Theory Interpretations for Topic Models

Felix Kuhr, Özgür L. Özçep
University of Lübeck
Institute of Information Systems
Ratzeburger Allee 160, 23562 Lübeck
{kuhr,oezcep}@ifis.uni-luebeck.de

Abstract

Many machine learning models have to incorporate latent variables to learn target concepts on training data. The variables are understood only statistically and optimize a statistical property such as likelihood, but usually they are not understood in human understandable semantical terms. An example for such a situation is that of topics in the generative Bayesian model called latent Dirichlet allocation, modelling topics as word distributions from the vocabulary of documents. This paper proposes a framework of classifications and theory interpretations to be used as a construction and analysis tool for exactly such situations. As a proof of concept an algorithm is considered that uses latent Dirichlet allocation topics induced by a corpus to enrich the given sets of RDF annotations on each text of the corpus. The general framework of classifications is used to discuss the role of the algorithm in finding representations of topics by RDF triples.

Introduction

The success of and interest in machine learning (ML) algorithms is reflected in the research of the semantic web community, and it is possible to observe fruitful and quite diverse mutual influences between both communities. The semantic community has profited from ML algorithms in typical tasks of information or document retrieval for the web. However, current ML algorithms—in particular those developed in the area of explainable AI—use ontologies as a means to make the results of an algorithm explainable to humans (Ribeiro, Singh, and Guestrin 2016). A technical aspect associated with the aim of making ML results human understandable is that of *representation*, motivated by the need to make required latent features of a statistical model learnable automatically (Schmidhuber 2014) and also understandable in qualitative semantical terms (Bengio, Courville, and Vincent 2013). Furthermore, ML algorithms use ontologies as hard constraints to semantically enhance statistical models (Deng et al. 2014) resulting in better performance. The main problem in bridging the worlds of ML and RDF or OWL ontologies are two main differences between models usually used in ML and models used in the semantic web. The first is that ML models are usually statistical models, whereas the models handled in RDF(S) are qualitative. Of course, in RDF(S) one can handle also numbers as data types but there is no probability integrated into the semantics. A second difference is that statistical models usually adhere to the closed world assumption (CWA), and many results of ML can be accomplished only by a CWA, e.g., considering independence between random variables in a Bayesian network work. These considerations rest heavily on the CWA. On the other hand, the semantics of RDF repositories adheres to the open world assumption (OWA), according to which the absence of a triple in a repository does not entail the negation of the triple. All the aforementioned diverse approaches, for bridging the quantitative world of ML algorithms based on CWA and the qualitative logico-semantical world of ontologies based on the OWA, exhibit a solution for specific scenarios and usually take the standpoint of one of the two worlds in order to incorporate some aspects of the other world. Thereby, the approaches usually describe how information from a model of one world can be used in the other world—but not vice versa. However, in particular for trending fields like internet of things (IoT) with many distributed entities producing heterogeneous data of both types, quantitative and qualitative, a more general theory of information flow is required: (i) coping with both types of models and, (ii) explaining information flow in both directions. Such a theory has to be built on a simple data structure capturing both, quantitative and qualitative aspects. There has been work around a structure which can take the role of such an abstract data structure. In the theory of distributed logics of (Barwise and Seligman 2008) this data structure is called a *classification*—and we are going to follow this terminology, as we will mainly refer to the general theory of (Barwise and Seligman 2008). The same structure is discussed in different terminology and different disguises as *polarities* (Birkhoff 1973) in lattice theory or as *contexts* in formal concept analysis (Ganter, Franzke, and Wille 2012), or as *Chu spaces* (Barr 2006) in theoretical computer science. The idea is that each model contains entities called *tokens* that can be classified as a type from a given set of *types*. Then, information between the models flows due to some regularities assumed to hold in the models.

Though these approaches are known in research for quite a time, they have not been considered for bridging the ML and the world of ontologies. The observation on which the contributions of this paper rest is that statistical models have emergent qualitative properties that can be described as classifications in the sense mentioned above. Hence, the integration of quantitative and qualitative models aimed at can be accomplished or at least analysed in a theory handling structure preserving mappings between classifications. There are various structures associated with a classification, but the main structure, describing regularities on the classification is that of a theory. And the kind of structure preserving mapping is that of a *theory interpretation*. The classification by itself is a "closed world" but the theory defined over the classification can be discussed w.r.t. other compatible classifications. So we have an integration of models based on the CWA with models based on the OWA.

We analyse a concrete representation task in topic analysis justifying our claim that an approach based on classifications and their structure-preserving mappings presents an adequate framework for constructing mappings between models of ML and the qualitative models associated with RDF repositories. In the setting of this paper we assume that there are annotations of the documents with RDF triples, and we consider a concrete EM-like algorithm that moves the RDF annotations between the documents until some fixed point is reached. We argue that this saturation easies the construction of a structure preserving mapping that maps topics—considered as types in a classification—and RDF annotations—also considered as types in a classification.

The remainder of this paper is structured as follows. We start with a look at related work. Then, we present background information about topic modelling and the algorithm for the mutual enrichment of RDF annotations of documents in a given corpus. We follow with an analysis of the representation problem for topics using the framework of classifications and theory interpretations. Then, we discusses related work and close with some general remarks on the role of our approach and an outlook on future work.

Preliminaries

This section gives a brief overview of latent Dirichlet allocation (LDA), annotation enrichment, and information flow.

Latent Dirichlet Allocation

(Blei, Ng, and Jordan 2003) have introduced the topic modelling technique called LDA which assumes that documents in a corpus $\mathcal{D}$ represent a mixture of topics where each topic is characterized by a distribution of words from a vocabulary $\mathcal{V}$ of words from the documents in $\mathcal{D}$. LDA generates a topic model from the documents in $\mathcal{D}$, learning latent structures of two forms, (i) a *document-topic distribution* θ representing each document $d \in \mathcal{D}$, i.e., the degree which the content of d is about each topic of a set of K topics, and (ii) a *topic-word distribution* ϕ describing the probability of each word from $\mathcal{V}$ occurring in each of the K topics. Both the document-topic distribution and the word-topic distribution depend on the documents in $\mathcal{D}$. The inputs for LDA are a corpus $\mathcal{D}$ of documents as defined above, the number of topics K as well as two hyperparameters α and β, where α conditions the per-document topic distributions θ_d and β conditions the per-corpus topic distributions ϕ_k, $k \in \{1, ..., K\}$. The hyperparameters trade off the following two goals to find groups of tightly co-occurring words: (i) Allocate words of documents to as few topics as possible (α), and (ii) assign high probability to as few terms as possible in each topic (β).

Formally, for each document d in corpus $\mathcal{D}$, LDA learns a discrete probability distribution θ_d that contains for each topic $k \in \{1, \ldots, K\}$ a value between 0 and 1 s.t. the sum of all values is 1. Each word in a document is assumed to come from one of the latent topics with a probability as given by the probability distribution θ_d. LDA also learns a discrete probability distribution ϕ_k for each topic $k \in \{1, \ldots, K\}$ that contains for each word $w \in \mathcal{V}$ a value between 0 and 1 s.t. the sum of all values is 1, too.

Annotation Enrichment of Documents

Annotations provide additional data for documents, supporting humans and machines to handle documents' content. The degree of *added value* of annotations for a document depends on the benefit for applications such as query answering. We consider the iterative algorithm from (Kuhr and Möller 2019), enriching documents with annotations from related documents. We analyse the algorithm in terms of classification in the next section. The itertaive algorithm 1 proceeds by considering i) the composition of documents in a given corpus, ii) the text of all documents in the corpus, and iii) the RDF triples (annotations) in the repositories of documents. The idea is to consider a topic-related *D-similarity* function (Sim_D) and a RDF-related *G-similarity* function (Sim_G) to enrich documents with annotations. Algorithm 1 alternates between an expectation step (E-step) and a maximization step (M-step), estimating for each document $d \in \mathcal{D}$ a subset of annotations t from the RDF repositories of d-related documents having a high expected relevance value for the document d. All d-related documents must have high values for Sim_D and Sim_G with document d, where Sim_D is a metric estimating a textual similarity between two documents by comparing their topic distributions, and is defined by: $Sim_D(d_i, d_j) = 1 - H(\theta_{d_i}, \theta_{d_j})$, and function $H(\theta_{d_i}, \theta_{d_j})$ calculates the Hellinger distance between the topic distributions θ_{d_i} and θ_{d_j}. The smaller the distance between θ_{d_i} and θ_{d_j}, the higher the D-similarity between the documents d_i and d_j. The second similarity measure, called G-similarity, compares the annotations between two RDF repositories with each other, resulting in a similarity value between both repositories. The similarity function $s(r(d_i)^k, r(d_j)^l)$ estimates the similarity between the k-th triple in $r(d_i)$ and the l-th triple in $r(d_j)$ by comparing their subjects, predicates, and objects, resp., with each other. The more similar two triples, the higher their similarity value. Function $f(d, z)$ identifies for RDF repository $r(d)$ the set of repositories $\mathcal{G}^d$ such that the G-similarity between $r(d)$ and all repositories in $\mathcal{G}^d$ is greater than threshold z. The function $f(d, z)$ returns the set of repositories and is defined by: $f(d, z) = \{\mathcal{G}^d \mid Sim_G(r(d), r(d_i)) > z\}_{i=1}^{|\mathcal{D}|}$, For fur-

Algorithm 1 Algorithm for RDF-Triple Enrichment of Texts

Input: $d, r(d), \mathcal{D}, h$
Define: $\epsilon = 0.1$, $\mathcal{D}^d$, $\mathcal{D'}^d$, $\mathcal{G}^{r(d)}$, $r(d)'$
Initialize: $\overline{Sim}_{G^{r(d)}} = \epsilon$, $\overline{Sim'}_{G^{r(d)}} = \overline{Sim}_{G^{r(d)}} - \epsilon$, $\mathcal{G}^{r(d)} = \emptyset$, $erv_t^{r(d)} = 0$,
Output: $r(d)'$
while $(\overline{Sim}_{G^{r(d)}} - \overline{Sim'}_{G^{r(d)}}) \geq \epsilon$ **do** ▷ E-Step
    $r(d)' \leftarrow r(d)$
    $\mathcal{D}^d \leftarrow \emptyset$
    $\mathcal{G'}^d \leftarrow f(d, (\overline{Sim}_{G^{r(d)}})$
    for each $d_k \in \mathcal{D} \setminus \{d\}$ **do**
        if $Sim_D(d, d_k) > h$ **and** $r(d_k) \in \mathcal{G'}^d$ **then**
            $\mathcal{D}^d \leftarrow \mathcal{D}^d \cup d_k$
    for each $t \in \mathcal{G}^{r(d)}$ **do**
        $erv_t^{r(d)} \leftarrow erv_t^{r(d)} + erv(d_t)$
    for each $t \in \mathcal{G}^{r(d)}$ **do**
        if $erv_t^{r(d)} > \overline{erv}^{r(d)}$ **then**
            $r(d)' \leftarrow r(d)' \cup \{t\}$
    ▷ M-Step
    $\overline{Sim'}_{G^{r(d)}} = \overline{Sim}_{G^{r(d)}}$
    $\overline{Sim}_{G^{r(d)}} = \frac{\sum_{k=1}^{|\mathcal{D}^d|} Sim_G(r(d), r(d_k))}{|\mathcal{D}^d|}$
return $r(d)'$

ther details we refer to (Kuhr and Möller 2019).

Classifications and Theories

Classification are structures $\mathfrak{A} = \langle tok(\mathfrak{A}), type(\mathfrak{A}), \models \rangle$ consisting of a set of tokens $tok(\mathfrak{A})$, a set of types $type(\mathfrak{A})$, and a binary satisfaction relation $\models_{\mathfrak{A}}$ between tokens as left and types as right arguments. If the set of tokens and types is finite, the satisfaction relation can be described with a *classification table* with rows standing for tokens and columns for types and an entry 1 for token b and type τ meaning that $b \models_{\mathfrak{A}} \tau$; accordingly an entry 0 means that not $b \models_{\mathfrak{A}} \tau$.

A *theory* T over a set of types Σ is a pair $T = \langle \Sigma, \vdash \rangle$ of types $\Sigma = type(T)$ and a binary consequence relation $\vdash = \vdash_T$ where the left and right arguments are subsets of Σ. $\Gamma \vdash \Delta$ is called a *sequent*. The entailment relation is required to fulfill some basic constraints that one would expect to be satisfied by a monotonic entailment relation. This leads to the notion of a *regular theory* which is a theory where $\vdash_T$ fulfills the property of identity, that is $\tau \vdash \tau$ for all $\tau \in \Sigma$, weakening, i.e., if $\Gamma \vdash \Delta$ then $\Gamma, \Gamma' \vdash \Delta, \Delta'$, and global cut, i.e., if $\Gamma, \Sigma_0 \vdash \Delta, \Sigma_1$ for each partition $\Sigma_0 \uplus \Sigma_1 = \Sigma'$ of Σ'.

For each classification $\mathfrak{A}$ the associated theory $Th(\mathfrak{A}) = \langle type(\mathfrak{A}), \vdash \rangle$ is defined to have the same types as $\mathfrak{A}$ and to have a consequence relations $\vdash$ given as follows: for all $\Gamma, \Delta \subseteq type(\mathfrak{A})$: $\Gamma \vdash \Delta$ iff for all tokens $b \in tok(\mathfrak{A})$ it holds that if $b \models_{\mathfrak{A}} \tau$ for all $\tau \in \Gamma$, then there is some $\tau' \in \Delta$ such that $\tau \models_{\mathfrak{A}} \tau'$. The definition immediately entails the fact that $Th(\mathfrak{A})$ is a regular theory.

A *regular theory interpretation* $f : T_1 \longrightarrow T_2$ of theories T_1 and T_2 is a function from $type(T_1)$ to $type(T_2)$ such that for each $\Gamma, \Delta \subseteq type(T_1)$ the following holds: If $\Gamma \vdash_{T_1} \Delta$ then $f[\Gamma] \vdash_{T_2} f[\Delta]$. Here, as usual, we used the notation $f[\Gamma] = \{f(\tau) \mid \tau \in \Gamma\}$ for the image of function f on set Γ.

Given a regular theory $T = \langle \Sigma, \vdash \rangle$, a sequent $\langle \Gamma, \Delta \rangle$ is called T-*consistent* iff not $\Gamma \vdash \Delta$. It is possible to specify a theory by its consistent partitions. This due to the result expressed in the following proposition.

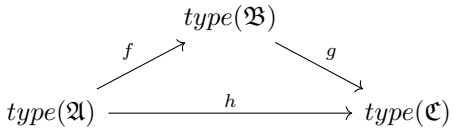

Figure 1: Commuting diagram for Prop. 3

Proposition 1 (Prop. 14 of (Barwise and Seligman 2008))**.** *Every set P of partitions of Σ is the set of consistent partitions of a unique regular theory on Σ.*

Given an arbitrary function $f : \Sigma_1 \longrightarrow \Sigma_2$ between sets of types Σ_1 and Σ_2 and a theory T over Σ_1 it is possible to define a corresponding theory over Σ_2. More concretely, let $T = \langle \Sigma_1, \vdash_T \rangle$ be a regular theory. Then the *image of T under f*, for short $f[T]$, is the theory whose type set is Σ_2 and whose consequence relation $\vdash_{f[T]}$ is defined by consistent partitions as follows: a partition $\langle \Gamma, \Delta$ of Σ_2 is $f[T]$-consistent iff $\langle f^{-1}[\Gamma], f^{-1}[\Delta] \rangle$ is T-consistent. Actually, the theory produced in that way is the one making f a theory interpretation.

Proposition 2 (Prop. 10.15 of (Barwise and Seligman 2008))**.** *Let T be a regular theory and consider a function $f : type(T) \longrightarrow \Sigma_2$. Then $f[T]$ is the smallest regular theory T' on Σ_2 such that f is a theory interpretation.*

A stronger notion of structure-preserving mapping between classifications is that of an *infomorphism*. Given classifications $\mathfrak{A}$ and $\mathfrak{B}$ an infomorphism $f : \mathfrak{A} \rightleftarrows \mathfrak{B}$ is a pair of contravariant functions $f = \langle f^\wedge, f^\vee \rangle$ such that $f^\wedge : type(\mathfrak{A}) \longrightarrow type(\mathfrak{B})$ is a function mapping types of $\mathfrak{A}$ to types of $\mathfrak{B}$ and $f^\vee : tok(\mathfrak{B}) \longrightarrow tok(\mathfrak{A})$ is a function mapping (in the reverse direction) the tokens of $\mathfrak{B}$ to tokens of $\mathfrak{A}$ and such that the following fundamental property of infomorphisms is fulfilled for all $b \in tok(\mathfrak{B})$ and types $\tau \in tok(\mathfrak{A})$: $f^\vee(b) \models_{\mathfrak{A}} \tau$ iff $b \models_{\mathfrak{B}} f^\wedge(\tau)$.

Infomorphisms can be composed to get new infomorphisms. A proposition relevant for our results is part of Lemma 4.17 in (Barwise and Seligman 2008).

Proposition 3 (Lemma 4.17(2) of (Barwise and Seligman 2008))**.** *Assume there are contravariant pairs of functions $f : \mathfrak{A} \rightleftarrows \mathfrak{B}$, $g : \mathfrak{B} \rightleftarrows \mathfrak{C}$, and $h : \mathfrak{A} \rightleftarrows \mathfrak{C}$ such that the diagram in Fig. 1 commutes. Then: if g and h are infomorphisms and $g^\vee$ is surjective, then f is an infomorphism.*

Topic Representations via Theory Interpretations

The current success of ML algorithms has lead to discussions on the understanding and interpretation of their outcomes in human understandable terms, which is the heart of the explainable AI. The motivation for investigating *representations* for deep learning algorithms is similar: in many state-of-the art algorithms the algorithm is allowed to adhere to latent structures whose meaning is usually not in formal

qualitative logico-semantical let alone human understandable terms, e.g., topic models, we use as the main example in this paper. In LDA, we can compare documents with each other w.r.t. their topic mixtures and as such allow a convenient method for information and document retrieval. However, the concrete "semantics" of the topics is not unveiled by LDA. When humans talk about topics of a text they talk about it in qualitative terms which can, to some degree, be represented as RDF triples. But how to bridge both worlds?

We assume topic mixtures to be of Boolean nature and that we have a corpus containing set of documents $d_1, \ldots, d_n$ which are our tokens, and a set of topics $\Sigma_1 = \{\tau_1, \ldots, \tau_m\}$. Each document d_i is represented as an incidence vector of length m, where a 1 at position j indicates that document d_i contains topic j and a 0 indicates the absence of that topic leading to a classification table termed DTM (document topic matrix) with types being all topics and tokens being all documents. We can consider the classification as emergent, qualitative properties resulting from a probabilistic model. Such a classification contains a logical structure we are using to represent the topics in a logical framework such as RDF(S). As defined in the preliminaries, for each classification $\mathfrak{A}$ one can define a canonical theory $Th(\mathfrak{A})$ which describes the entailments. Considering the classification of documents an example sequent $\Gamma \vdash \Delta$ would be $\tau_1, \tau_2 \vdash \tau_3, \tau_4$ where $\Gamma = \{\tau_1, \tau_2\}$ is the antecedent consisting of topics τ_1 and τ_2, where Δ is the succedent consisting of the topics τ_3, τ_4. Such a sequent holds in the classification if for each document of type τ_1 and type τ_2 it is the case that it is of type τ_3 or type τ_4. (Note that being of type τ_3 or of type τ_4 may be different for each document.) There are some sequents that one expects to hold purely due to "logical reasons"—independent of the classification, e.g., in any classification, any sequent containing a type both in the antecedent and the succedent holds in the classification. This and two other properties lead to the notion of a regular theory (see preliminaries). And in fact, any theory induced by a classification can be easily shown to be regular. The reason to consider not only theories induced by classifications is that we consider to move theories from one classification to another. Along this movement the theory may possibly not be represented as the canonical theory induced by a classification. This notion of a theory still does not talk about logical constructors but it is a logical theory in the sense that it treats the comma in the antecedent of a sequent as conjunction and the comma in the succedent as a disjunction. Moreover, with the regularity property the notion of a theory reflects some intuitive properties from an entailment relation $\vdash$. As for the logical constructors, we note that in the closed environment given by the classification one can check whether such a classification is rich enough to provide, say, Boolean operators—and if not, one can close up the classification so that it does. The classification with topics has proved very useful for typical tasks of document and information retrieval. The main problem is that topics are given only as distributions of words resulting from an optimization process on a Bayesian network. Such a description is only of marginal use for humans which would rather profit from characterizing each topic by a simple label, say an RDF type such as $DBpedia : car$, denoting the topic of cars or $DBpedia : dog$ denoting the dog topic. Let us denote the classification consisting of a set of topic RDF descriptions as types Σ_3 and the same set of documents as tokens for DT by RNT for RDF named topic. For each topic label tl we may assume a corresponding RDF database $f'(tl)$ which exactly circumscribes the topic. For example, for $DBpedia : car$ we may have RDF triples stating that a car has wheels.

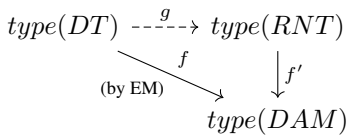

Figure 2: The general situation

An annotation of documents with topic labels is either not given, not known or not complete. But usually, there are other annotations describing some facts in the document the annotations are associated with. We assume that for all documents $d_i \in \mathcal{D}$, we may have annotations as RDF triples, and that for each document there is an associated annotation RDF repository. As the annotations may be of different nature we allow the annotations to refer to the sub-graph mechanism to distinguish between them. So, we have a classification matrix DAM (document annotation matrix) with the same set of documents but with a different set of types $\Sigma_3 = \{g_1, \ldots, g_k\}$, where g_i denotes the repository of d_i.

We only know about the function f' mentioned above. The main function g, we are interested in, is the function $g : \Sigma_1 \longrightarrow \Sigma_2$ which assigns each topic an RDF type and we would need at least a clue on the function $f : \Sigma_1 \longrightarrow \Sigma_3$ to find g. We are going to argue that the EM algorithm gives the necessary support in finding f. Having f and f', one can find the representation function g one is aiming for. Figure 2 describes the general situation.

But, what kind of further constraints functions f, f', g have to fulfill to guide the construction? In the setting of (Barwise and Seligman 2008) the strongest constraint is an infomorphism between classifications. Below, we consider this strong constraint and why it is not the first choice to follow. Here, we rather follow the idea of moving theories from one classification to another—though we note that each theory interpretation induces also an infomorphism between canonical classifications associated with them. Each such adapting the theory of the annotation classification to the theory induced by the topic classification. The main task is to find an interpretation g by trying to construct an appropriate theory interpretation f assuming that f' can be constructed as a theory interpretation and the EM algorithm presented in Section easies the construction of f.

Finding a Theory Interpretation f

(Barwise and Seligman 2008) assumes the existence of infomorphisms or interpretations f to develop their whole theory. But in many cases, the function f is unknown but only some classifications as domain and range of the function are known. We construct such an infomorphism using

these classifications and the other knowledge on the domains. In our case, we assume that the classification given by the classification with topics (DT) is trustworthy and complete, but the classification with the RDF annotations is rather not complete. So we will allow ourselves changing the classifications w.r.t. the RDF annotations when searching for an appropriate function f. We are seeking a function $f : type(DT) \longrightarrow type(DAM)$ that is a theory interpretation w.r.t. the canonical theory $Th(DT)$ over the classification and some regular theory T over the classification DAM, which has its own canonical theory $Th(DAM)$ but due to the incompleteness of the annotations we do not necessarily expect this theory to be exactly one allowing to define f as theory interpretation. Thus, we allow a change of a set of tokens of DAM to change its associated theory. We note, that our assumptions give rise to a special asymmetrical scenario but the framework allows to consider the case where classification DT is incomplete, too. In that case one would consider moving the canonical theory of DAM to DT in the inverse direction leading to the theory $f^{-1}[Th(DAM)]$. The idea of the EM algorithm in terms of theory interpretation approach is to consider a similarity function between the documents $\sim_{DT}$ based on a distance function. The distance function is defined on the base of types such that it depends monotonically on the Hamming distance of the documents. A similar assumption is made for the similarity relation $\sim_{DAM}$ over the classification DAM.

The main idea of the iterative algorithm 1 is to make those documents that are similar w.r.t. $\sim_{DT}$ and also similar w.r.t. $\sim_{DAM}$ even more similar w.r.t. $\sim_{DAM}$. This is justified w.r.t. the general aim of constructing a theory interpretation. The less different documents are the less counter models exist for sequents, and making documents more similar leads to more sequents not being falsified in classification DAM. Of course, the general strategy of considering a smaller set of tokens leads to more theories being accepted. But algorithm 1 considers only documents that are justified to be made similar in the sense that they are similar w.r.t. the categorization DT.

Proposition 4. *Algorithm 1 changes the category DAM such that the probability of finding a function $f : type(DT) \longrightarrow type(DAM)$ considered as theory interpretation between $Th(DT)$ and $Th(DAM)$ is increased.*

For the resulting saturated classification DAM there may stil be many different potential candidates f for a theory interpretation or, still, further changes on DAM' may be required. We follow a general Occam's razor principle and require the additional changes, resulting in DAM'', to be minimal, i.e., a candidate theory interpretation f has to be such that $f = argmin_{f'} \mid DAM' - DAM'' \mid$. We used Hamming weight $\mid A \mid$ for a matrix A: it is the number of ones in A.

Finding an Infomorphism f

When searching for functions f, f', and g we chose to consider a constraint on the functions that requires them to be theory interpretations. A stronger notion is that of an infomorphism, which is based on a contravariant pair of functions and has to fulfill the fundamental property of infomorphisms. We discuss here what it would mean to require the functions f, f', and g to be infomorphism by illustrating it for f. As we work with the same set of documents we assume that the token-relating function $f^{\vee}$ of the infomorphism is the identity function. So, function f between types that we seek is the function $f : \Sigma_1 \longrightarrow \Sigma_2$ such that for all tokens b and types $\tau \in \Sigma_1$ it holds that $b \models \tau$ iff $b \models f(\tau)$. The fundamental property says that the topic classification must be contained in the classification table w.r.t. the RDF annotation—after applying possibly some permutation of the columns. This, may be rather seldom due to the incompleteness and incorrectness of the annotations. Thus, some changes must be allowed, but we require these changes to be minimal, following a general Occam's razor principle.

For a subset of types $S \subseteq \Sigma_2$, let DAM_S denote the submatrix where the columns of DAM are restricted to those appearing in S. So, what we are seeking is a function f such that $f = argmin_{f'} \mid DTM - DAM_{f'[\Sigma_1]} \mid$.

In the topic scenario, the constraints associated with infomorphisms seem too strong. Though there seems to be correspondence between the topics and the RDF annotations we do not expect this correlation be in such a way that a document fulfills a topic if and only if it fulfills some annotation. Rather, there is a more holistic correspondence which was the main theme of the section before, namely, the preservation of relations between sets of topic types Γ, Δ (in form of sequents $\Gamma \vdash \Delta$ of $Th(DT)$) as relations between their images (as sequents $f(\Gamma) \vdash f(\Delta)$ in $Th(DAM)$).

Related Work

Our approach is couched in the framework of distributed logics (Barwise and Seligman 2008), which shares its main ideas with the theory of Chu spaces (Barr 2006). We have touched only a small part of the framework from Barwise and Seligman and much more of this work is relevant for our analysis of topics. We note here only that the notion of theory we used is part of the notion of a *local logic*, for which—next to a classification and a theory—one has to specify a set of so-called *normal* tokens. This additional structure allows to handle exceptions in a similar way as done in non-monotonic logics, which is also helpful in our scenario where there may be accidental or exceptional annotations which one wants to account for. (Barwise and Seligman 2008) use the notion of a logic very general, but this generality is appropriate because very diverse systems, as in our scenario of text analysed in LDA and RDF terms, have to be accounted for. A precursor to a very general approach to logics can be found under the term *institutions* as developed by (Goguen and Burstall 1984). Barwise and Seligman also discuss a general approach for representation, referring in particular to the results of Shimojima's PhD thesis (Shimojima 1996). The notion of representation underling our approach is just that of a theory interpretation. As we gave only rough descriptions by two classifications (without going onto the word level of each text in the corpus) there was no further requirement to refer to the fine-grained notion of representation according to (Barwise and Seligman 2008).

A further class of related approaches are those based on RDF vector space embeddings, such as RDF2vec (Ristoski and Paulheim 2016) or its global variant as described in (Cochez et al. 2017), having the idea to project RDF repositories to a low-dimensional domain and thereby to provide a data structure that can serve as an input for ML algorithms. The exact connections of our framework to that of (Ristoski and Paulheim 2016) and (Cochez et al. 2017) still have to be worked out, but, on the first sight, the kind of integration we propose is symmetrical, allowing information flow in both directions, whereas in (Ristoski and Paulheim 2016) and (Cochez et al. 2017) the integration is one-sided—the RDF world serving the ML world. Moreover, the framework we propose to use has a notion of a theory, whereas that of (Ristoski and Paulheim 2016) and (Cochez et al. 2017) is not intended to do.

Somehow related to our approach are approaches that deal with ontology mappings and multi-context systems. We refer the reader to (Kalfoglou and Schorlemmer 2003) for a ontology mappings and to (Brewka and Eiter 2007) for multi-context systems. Multi-context systems interlinking heterogeneous knowledge sources by modelling the flow of information among different contexts. The aim of ontology mapping is similar to the aim as illustrated in our LDA scenario: one has to find a mapping between the possibly very heterogeneous ontologies. There is still much interest in ontology mappings also in the semantic web community as, e.g., the ontology alignment initiative.

Conclusion and Outlook

We propose to use classifications as the lowest common denominator for statistical models developed in ML on the one hand and logical models used in the semantic web community, on the other hand. Working with such a data structure and additional concepts definable over them allows a symmetric form of semantic integration that many upcoming applications in the intersection of ML and the semantic web are aiming at. We considered a concrete application (annotation enrichment) in the intersection of information retrieval and the semantic web and showed that it can be analysed in the framework of classifications. There are still some points that call for further investigations. For example, the exact interplay of similarity relations and the notion of a theory has to be worked out in more detail. Moreover, we think that our analysis based on the notion of a theory on a classification is not the whole story, though it is surely the core of it: the notion of local logics and the stronger notion of infomorphisms would also have to be accounted for. But, we think that the observations and ideas of this paper could be of interest for other researchers in the AI community who work in the intersection of ML and the semantic web.

References

Barr, M. 2006. The chu construction: history of an idea. *Theory and Applications of Categories [electronic only]* 17:10–16.

Barwise, J., and Seligman, J. 2008. *Information Flow: The Logic of Distributed Systems*. New York, NY, USA: Cambridge University Press, 1 edition.

Bengio, Y.; Courville, A.; and Vincent, P. 2013. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* 35(8):1798–1828.

Birkhoff, G. 1973. *Lattice Theory*, volume 25 of *American Mathematical Society Colloquium Publications*. Providence, R.I.: American Math Society.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3:993–1022.

Brewka, G., and Eiter, T. 2007. Equilibria in heterogeneous nonmonotonic multi-context systems. In *AAAI*, volume 7, 385–390.

Cochez, M.; Ristoski, P.; Ponzetto, S. P.; and Paulheim, H. 2017. Global RDF vector space embeddings. In d'Amato, C.; Fernández, M.; Tamma, V. A. M.; Lécué, F.; Cudré-Mauroux, P.; Sequeda, J. F.; Lange, C.; and Heflin, J., eds., *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part I*, volume 10587 of *Lecture Notes in Computer Science*, 190–207. Springer.

Deng, J.; Ding, N.; Jia, Y.; Frome, A.; Murphy, K.; Bengio, S.; Li, Y.; Neven, H.; and Adam, H. 2014. Large-scale object classification using label relation graphs. In Fleet, D.; Pajdla, T.; Schiele, B.; and Tuytelaars, T., eds., *Computer Vision — ECCV 2014*, volume 8689 of *Lecture Notes in Computer Science*. Springer International Publishing. 48–64.

Ganter, B.; Franzke, C.; and Wille, R. 2012. *Formal Concept Analysis: Mathematical Foundations*. Springer Berlin Heidelberg.

Goguen, J. A., and Burstall, R. M. 1984. Introducing institutions. In Clarke, E., and Kozen, D., eds., *Logics of Programs*, 221–256. Berlin, Heidelberg: Springer Berlin Heidelberg.

Kalfoglou, Y., and Schorlemmer, M. 2003. Ontology mapping: The state of the art. *Knowl. Eng. Rev.* 18(1):1–31.

Kuhr, F., and Möller, R. 2019. Constructing and maintaining corpus-driven annotations. In *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*, 462–467.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, 1135–1144. New York, NY, USA: ACM.

Ristoski, P., and Paulheim, H. 2016. Rdf2vec: RDF graph embeddings for data mining. In Groth, P. T.; Simperl, E.; Gray, A. J. G.; Sabou, M.; Krötzsch, M.; Lécué, F.; Flöck, F.; and Gil, Y., eds., *The Semantic Web - ISWC 2016 - 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part I*, volume 9981 of *Lecture Notes in Computer Science*, 498–514.

Schmidhuber, J. 2014. Deep Learning in Neural Networks: An Overview. *ArXiv e-prints*.

Shimojima, A. 1996. *On the Efficacy of Representation.* Ph.D. Dissertation, Indiana University, Bloomington, IN.