaeontology.* Blackwell, Malden (Mass.), 2006.

[Bor97] Willem N. Borst. *Construction of Engineering Ontologies for Knowledge Sharing and Reuse.* PhD thesis, University of Twente, 1997.

[BS10] Jürgen Bortz and Christof Schuster. *Statistik für Human- und Sozialwissenschaftler.* Springer, Berlin, 2010.

[DH05] AnHai Doan and Alon Y. Halevy. Semantic-integration research in the database community: A brief survey. *AI Magazine*, 26(1):83–94, 2005.

[DR02] Hong-Hai Do and Erhard Rahm. COMA: A system for flexible combination of schema matching approaches. In *Proceedings of the 28th international conference on Very Large Data Bases (VLDB)*, pages 610–621, 2002.

[ES07] Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching.* Springer, Berlin, 2007.

[FH11] Kai von Fintel and Irene Heim. *Intensional Semantics (Lecture Notes).* MIT Spring Edition, 2011.

[GG95] Nicola Guarino and Pierdaniele Giaretta. Ontologies and knowledge bases: Towards a terminological clarification. In Nicolaas Mars, editor, *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, pages 25–32. IOS Press, Amsterdam, 1995.

[GOS09] Nicola Guarino, Daniel Oberle, and Steffen Staab. What is an ontolgy. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, volume 2 of *International Handbooks on Information Systems*, pages 1–17. Springer, 2009.

[GPFLC04] Asunción Gómez-Pérez, Mariano Fernández-López, and Oscar Corcho. *Ontological engineering: With examples from the areas of knowledge management, e-commerce and the Semantic Web*. Springer, London, 2004.

[Gro14] Anika Groß. *Evolution von ontologiebasierten Mappings in den Lebenswissenschaften*. PhD thesis, Universität Leipzig, 2014.

[Gru93] Thomas R. Gruber. A translation approach to portable ontology specification. *Knowledge Acquisition*, 5(2):199–220, 1993.

[Gui09] Giancarlo Guizzardi. The problem of transitivity of part-whole relations in conceptual modeling revisited. In *21st International Conference on Advanced Information Systems Engineering (CAISE)*, Amsterdam, 2009.

[HS06] Hans-Jörg Happel and Stefan Seedorf. Applications of ontologies in software engineering. In *2nd International Workshop on Semantic Web Enabled Software Engineering (SWESE)*, 2006.

[KGHR11] Toralf Kirsten, Anika Groß, Michael Hartung, and Erhard Rahm. GOMMA: A component-based infrastructure for managing and analyzing life science ontologies and their evolutioin. *Journal of Biomedical Semantics*, 2(6):1–24, 2011.

[KSVS08] Johan W. Klüwer, Martin G. Skjæveland, and Magne Valen-Sendstad. ISO 15926 templates and the Semantic Web. In *W3C Workshop on Semantic Web in Energy Industries; Part I: Oil and Gas*, 2008.

[LM01] Ora Lassila and Deborah McGuinness. The role of frame-based representation on the Semantic Web. Technical report, Knowledge System Laboratory. Standford University, Standord (Calif.), 2001.

[MKSW99] John Makhoul, Francis Kubala, Richard Schwartz, and Ralph Weischedel. Performance measures for information extraction. In *Proceedings of DARPA Broadcast News Workshop*, pages 249–252, 1999.

[Obe14] Daniel Oberle. How ontologies benefit enterprise applications. *Semantic Web*, 5(6):473–491, 2014.

[OR89] Charles K. Ogden and Ivor A. Richards. *The meaning of meaning: A study of the influence of language upon thought and the science of symbolism*. Harcourt Brace Jovanovich, Orlando (Fla.), 1989.

[RB01] Erhard Rahm and Philip A. Bernstein. A survey of approaches to automatic schema matching. *VLDB Journal*, 2001.

[SBF98] Rudi Studer, Richard V. Benjamins, and Dieter Fensel. Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25(1-2):161–198, 1998.

[SE13] Pavel Shvaiko and Jérôme Euzenat. Ontology matching: state of the art and future challenges. *IEEE Transactions on Knowledge and Data Engineering*, 25(1):158–176, 2013.

[SHB+09] Jetendr Shamdasani, Tamás Hauer, Peter Bloodsworth, Andrew Branson, Mohammed Odeh, and Richard McClatchey. Semantic matching using the UMLS. In Lora Aroyo and Paolo Traverso, editors, *The Semantic Web: Research and Applications*, pages 203–217. Springer, Berlin, 2009.

[Stu11] Heiner Stuckenschmidt. *Ontologien: Konzepte, Technologien und Anwendungen*. Springer, Berlin, 2011.

[UG96] Mike Uschold and Michael Gruninger. Ontologies: Principles, methods and applications. *The Knowledge Engineering Review*, 11(2):93–136, 1996.

[UG04] Mike Uschold and Michael Gruninger. Ontologies and semantics for seamless connectivity. *Speical Interest Group on Management of Data (SIGMOD) Record*, 33(4):58–64, 2004.

[WC12] A.J. Wiebe and C.W. Chan. Ontology driven software engineering. In *25th IEEE Canadian Conference on Electrical Computer Engineering (CCECE)*, pages 1–4, 2012.

[WDO04] David B. Weishampel, Peter Dodson, and Halszka Osmólka. *Dinosauria*. University of California Press, Berkeley, 2004.

Selbstständigkeitserklärung

Ich versichere, dass ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe, insbesondere sind wörtliche oder sinngemäße Zitate als solche gekennzeichnet. Mir ist bekannt, dass Zuwiderhandlung auch nachträglich zur Aberkennung des Abschlusses führen kann.

Ort	Datum	Unterschrift