Diplomarbeit

Semi-automatic ontology engineering and ontology supported document indexing in a multilingual environment

von

Boris Lauser

eingereicht am 8.1.2003 beim
Institut für Angewandte Informatik
und Formale Beschreibungsverfahren
der Universität Karlsruhe

Referent: Prof. Dr. Rudi Studer
Betreuer: Raphael Volz, Andreas Hotho

Heimatanschrift:
Taubenstrasse 9
74906 Bad Rappenau

Studienanschrift:
FAO of the UN
Library & Documentation Systems Division
Viale delle Terme di Caracalla
00100 Rome, Italy

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

1 Introduction

1.1 Motivation

The management of large amounts of information and knowledge is of ever increasing importance in today's large organisations. With the ongoing ease of supplying information online, especially in corporate intranets and knowledge bases, finding the right information becomes an increasingly difficult task. Today's search tools perform rather poorly in the sense that information access is mostly based on keyword searching or even mere browsing of topic areas. This unfocused approach often leads to undesired results. The following example illustrates the problem more clearly:

> An agriculture scientist would like to find out which organisation established the Agreement on Agriculture. A simple search for "establish Agreement on Agriculture" might result in a huge list of documents containing these words, but actually none of them containing the desired result: WTO or World Trade Organisation. The problem becomes even worse if the result searched for only appears in a foreign language document.
>
> Figure 1 shows an extract of an ontology, which could solve this problem by following links in a graph. The grey ellipses represent generic concepts, whereas the white ones represent specific instances of these concepts. The two concepts shown here are linked by a relationship. An ontology-enabled search application would first identify "Agreement on Agriculture" as a "standard" and would then detect the relationship "establish" to "international organisation" and its instances, and hence solve the problem by extending the search query. This example shows how ontologies can help to improve the management of information. Furthermore, it could provide added value by detecting other relationships that provide the user with more possibilities: for example, standards of other organisations could be presented.

Semantically annotated documents, i.e. documents that are indexed with ontological terms and concepts instead of simple keywords, provide several advantages. First, the ontological abstraction provides robustness against changes in the document. In the above example, the document representation might change using the term 'Agricultural Agreement' instead of 'Agreement on Agriculture'. However, since the document has been annotated with the ontological semantics, this will not affect the search results. Second, since the ontology used

for annotating the document in this example is domain-specific, the semantic meanings and interpretations of keywords are bound to that domain and therefore the retrieval is likely to be more efficient. A term can have several meanings in different domains. By first mapping the keyword to its semantic representation in a specific ontology and using the ontology's linked knowledge structure, a much more focused search approach can be taken. Third, document specific representations no longer affect the search. This is extremely important in the case of multilingual representations. Keywords of several languages are mapped to the same concept in an ontology and are therefore given the same meaning. Multilingual search portals can be established to produce the same results, no matter which language is used for retrieval.

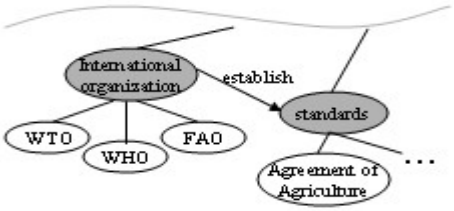


Figure 1: Ontology example, excerpt

An important task in knowledge management facilitating above described search scenario id the classification and indexing of documents. At present, subject specialists are responsible for this time consuming process. However, with today's vast amount of available information on the WWW, automatic support is needed to efficiently manage this task. Ontologies play a critical role in supporting the machine readable semantics needed to facilitate automation. They can be used for providing the categories and keywords needed to describe the content of documents. Automatic text classification tools still lack the necessary precision to replace human indexers and need to be extensively evaluated in different domains.

Before such powerful Semantic Web[1] applications can be built and used within certain domains of knowledge, the basic requirement - a machine readable vocabulary represented by a domain ontology - has to be established. The creation of ontologies is a time consuming task and often carried out in an ad-hoc manner. Only few methodologies exist and existing ones are often extremely complex and need extensive training and expertise. Even less automated tool support is available. Constituting the knowledge base for future Semantic Web applications, domain ontologies have to be created continuously in all possible areas and communities. The need for a reusable methodology is evident.

[1] Refer to [Pal01] for a short introduction to the Semantic Web.

1.2 Approach

The thesis introduces a comprehensive framework for building a domain-specific ontology. The approach combines classical methodologies for human-based ontology engineering with semiautomatic support of a heuristic toolkit. Two methods for ontology acquisition are applied in order to create the domain ontology. The first is to create a small, domain-specific core ontology from scratch. This step is supported by automatically extracting interesting concepts from a corpus of domain texts, which can be used to extend this base ontology. The second acquisition approach takes a well-established thesaurus as a basic vocabulary reference set, and converts it into an ontology representation. Then, a domain specific and a general corpus of texts are used to remove ontology concepts that are not descriptive for the domain from this converted representation. The rational used here is that domain specific concepts are more frequent in the domain-specific text corpus. The results of these steps are assessed to assemble a first version of the domain specific ontology. This ontology is then accessible through a multilingual web portal to be incorporated into other applications, such as document indexing or keyword searching of indexed documents. It could eventually be used to automatically index documents available through this kind of search application.

Carried out in collaboration with the Food and Agriculture Organisation (FAO)[2] of the United Nations (UN), the main focus of this thesis is on the adoption of the proposed framework to the specific environment and needs of this large organisation. The framework has been applied to create a prototype biosecurity ontology for the domain of Food Safety, Animal and Plant Health to be incorporated into an Internet Portal to this domain. Within this context, the conversion of a thesaurus into an ontology and evaluations of two automatic tools especially, constitute the central parts of the academic research work. The first evaluation is on a tree-pruning algorithm used in the ontology creation process to retrieve domain specific concepts from the converted thesaurus. The second evaluation is on a text classification application based on support vector machines, enhanced by a domain specific ontology serving as background knowledge for the classification algorithm.

[2] [http://www.fao.org].

1.3 Outline

The next section gives an introduction and overview about the Food and Agriculture Organisation, and the Agricultural Ontology Service (AOS) Project, which provides the bigger context in which the research work of this thesis is embedded. The current information management structure will be introduced briefly, outlining the overall current status and problems within the organisation.

In section 3, I will give an introduction to the idea of the Semantic Web as well as to ontologies and their various representations and engineering approaches. The comprehensive framework for the creation of a multilingual domain ontology is covered in section 4. The application of the framework will be described in the context of the above-mentioned project to establish an International Portal on Food Safety, Animal and Plant Health. The conversion of an existing thesaurus into an ontology representation as well as the adaptation of a multilingual ontology web browser to be embedded into the system is discussed here in detail.

Sections 5 and 6 describe in detail the adaptation and evaluation of two automatic tools constituting parts of the framework. Section 5 describes the thesaurus pruning algorithm used within the ontology creation framework and discusses the results of an empirical evaluation carried out within the context of the project. Section 6 introduces the reader to the area of automatic text classification and describes the adaptation of an already existent automatic text classifier based on support vector machines to incorporate domain specific ontologies. Several evaluation results are discussed against the question of the applicability of the classifier in the context of the FAO and against results of earlier evaluations. Finally, section 7 summarises the findings and results and provides an overview on future work.

2 The project environment

2.1 FAO and the AOS

The Food and Agriculture Organisation (FAO) of the United Nations (UN) was founded in 1945 with a mandate to raise levels of nutrition and standards of living, to improve agricultural productivity, and to better the condition of rural populations. Today, FAO is one of the largest specialised agencies in the United Nations system and the lead agency for agriculture, forestry, fisheries and rural development. As an intergovernmental organisation, FAO has 183 member countries plus one member organisation, the European Community.

Considering the scope of the organisation, knowledge management is vital for effective decision-making. One of the FAO's visions within the context of its strategic framework is to be a centre of excellence and an authoritative purveyor of knowledge and advice in the sphere of its mandate. FAO has a mandate to collect, analyse, interpret and disseminate information relating to nutrition, food, agriculture, forestry and fisheries. The Organisation serves as a clearing-house, providing farmers, scientists, government planners, traders and non-governmental organisations with the information they need to make rational decisions on planning, investment, marketing, research and training.

The World Agricultural Information Centre (WAICENT)[3] is FAO's strategic inter-departmental programme on information management and dissemination. WAICENT provides a corporate information platform for the acquisition, updating and dissemination of FAO information.

There is no doubt that the Web provides a potential platform for global access to this information, but it was not initially envisioned as a tool for global access to information, and the underlying standards for information management are not entirely adequate. By the very nature of the Internet's architecture, information on similar subjects is scattered across many different servers around the world, yet there are few tools to integrate related information from different sources. As a result, it is often very difficult to find things on the Web. This is equally evident in FAO's information system and will be described further in the next section, when the structure of this system will be introduced.

Such problems can only be solved if action is taken to establish appropriate norms, vocabularies, guidelines and standards to facilitate the integration of data from different sources, and to engage in effective data exchange. Through the adoption of international

[3] [http://www.fao.org/waicent/index_en.asp].

classification schemes, controlled vocabularies, open standards, and common data models we will eventually overcome many of the information management problems of the Internet; through the development of tools that exploit such standards it will ultimately be possible to provide an effective framework for "one-stop shopping", where people can search for agricultural information resources in one place, without having to explore many different individual web sites.

In the agricultural sector there exist already many well-established and authoritative controlled vocabularies, such as FAO's AGROVOC Multilingual Thesaurus, the CABI Thesaurus[4], and AgNIC[5], the thesaurus of the National Agricultural Library in the United States. Ontology is a new concept extending the traditional thesaurus approach by structuring the concepts more formally and providing richer relationships among those. By more formally structuring the context and meaning of terms, ontologies become an integral part of the Semantic Web, described by Tim Berners-Lee in [BHL01] as "an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in co-operation".

In response to such a new approach to managing vocabularies, WAICENT has recently issued a Concept Note ([AOS01]) for the development of an Agricultural Ontology Service (AOS). The AOS project will function as a tool to help structure and standardise agricultural terminology in multiple languages for use by any number of different systems around the world. The main objectives of the AOS are to provide a framework for:

- Better indexing of resources;
- Better retrieval of resources; and
- Increased interaction within the agricultural community.

With respect to the Semantic Web initiative, the AOS would strive to:

- Increase the efficiency and consistency with which multilingual agricultural resources are described and associated together;
- Increase functionality and relevance in accessing these resources; and
- Provide a framework for sharing common descriptions, definitions and relations within the agricultural community.

[4] The CABI Thesaurus is a thesaurus of the applied life sciences and the world's largest for agricultural sciences and related subjects, currently available at [http://194.203.77.66/, Dec2002].

[5] Available at [http://www.agnic.org/, Dec 2002].

Once constructed, the Agricultural Ontology Service will offer a contextually rich and modern framework for modelling, serving, and managing agricultural terminology. When integrated with Web-based search tools, it will facilitate resource retrieval, not only providing access to the specific documents that a particular individual is looking for, but also offering suggestions for other related resources that are potentially relevant to the topic of interest. As an integral part of WAICENT, the AOS will pay a strategic role in FAO's effort to fight hunger with information.

The research work of this thesis is carried out in the context of the AOS project and creates a first step towards its main objectives. By providing a comprehensive framework for semiautomatic creation of multilingual domain ontologies and showing first results of embedding them into an automatic text classifier, the thesis acts as a feasibility study to achieve the overall objective of integration of information across all agriculture domains.

2.2 Information management at the FAO

The FAO stores and manages a vast amount of data across all agriculture domains. Information at FAO is stored and made available at two different levels: FAO-wide as well as in the respective departments. Currently, there is no single access point through which all information resources are accessible and the various information resources are scattered across the different systems and departments. Hence, different storage bases have to be accessed in order to find the necessary information. The following gives a rough overview about the current system:

2.2.1 Resources and metadata

FAO manages different types of resources. Resource in this context means a piece of information or an information item in digital, print or any other media format. Mainly the following resources are made available through the various FAO information systems, though the resources themselves are not necessarily electronically available.:

- Monographs (Books, Newspapers, Journals…)
- Analyticals (single articles)
- WebPages
- Photos and multimedia items
- Press releases
- Publications (printed and not changeable resources)

FAO provides electronic access by describing all resources with metadata, which is basically data about other data. The Agricultural Metadata Element Set Project (AgMES)[6] extends the proposed elements of the Dublin Core Metadata Initiative (DCMI)[7] to provide a resource description element set for FAO's agricultural resources. Elements such as title, author or subject can describe a resource. The full set of elements can be seen at the respective web sites. The subject element of a resource description set captures the content of a resource with some representative keywords and is considered the most delicate and difficult to create element, since it is basically responsible for discovery of the resource in the system. The work presented throughout the remainder of this thesis basically all deals with this metadata element. Metadata in the FAO is stored in various databases and made accessible to the users through different access points. The following paragraph will give an overview on that system.

2.2.2 The information management system

Currently the FAO basically stores documents and metadata using two different systems:

- EIMS (Electronic Information Management System)
- FAO Document Online Catalogue (FAODOC)[8]

The **EIMS (Electronic Information Management System**) collects and manages metadata and keywords linked to any electronic information object, such as publications, web pages, images or videos, produced by every Department. It stores this metadata in different databases:

- FAO Corporate Document Repository (FAO DocRep)
- Website database
- Multimedia database

The **FAO Corporate Document Repository** (DocRep) houses FAO documents and publications, as well as selected non-FAO publications, in electronic format. The other databases house their respective information. This electronically available information can be accessed through

- FAO Information Finder[9]

[6] [http://www.fao.org/agris/agMES/default.htm].
[7] [http://dublincore.org/].

- FAO Document Repository web interface

While the latter only queries the Document Repository, the Information Finder queries the whole EIMS system.

FAODOC contains metadata about analytical (articles) and monographic records (books, serial titles). Large parts are therefore not electronically available. The FAODOC Database stores metadata about all these items. Currently the FAO Online catalogue can only be queried through the Online Catalogue Interface. If a document is available in electronic format (in the FAO Document Repository), a link to that document into the Document Repository is provided. No further integration with the EIMS system exists so far.

Whereas the metadata information stored in FAODOC has been created and maintained by a rather small, well-trained group of people over a long time period, metadata in the EIMS is edited by a bigger, less trained group of people, which might lead to less consistent records. Figure 2 shows an overview of the current system information flow. It shows the different interfaces through which publishers can populate the system with metadata, the information flow between the systems and the interfaces through which users can access the information.

In addition to these FAO wide cross-domain systems, each department maintains its own department web site hosting information not necessarily retrievable through the FAO wide information management system. The hosting of information on these sites is therefore rather uncontrolled and the amount of available data and also the speed with which it is produced in the various areas forbids it to keep track of all of it in the centralised system.

Moreover, as opposed to the FAO wide systems, information in the various departments is not necessarily described using metadata. Besides lack of time and human resources, this fact also arises from the lack of domain specific vocabularies needed to describe the subject of the resources. The next section will give a more detailed introduction into controlled vocabularies and their use in subject indexing of resources.

[8] [http://www4.fao.org/faobib/index.html].
[9] [http://www.fao.org/waicent/search/default.asp].

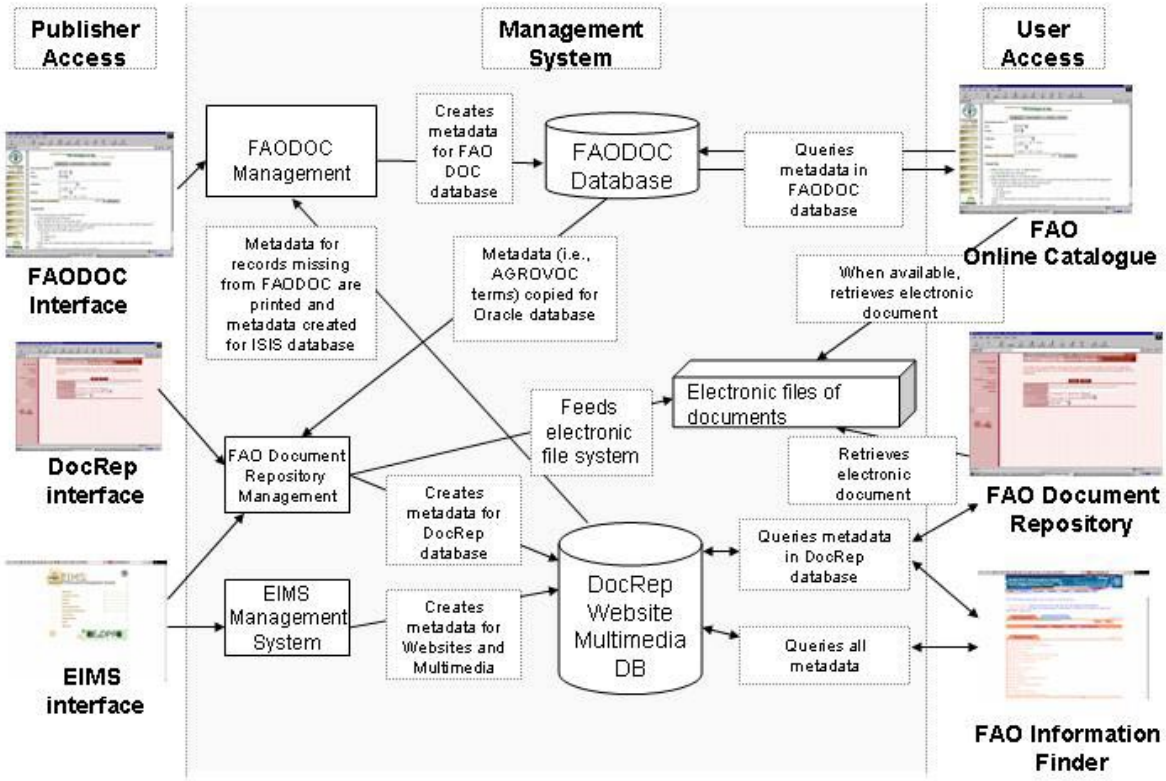


Figure 2: Information management system at the FAO

2.2.3 AGROVOC Thesaurus and Document Indexing

Subject indexing is the act of describing a document in terms of its subject content. The purpose of subject indexing is to make it possible to retrieve easily references on a particular subject. It is the process of extracting the main concepts of a document, representing those concepts by keywords in the chosen language and associating these keywords with the document. In order to be unambiguous and carry out this process in a more standardised way, keywords should be chosen from a controlled vocabulary. The subject element of the metadata element set, as described before, contains such keywords to describe a resource.

AGROVOC[10] is a multilingual agricultural thesaurus designed to improve information indexing and retrieval through the use of a controlled vocabulary in the agriculture domain. It was developed by FAO and the European Community (EC). The Third Edition was published in 1996, and a supplement followed. It exists in the 5 official FAO languages English, French,

[10] The full AGROVOC is available online at [http://www.fao.org/agrovoc/].

Spanish, Arabic and Chinese and has been translated into further languages, such as Portuguese, Thai and others. Other versions of AGROVOC have been prepared and are being maintained by national centres or by groups of countries sharing those languages. It is a controlled vocabulary designed to describe information resources in the fields of agriculture, forestry, fisheries, food and related domains (such as environmental terms).

The main role of a thesaurus is to standardise the indexing process through a controlled vocabulary. For example, it informs users and indexers that systems using AGROVOC use the term INSECTICIDES to subject index records that pertain to this concept instead of LARVICIDES or APHICIDES.

The vocabulary of the AGROVOC consists of a collection of keywords, which are descriptors or non-descriptors. A descriptor is a preferred term/keyword to index a document, whereas a non-descriptor is a non-preferred term, to be replaced by its associated descriptor(s) for indexing purposes. In the above example, INSECTICIDES is a descriptor and the other two keywords represent non-descriptors. Only descriptors should be used for indexing purposes. In the current version of the AGROVOC, there are 16607 descriptors and 10760 non-descriptors. Descriptors are arranged in a broader term – narrower term taxonomic hierarchy structure, i.e. for each descriptor there might be several more special as well as more general terms. Other relationships linking the keywords are:

- **Related term**, expressing some kind of relationship between this keyword and another.
- **Use**, declaring the keyword to be a non-descriptor to use another keyword for indexing purposes.
- **Used for**, showing that this keyword is used as a descriptor for another keyword.
- **Used for+**, expressing that this descriptor has to be used in conjunction with another descriptor to replace the linked non-descriptor.

Moreover, each keyword is translated into the different languages. Figure 3 shows a descriptor and a non-descriptor with its hierarchy structure and relationships in the current version of the AGROVOC.

The keywords of the AGROVOC are, furthermore, mapped to a collection of 116 subject categories. These categories are used besides the keywords for indexing purposes. A full listing of all AGROVOC categories is attached in Appendix C.

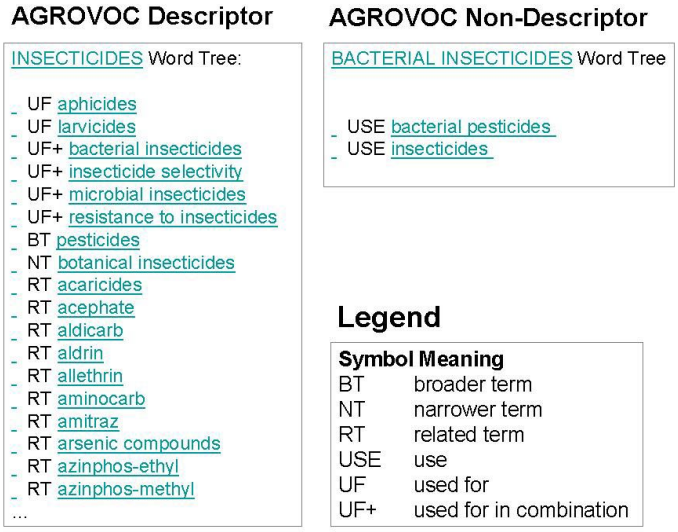


Figure 3: AGROVOC thesaurus: A sample extract showing a descriptor and a non-descriptor

The kinds of relationships used in this thesaurus certainly limit its expressiveness. The relationship 'related term' for example does not say anything about the kind of relationship. The terms could be related in any possible way. This lack of expressiveness will be examined further in section 3 when ontologies are introduced. Moreover, there are some multilingual issues and modelling restrictions that cannot be addressed using the limited thesaurus structure. AGROVOC has been translated into different languages. This translation has been done in a simplified way, starting from the English collection of keywords and then trying to find a direct translation for each term. Whenever there is no translation for a keyword, because it does not exist in the target language, the English word remains. The translation of concepts into another language is, however, more complex in reality. A keyword in one language can sometimes only be described by more than one keyword in another language. On the other hand, in one language many different concepts exist, all having the same basic meaning than one concept in another language. Consider an example taken from the Chinese translation of the AGROVOC. The English word 'abortion' expresses the sense of concept no matter in which context it is used. In Chinese, there is no perfect equivalent. In fact, there are three different concepts to express the concept of abortion in the human, the plant and the animal domain respectively. The simplified structure of the AGROVOC thesaurus cannot capture this information. In the case of AGROVOC, only one Chinese term (the human sense) has been chosen to represent the concept of 'abortion' in Chinese. By using this term to index

documents, a subset of Chinese documents could not be indexed or would be indexed wrongly. Hence, a Chinese searching for information on the concept of plant abortion would retrieve as well information on human and animal abortion. Currently, the AGROVOC thesaurus does not provide a solution for this problem. Another good example of this multilingual translation and concept-mapping problem is explained in [Rol01], where the non-compatible translation of the English term river into either rivière or fleuve in French is discussed. Ontologies, as introduced in the next chapter, provide the modelling capabilities to address such issues.

2.3 Problems with the current system and proposal

Currently, there is no single access point for users to effectively search for information on FAO's web sites. They are forced to browse many pages and perform many searches through trial and error. Two different groups feed two different systems with metadata in an inconsistent way. The same documents might be indexed twice in the different system in an inconsistent manner. In the Document Repository, the indexing is sometimes not done according to the rules and non-descriptors might be used for indexing. FAODOC metadata is basically more reliable and consistent, due to fewer and better-trained indexers working on it. Only 5-10% of FAO's web sites are stored in the EIMS system. Therefore, information retrieval about specific department web sites is rather poor. Lack of human resources and the fast growth of information resources create a huge backlog in metadata creation. This fact, along with regular processing inefficiencies within large organisations, makes it impossible to gather all the information in one centralised system. The decentralised structure will therefore remain and domain specific information will be available through the various domain specific systems.

The integration of these domains is the vision of the AOS. Crucial to integration of this information is, however, the creation of metadata of all these resources within the departments in a consistent and controlled way. Subject indexing especially will be responsible for retrieval of the respective resources. Automatic support for this time consuming task would be invaluable. Using controlled vocabularies to subject index domain resources sets the basis for harvesting information across several domains. The AGROVOC is not specific enough for all areas in order to be used for subject indexing in specific domains. Some agriculture domains are either not or not sufficiently captured in the AGROVOC (for example fishery, forestry or food safety). Domain specific controlled vocabularies therefore need to be established.

This is where the framework for the creation of domain specific ontologies and automatic text classification fits into the context of the Food and Agriculture Organisation. The main work of this thesis will therefore focus on these two fields. In the next chapter, I will give an introduction to the Semantic Web and define ontologies in their here used context. The underlying terminology for understanding the following chapters and the broader technological context in which this project is embedded will be introduced here.

3 Semantic Web

3.1 The idea

The idea of the Semantic Web introduced by Tim Berners-Lee the first time in 1996 [Ber96] has been described by himself as follows:

> "The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in co-operation" ([LHL01]).

The Semantic Web is basically the idea of linking information objects on the web in such a way to make them easily processable for machines. The problem with the majority of data on the Web at the moment is that it is difficult to use on a large scale, because there is no global system for publishing data in such a way as it can be easily processed by anyone. XML as specified in [BPSM00] has a widespread use in representing data in an interchangeable and reusable format. The status today is, however, that all over the web, loads of data basically talking about the same or similar issues is made available and described in XML or pure HTML. These languages lack the semantics needed in order to resolve similarity issues. Same data is modelled an indefinite number of times in different locations using different representations. The Semantic Web is an effort to unambiguously define and identify resources on the World Wide Web and to interconnect them with semantic relationships in order to provide the described resources in a machine-readable, understandable and reusable form to anyone who wants to make use of them. The Semantic Web moves from the idea to relate pieces of meaningless text, as it is basically done now with HTML and hyperlinks, towards affiliating objects with semantic relationships.

In Semantic Web terminology, every object in the world is a **resource**[11] and can be linked to any other resource. A resource can be uniquely identified by its **Uniform Resource Identifier (URI)**[12]as specified in [BFIM98]. A URI is defined as a compact string of characters for identifying an abstract or physical resource. The ability to uniquely reference and identify a resource sets the basis for the Semantic Web.

[11] Similar to resource as defined in the context of the FAO in the previous chapter, where every information object is called a resource.

[12] See also [http://www.w3.org/Addressing/].

The **Resource Description Framework (RDF)** is a foundation for exposing and processing metadata as recommended in [LS99]. It has been designed to provide interoperability between applications that exchange machine-understandable information on the WWW by offering the possibility to express statements about resources in a machine processable format. An RDF statement is a triple, always consisting of a subject, predicate and object, making it similar in format to a natural language expression. The difference here is that each part is a URI. Let us consider the following RDF statement:

<http://www.borislauser.de> <http://www.relationships.com/schema/isStudentAt> <http://www.uni-karlsruhe.de>

The subject, Boris Lauser, is a person (i.e. the resource to be described). The predicate describes that the subject is a student at some other resource (i.e. a property of the resource person). The object is the University of Karlsruhe (i.e. the value of the property; in that case, another resource). This statement can be read and processed by machines. By using URIs, everyone can use RDF to make statements about anything. RDF makes it possible to create interchangeable metadata and publish it on the web to be reused by others. So if other parties make other statements about the subject Boris Lauser, an application collecting all these statements, could relate and combine the information given in them and infer other statements.

XML has evolved as the standard format for information interchange. Therefore XML is now widely used to encode RDF statements and is suggested as the standard syntax by the W3C. RDF and XML are therefore complementary in that RDF describes a model, which can be represented using different syntaxes. XML is one syntax for doing so. Another example is Notation 3 or N3[13]. Figure 4 shows a possible XML encoding of above statement.

```
<?xml version="1.0"?>
<RDF   xmlns="http://www.w3.org/1999/02/22- rdf-syntax-ns#"
       xmlns:s="http://www.relationships.com/schema/">

       <Description about="http://www.borislauser.de">
         <s:isStudentAt resource="http://www.uni-karlsruhe.de"/>
       </Description>
</RDF>
```

Figure 4: XML serialisation of RDF, example

[13] Refer to http://www.w3.org/2000/10/swap/Primer.html for a good overview on N3.

The Semantic Web is herewith a means of creating metadata about arbitrary resources on the web, just like metadata about information resources in the FAO as described in the previous chapter. These information resources can be located by a URI, and hence, the Semantic Web idea is highly applicable in this context.

The problem so far, however, is that all these objects represented by URIs can now be talked about and processed by machines to infer statements and expressions about them, but they are nowhere defined yet. As a human being, we know, that 'Boris Lauser' is a person, but a machine does not. And different machines and statements referring to this very object might therefore interpret it differently and in an unintended way. An ontology can solve this problem by providing the opportunities to define and specify the meaning of and relationship between terms.

3.2 Ontologies

3.2.1 Introduction

The term ontology originally evolved from a branch of philosophy that deals with the nature and the organisation of reality [Gua98]. In terms of information management and in the context of the Semantic Web, many definitions of the term have been named. In [Gru95], an ontology is defined as "an explicit specification of a conceptualisation". When talking about conceptualisation in this context it is meant to identify concepts and other entities describing a domain of interest and the relationships that hold amongst them. It refers to an abstract model of how people think about physical or abstract objects in the world, usually restricted to a particular subject area. An explicit specification means the concepts and relationships of the abstract model are given explicit terms and definitions. In the context of the AOS, ontologies are referred to as a collection of terms, the definition of these terms, and the specification of relationships amongst them as stated in [AOS01]. The definition can get as loose as "a vocabulary of terms and some specification of their meaning" in [UG96].

It is not my intention to give a universal definition for ontologies here, but rather focus on how they are defined and used within the context of this research and project environment.

The definition, which probably suits best the approach taken here is given in [SBF98]: An ontology is an explicit, formal specification of a shared conceptualisation of a domain of interest. It is shared because in a certain domain (more about domains in the next section), everybody agrees and has the same view on this explicit specification. To provide the necessary formalisation, a more mathematical definition of a conceptual modelling approach

of ontologies is specified in [MMV02]. The following definitions are taken from this specification and introduce the base terminology of the ontology definition used and built upon throughout the further work of this thesis:

Defnition 1 (OI-model Structure). An OI-model (ontology-instance-model) structure is a tupelo OIM := (E; INC) where:

- E is the set of entities of the IO-models,
- INC is the set of included OI-models.

An OI-model represents a self-contained unit of structured information that may be reused. Elements in an OI-model are entities. An OI-model may include a set of other OI-models (represented through the set INC). Definition 5 lists the conditions that must be fulfilled when an OI-model includes another model.

Definition 2 (Ontology Structure). An ontology structure associated with an OI-model is a 10-tuple O(OIM) := (C; P; S; T; INV;HC;HP ; domain; range; mincard; maxcard) where:

- $C \subseteq E$ is a set of concepts,
- $P \subseteq E$ is a set of properties,
- $S \subseteq P$ is a subset of symmetric properties,
- $T \subseteq P$ is a subset of transitive properties,
- $INV \subseteq P \times P$ is a symmetric relation that relates inverse properties, if (p1; p2) $\in$ INV, then p1 is an inverse property of p2,
- $H_C \subseteq C \times C$ is an acyclic relation called concept hierarchy, if (c1; c2) $\in H_C$ then c1 is a sub-concept of c2, c2 is a super-concept of c1,
- $H_P \subseteq P \times P$ is an acyclic relation called property hierarchy, if (p1; p2) $\in H_P$ then p1 is a sub-property of p2, p2 is a super-property of p1,
- Function domain: $P \rightarrow (2^C \setminus \{\varnothing\}) \bigcup \{L\}$ gives the set of domain concepts for some property p $\in$ P,
- Function range: $P \rightarrow (2^C \setminus \{\varnothing\}) \bigcup \{L\}$ gives the set of range concepts for some property p $\in$ P,
- Function mincard: C $\times$ P $\rightarrow N_0$ gives the minimum cardinality for each concept-property pair,

– Function maxcard: $C \times P \rightarrow (N_0 \cup \{\infty\})$ gives the maximum cardinality for each concept-property pair.

Each OI-model has an ontology structure associated with it, consisting of a set definitions regulating how instances should be constructed. An ontology consists of concepts (sets of elements) and properties (specification how objects may be connected). Each property must have at least one domain concept, while its range may either be a literal, or a set of at least one concept. Domain and range concept restrictions are treated conjunctively - all of them must be fulfilled for each property instantiation. Some properties may be marked as transitive, and it is possible to say that two properties are inverse. For each class-property pair, it is possible to specify the minimum and maximum cardinalities, defining how many times a property may be specified for instances of that class. Concepts and properties can be arranged in a hierarchy, as specified by the H_C (H_P) relation. This relation relates directly connected concepts (properties), whereas its transitive closure follows from the semantics, as defined in the next subsection.

Definition 3 (Instance Pool Structure). An instance pool associated with an OI-model is a 4-tuple IP(OIM) := (I; L; instconc; instprop) where:

– $I \subseteq E$ is a set of instances,
– L is a set of literal values, $L \cap E = \varnothing$,
– Function instconc : $C \rightarrow 2^I$ relates a concept with a set of its instances,
– Partial function instprop : $P \times I \rightarrow 2^{I \cup L}$ assigns to each property-instance pair a set of instances related through given property.

Each IO-model has an instance pool associated with it. An instance pool is constructed by specifying instances of different concepts and by establishing property instantiation between instances. Property instantiations must follow the domain and range constraints, and must obey the cardinality constraints.

Definition 4 (Root OI-model Structure). Root OI-model is defined as a particular, well-known OI-model with structure $ROIM := (\{ROOT\}, \varnothing)$. ROOT is the root concept, each other concept must subclass ROOT (it may do so indirectly). Each other OI-model must include ROIM and thus gain visibility to the root concept. This is similar to object-oriented languages approaches - for example, in Java every class extends java.lang.Object class.

Definition 5 (Modularization Constraints). If OI-model OIM imports some other OIModel OIM_1 (with elements are marked with subscript 1), that is, if $OIM_1 \in INC(OIM)$ must satisfy following modularization constraints:

- $R_1 \subseteq R, C_1 \subseteq C, P_1 \subseteq P, T_1 \subseteq T, INV_1 \subseteq INV, H_{C1} \subseteq H_C, H_{P1} \subseteq H_P$,
- $\forall p \in P_1 domain_1(p) \subseteq domain(p)$,
- $\forall p \in P_1 range_1(p) \subseteq range(p)$,
- $\forall p \in P_1, \forall c \in C_1 \min card_1(c,p) \geq \min card(c,p)$,
- $\forall p \in P_1, \forall c \in C_1 \max card_1(c,p) \leq \max card(c,p)$,
- $I_1 \subseteq I, L_1 \subseteq L$,
- $\forall c \in C_1 instconc_1(c) \subseteq domain(c)$,
- $\forall p \in P_1, i \in I_1 \ instprop_1(p,i) \subseteq instprop(p,i)$.

If an OI-model imports some other OI-model, it contains all information - no information may be lost. Modularization constraints just specify structural consequences of importing an OI-model. This is independent from the implementation - imported OI-models may be physically duplicated, whereas in other cases they may be linked.

3.2.2 Types of ontologies

The notion of OI-model introduced above already incorporates the well-known software engineering paradigms of modularity and reusability provided by the INC set. These paradigms are extremely important in the discipline of ontology engineering. This becomes evident when thinking about possible different levels of how to describe things in an ontology. On one hand, an ontology might be sufficient describing the structure of an organisation on the top level, only representing the main organisational units. On the other hand, it might be necessary for an application (like a corporate knowledge base) to capture the whole organisation with all its employees in an ontology. The first high-level ontology can certainly be used as part of the second, so that common concepts don't have to be remodelled there. However, both ontologies are on different levels. Figure 5 shows an overview about the different types of ontologies as identified in [Gua98]. Guarino differentiates between four types:

Top-level ontologies describe very general concepts like space, time, matter, object, event, action, etc., which are independent of a particular problem or domain: it seems therefore

reasonable, at least in theory, to have unified top-level ontologies for large communities of users.

Domain ontologies and **task ontologies** describe, respectively, the vocabulary related to a generic domain (like medicine, or automobiles) or a generic task or activity (like diagnosing or selling), by specialising the terms introduced in the top-level ontology.

Application ontologies describe concepts depending both on a particular domain and task, which are often specializations of both the related ontologies. These concepts often correspond to roles played by domain entities while performing a certain activity, like replaceable unit or spare component.

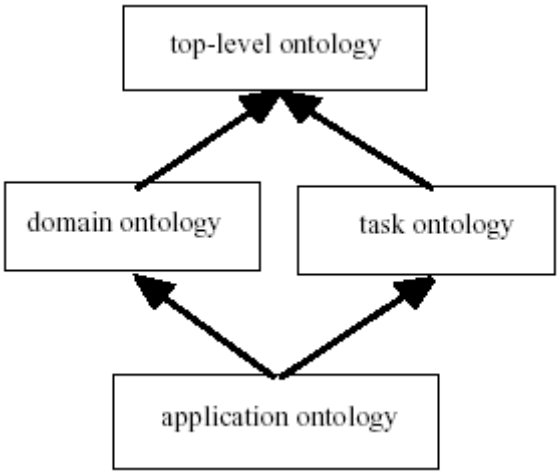


Figure 5: Ontology types

In [MIK96], this typology is even more refined taking into consideration the usage of the ontology, introducing qualifiers like task- or application-dependent/-independent. For this purpose, and taken into consideration the stage of ontology development and usage, and the insecurity inherent in it, the above given differentiation is sufficient. Given the project scope of this thesis, which aims on the integration of different domains using ontologies as discussed in the previous chapter, the further work will focus on domain ontologies respectively. Whenever using the term ontology in the following, I always refer to this particular type.

Given the definition and scope of ontologies, I will now introduce to several ways of ontology representation and the approach taken here in this context.

3.2.3 Ontology representation languages

The term 'formal' in the above ontology definitions refers to the fact that the established conceptualisation has to be formalised in a way that is unambiguous and in the context of the Semantic Web machine-readable. Several formal representation languages exist and are used today. Among the most common ones used in the past and today are:

- RDF/RDFS [BG02],
- DAML + OIL [CHH+01],
- Topic Maps [PM01],
- Ontolingua [FFR97],
- FLogic [KLW90] and
- LOOM [Bri93].

Traditional ontology representation languages like Ontolingua, FLogic or LOOM evolved from different underlying paradigms: frame-based, description logic, first and second order predicate calculus and object-oriented. Recently, new languages for the web have been created like XML, RDF and RDF Schema (RDFS). Ontology representation languages like DAML/OIL or the latest development effort OWL, are created as extensions of these. Figure 6 shows an overview of the degree of expressiveness of different representation languages taken from [CG00]. Obviously, the languages grow in expressiveness, but also in complexity from bottom to the top. Given the rather web based context of the thesis, I will briefly introduce the web based representation languages, especially RDFS, as well as the proprietary extension of it which is used within this project for modelling and representing ontologies. A comprehensive, in-depth evaluation of different languages based on their expressiveness and reasoning capabilities is given in [RC00].

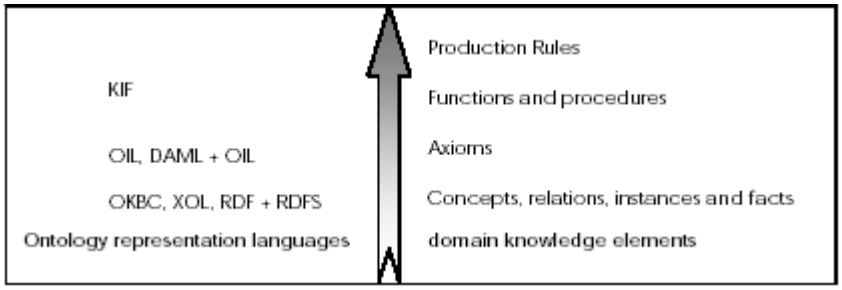


Figure 6: Ontology representation languages and their expressiveness taken from [CG00]

RDFS (Resource Description Framework Schema)

The most basic vocabulary description language and also adopted within the work of this thesis is RDFS as proposed in [BG02]. The RDF data model itself, as discussed in the previous section, provides no mechanisms for describing classes or properties, nor does it provide any mechanisms for describing the relationships between these properties and other resources. The RDF vocabulary description language defines classes and properties that can be used to describe other classes and properties. Figure 7 shows how above introduced RDF statement could now be deducted from an RDF Schema defining the classes and their relationships. In RDFS, every entity is a subclass of a resource. The basic and most important constructs, a class and a property in RDFS (according to concept and property in Definition 2 of the ontology model) are both subclasses of the root class resource. These can now be used to define own classes and relationships between them constituting the ontology layer. Here, we defined the new classes 'Person' and 'University'. A property has a domain and a range of resources (in accordance to Definition 2). The domain of the relationship is the 'Person' class and its range is the 'University' class, saying, that a person in the context of this ontology is a student at some university. The prefixes (like 'rdfs:') refer to the namespace[14], where the resource is defined. On the RDF application layer, RDF Description statements can now be created as shown in Figure 7.

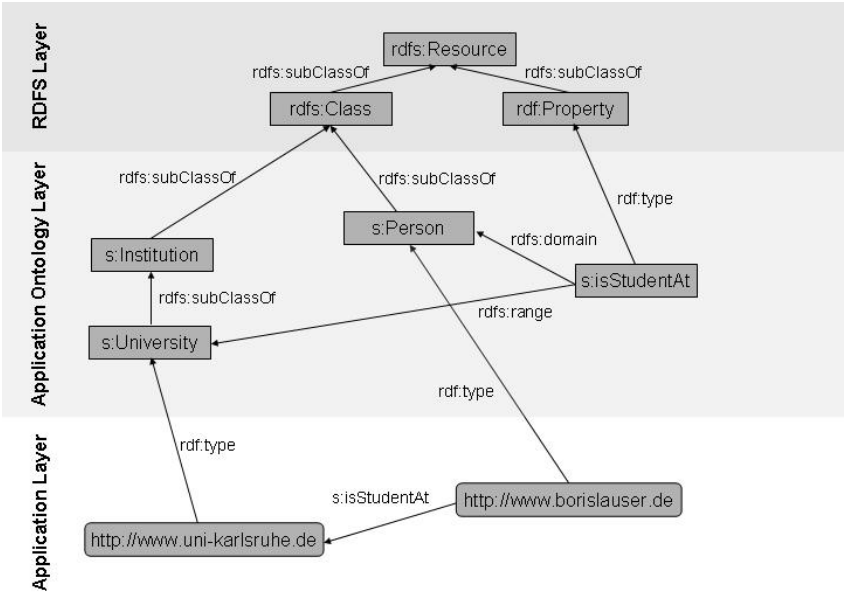


Figure 7: RDF Schema example model

[14] Namespaces are recommended by the W3C consortium in [BHL99].

RDFS provides a very basic set of modelling principles needed to represent ontologies. I will abstain from an exhaustive introduction to RDFS at this stage to focus only on the important parts regarding the work of this thesis. RDFS evolves as a standard in the context of the semantic web and other languages (as seen in the following) are extending on this base model. An XML encoding scheme for RDFS exists and is widely used to represent RDFS vocabularies. It is therefore highly interoperable and web compatible, therefore suiting well the requirements of the semantic web and the AOS project. However, some lacks in expressiveness (like symmetric or transitive properties as well as language representation capabilities) keep it from being able to fully represent above introduced conceptual ontology model. These issues will be addressed in more detail in the next section.

DAML + OIL

DAML is an RDF schema language; more precisely it is an extension of the RDFS vocabulary description language and also referred to as an ontology description language. The extension basically adds the opportunity to restrict the use of properties, for example a property can be made transitive or it can be expressed that a property of one schema is actually the same as the property of another schema. Moreover, constraints like property cardinalities or types can be modelled. A more detailed overview on this is given at [CHH+01].

OWL

The Web Ontology Language OWL is the most recent effort in ontology languages and further extends the DAML/OIL language. It is at the moment a W3C working draft and the features and details are described in [SHH02].

KAON

KAON provides another ontology language extending and building upon RDFS. The ontology definition introduced in the previous section has been taken from the KAON environment. The conceptualisation approach taken throughout the remainder of this work builds upon this model. The next section introduces the most important extensions and features of the KAON ontology language for which it has been chosen.

3.2.4 KAON

The Karlsruhe Ontology and Semantic Web Framework (KAON)[15] provides an ontology language and modelling approach extending that of RDF Schema. Moreover, it provides a whole environment of tools built upon this modelling approach. The extensions wrt plain RDFS made here are extremely important regarding the project environment discussed in chapter 2 and therefore the work of this thesis has been built upon this framework.

The ontology definitions, introduced earlier are all implemented within the KAON framework. The most important extensions provided by the KAON modelling approach regarding the requirements of the project framework are the presence of a Lexical OIModel, hence giving multi-lingual support, as well as a meta-modelling approach.

Definition 6 (Meta-concepts and Meta-properties). In order to introduce meta-concepts, the following constraint is stated: $C \cap I$ may, but does not need to be $\varnothing$. Also, $P \cap I$ may, but does not need to be $\varnothing$. The same element may be used as a concept and as an instance, or as a property and as an instance in the same OI-model.

Definition 7 (Lexical OI-model Structure). Lexical OI-model structure LOIM is a well-known OI-model with the structure matching that presented in the Figure 8.

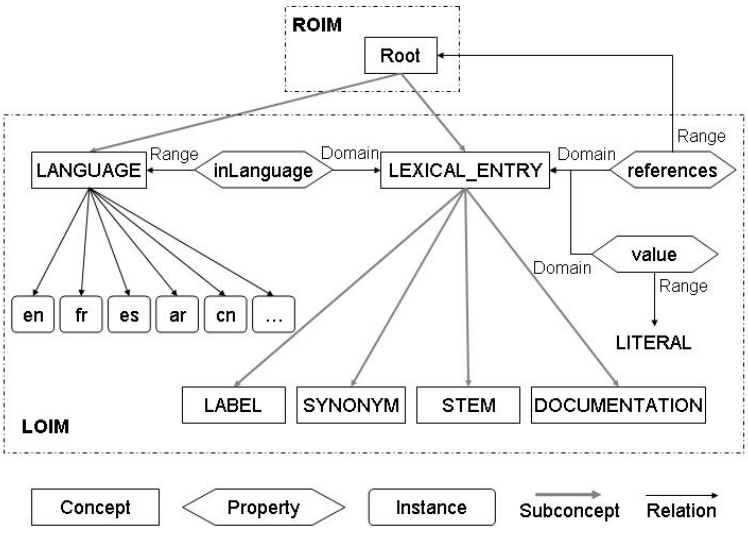


Figure 8: Lexical OIModel

Lexical entries (instances of the LEXICAL ENTRY concept) reflect various lexical properties of ontology entities, such as a label, stem or textual documentation. There is an n : m

[15] Refer to the KAON web site for details and all available resources: http://www.kaon.semanticweb.org.

relationship between lexical entries and instances, established by the property references. Thus, the same lexical entry may be associated with several elements (e.g. jaguar label may be associated with an instance representing a Jaguar car or a jaguar cat). The value of the lexical entry is given by property value, whereas the language of the value is specified using the inLanguage property. Concept LANGUAGE represents the set of all languages, and its instances are defined by the ISO standard 639. A careful reader may have noted that LOIM defines the references property to have the ROOT concept as the domain. In other words, this means that each instance of ROOT may have a lexical entry. This excludes concepts from having lexical entries - concepts are not instances, but are subclasses of root. However, it is possible to view each concept as an instance of some other concept (e.g. the ROOT concept), and thus to associate a lexical value with it. This paradigm becomes clearer in the next definition.

Definition 8 (Spanning Object). Under interpretation I, for each entity $e \in E$ the spanning object is defined as a triple $SO(e) := (C^I(e), P^I(e), I^I(e))$ that combines different interpretations of the entity e.

Consider the concept APE as an example to explain this more clearly. In this model element APE plays a dual role. Once it is treated as a concept, in which it has the semantics of a set, and one can talk about the members of the set, such as ape1. However, the same object may be treated as an instance of the (meta-)concept SPECIES, thus allowing information such as the type of food to be attached to it. Both interpretations of the element SPECIES are connected by the spanning object.

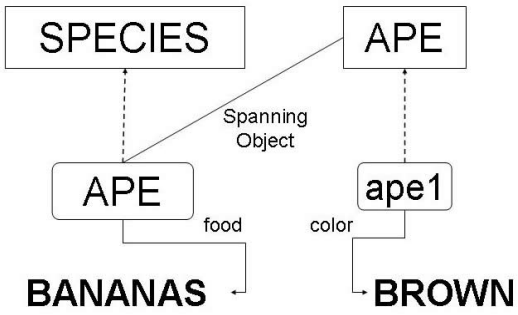


Figure 9: Spanning Object Example

In [WF94] the problems of considering concepts as instances are well explained. Wielinga et al. [WSWS01] stated the fact of not allowing concepts to be treated as instances and vice versa to be a weakness of many description-logic languages, which require strict separation.

Martin [MO97] considers class – instance flexibility as a central requirement for adequate conceptual modelling. A solution proposed in the paper is to isolate different domains of discourse. What is a concept in one domain, may become an instance in a higher-level domain. Elements from two domains are related through so called spanning objects. Our approach builds on that, however, without explicit isolation of domains of discourse. This has subtle consequences on how an OI-model should be interpreted. It is not allowed to ask: What does entity e represent in my model? Instead, one must ask a more specific question: what does e represent if it is considered either as a concept, a property or an instance. Before interpreting a model, the interpreter must filter out a particular view of the model - it is not possible to consider multiple interpretations simultaneously. However, it is possible to move from one interpretation to another - if something is viewed as a concept, it is possible to switch to a different view and to look at the same thing as an instance.

The usage of the Lexical Model and the meta-modelling approach becomes more evident in the next chapter, when the conversion of the AGROVOC thesaurus into the KAON language is discussed. KAON provides access to the conceptual model introduced here through an API. The whole system is based on Java technology and therefore platform independent. Several API's provide access to different sources of ontologies, amongst them one optimised for distributed ontology engineering on a database ontology source.

3.2.5 Ontology Engineering

Ontology engineering can be briefly described as the process of identifying and specifying the concepts that describe the target domain and establishing relationships[16] between them. The process of ontology engineering is a time consuming task carried out mainly by human beings. Until now, few domain-independent methodological approaches have been reported for building ontologies and none of them have been tested sufficiently. Most of the reported methodologies are mainly overall lifecycle models providing a more generic framework for the ontology creation process, but giving little support for the actual task of building the ontology. A comparative study of ontology building methodologies from scratch can be found in [FGPP99]. The METHONTOLOGY methodology, as described in [FBGG98] is a good example of a life cycle model. It proposes an evolving prototyping life cycle composed of development-oriented activities (requirements specification, conceptualisation of domain

[16] The term relationship will be used synonymously to the term property as defined in Definition 2 throughout the rest of this document.

knowledge, formalisation of the conceptual model in a formal language, implementation of the formal model and maintenance of implemented ontologies), support oriented activities (knowledge acquisition, documentation, evaluation, integration of other ontologies) and project management activities. The definition of ontology engineering within this work more specifically addresses the actual creation and maintenance of the ontological structure. A methodology, which adheres more closely to this definition, is the ONIONS methodology, as explained in [GSG96]. This methodology has been developed to analyse and integrate domain ontologies. 6 main phases have been suggested, guiding the process from the identification of resources to be used until the implementation and representation of an ontological structure.

Ontology engineering as defined and applied in the context of this project is the effort of combining human support and automatic tool support as provided by the KAON environment in a reusable, controlled framework to build and maintain a domain ontology structure such as introduced in the previous sections. The usage and evaluation of parts of this tool environment within the framework constitute the central part of this research work. The engineering framework proposed here depicts the development-oriented activities within the METHONTOLOGY methodology. It is a higher level framework focusing on providing a controlled flow of process steps in order to combine human and tool support in the phases of ontology acquisition, merging, refinement and evaluation. It can therefore be embedded into the broader METHONTOLOGY methodology as well as refined by more specific methodologies, focusing on more detailed parts within the here created approach. Guarino et al. in [Gua98] provide a set of methodologies for ontology-driven conceptual analysis, which could give support at different stages. An overview of these methodologies can be accessed through his web site. Being not the main focus of this work, I will not further explore on such specific methodologies.

The next chapter presents the ontology engineering framework and its prototype application to create a domain ontology. The specific project and the requirements regarding the overall environment introduced in Chapter 2 will be described first.

4 Introduction of ontology based information management system at the FAO

As I outlined in the previous chapters the Semantic Web promises opportunities for better knowledge organisation and retrieval, enhanced search capabilities and resource identification across several domains, provided that the data is presented and published in the right format, i.e. metadata about resources in a machine-readable format. The AOS, as introduced in chapter 2 strives to facilitate the integration of resources across a variety of agricultural domains by creating description vocabularies to describe these resources, facilitate easy access to them and ensure better indexing and searching. Domain ontologies constitute the basis for describing the resources in the different domains. A reusable framework is necessary to create domain ontologies in different domains accompanied by tools, using them for resource description. The formulation of such an ontology engineering framework, as well as the evaluation of tools, used within this framework, constitutes the central part of this thesis. The framework has been applied in a prototype project, which will be introduced in the following section. Afterwards, I will present the entire framework and its application within the project in detail. The adaptation of an ontology browser to serve the requirements of the project and supporting part of the framework will be discussed at the end of this chapter.

4.1 The prototype project

The prototype project, within which context the framework has been established and applied, is the creation of an Internet Portal on Food Safety, Animal and Plant Health (IP-FoAPH)[17]. The portal is an access point for official national and international information relating to bioprotection, the risks associated with agriculture (including fisheries and forestry) and food production, whether these risks affect the safety or wholesomeness of food; arise from the introduction of new technologies; or are caused by plant and animal pests or diseases, or by zoonoses.

The portal contains official information of interest to all national and international agencies responsible for managing biosecurity risks, from trade or customs regulation at points of entry to a country to broadly-based multidisciplinary agencies with a remit to cover all aspects of potential risk. It is also used by producers, or companies trading agricultural or food products.

Any interested user can access the portal through the Internet. In addition, certain nationally nominated users are allowed to log in to the system, and have the right to update

[17] Refer to [http://193.43.36.96/cds_biosec_test/Biosec/En/default.htm, Dec 2002] for a first draft.

the data contained in the portal by adding documents or URLs, which point to other web sites outside the portal. Logged-in users may also have the right to view restricted items of data where these have been defined in conjunction with the portal administrator (FAO).

Given this context, we can now derive certain requirements regarding the AOS and its support towards this portal. This is done in the following section.

4.2 Requirements regarding the AOS

Given the environment described before and the generic purpose, which, up to now, has still not been exhaustively defined, the users of the portal can be divided into two major groups:

- **Information searchers**
  Users, browsing the portal to find information of interest
- **Indexers**
  Users who update the portal with information, which means indexing documents with keywords taken from a controlled vocabulary, in this case the Ontology on Food Safety, Animal and Plant Health.

The following use cases have been established for each group:

Use Case 1: Document Indexing

The portal provides a separate section, where document indexers and information updaters of the portal can enter with their respective login. This section therefore is not open to the public, but only to authorised indexers. The indexer is given a screen, where he can enter metadata about a document, journal, article, etc. and has also the possibility to upload an electronic version of the respective file. One part of the document metadata consists of keywords describing the content of the document. These keywords are taken from the Ontology on Food Safety, Animal and Plant Health to assure consistency. The indexer is not necessarily a subject specialist in the area in which he indexes a document. Therefore, he needs guidance finding the right concepts to describe the resource. Two possible applications could be used to assist the indexer in his task. The first one is an ontology browser where the user can browse and search the ontology in his language (the language of the document, he wants to insert into the portal) as well as mark the terms, he wants to index the document with. If he decides not to take an already marked term, he can delete it again from the list of chosen terms. This assures that no arbitrary terms will be used for indexing a document. The second possible

form of assistance would be an automatic text classification tool, suggesting the user concepts taken from the ontology, which most likely describe the documents. This could be especially useful for non-expert indexers not familiar with the indexing vocabulary in order to provide them with initial ideas.

Use Case 2: Keyword searching of information on the portal

The second use case focuses on the public user, searching any information of interest on the portal. The user can perform a full text search or a keyword search over all the documents and information in the portal. In case of a keyword search, the keywords used for the search should be taken from the Ontology on Food Safety, Animal and Plant Health, This is because the keyword search only searches the metadata of the documents and they have been indexed with the terms from the ontology. In this case, the user will be able to compile a search query from the ontology, rather than typing arbitrary terms into the search box. An ontology browser, where he can search and browse the ontology for the keywords he wants to search for gives the necessary support. The user can mark terms, he wants to include in his search (and also delete them again from the list, if deciding so) and by closing the browser window, the search terms will show up in the keyword search box. After issuing the search, all information in the portal, indexed with those keywords will be returned.

Within the scope of the AOS project, we can therefore identify the following deliverables to meet the above requirements:

1. An ontology on food safety, animal and plant health
2. An ontology browser to browse and search the ontology and compile index or search queries
3. An automatic text classification application, supporting the indexer with suggestions on the index terms

The ontology shall cover all subject areas represented in the portal and be extensive enough to describe the types of documents and information, which shall be available through the portal. Subject specialist knowledge as well as already existing resources in the form of documents, keyword collections or already existing vocabularies shall be exploited for creation.

The ontology browser shall interface with the Internet portal to be built. The portal is being built on top of the already existing Community Directory Service (CDS) system. The CDS

provides the architecture and functionality to store, index and retrieve information to be shown in the portal. The system maintains the database, where metadata about the indexed documents are stored as well as the document repository with the electronic versions of such documents.

The automatic text classification tool should plug in between the CDS system and the ontology browsing application, first suggesting the indexer a list of possible concepts, which he can then refine with the ontology browsing application.

In the remainder of this chapter, the overall framework for the creation of a domain specific ontology will be introduced first, together with its application to the prototype project to create the above required ontology. Later, the adaptation of an already existing ontology browser to be incorporated into the portal according to above described requirements and use cases will be presented.

At the time of the assembling of this thesis, automatic text classification within the environment used here is not yet ready to be incorporated into an application: there is still an extensive need for evaluation and research on it. Therefore, automatic text classification has not yet been included to support the indexing task at this time. A detailed discussion on the adaptation and evaluation of an already existing text classifier and its potential use follows in chapter 6.

4.3 Ontology Engineering Framework

4.3.1 Overview

The ontology engineering methodology applied within the above project prototype, to create the domain ontology on Food Safety, Animal and Plant Health, consists of five basic phases, which can be reapplied in iterative steps:

- **Resource Selection**
  The identification of resources for ontology development, i.e. documents of the domain, existing vocabularies, domain specialists and alike.
- **Semiautomatic ontology acquisition**
  Two different approaches, a manually created core ontology and the reuse and extraction from existing vocabularies are addressed in this phase
- **Merging of ontologies**
  The Merging of the above created ontologies using automatic support.

- **Extension and Refinement**

  The extension and refinement of the merged ontology by subject specialist assessment

- **Evaluation**

  The evaluation by users of the ontology.

In figure 10, this process is represented graphically.

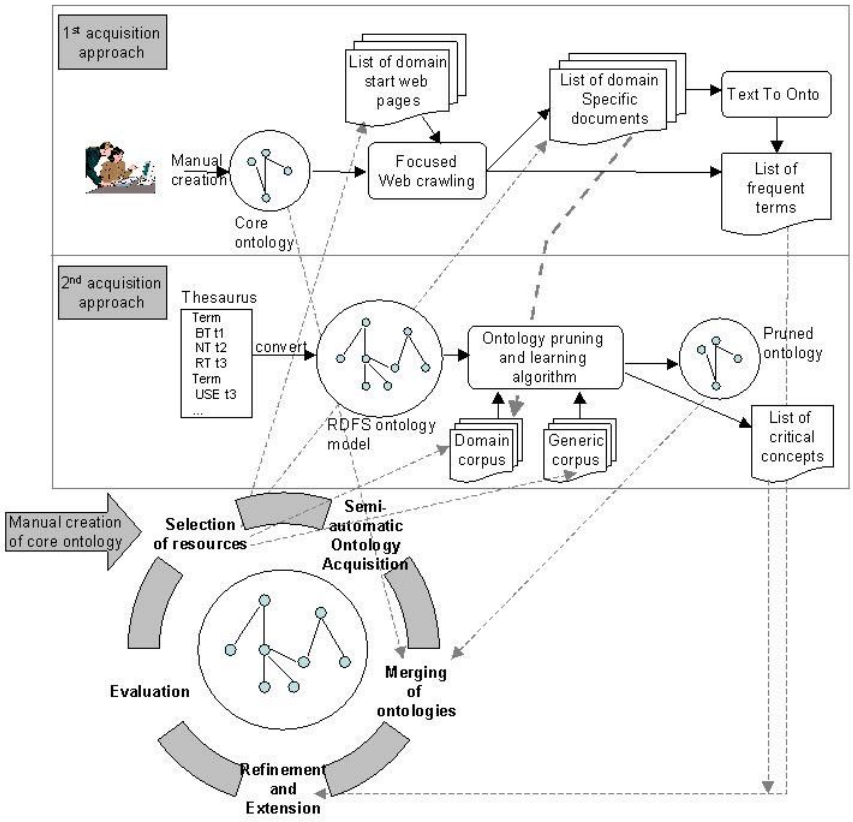


Figure 10: The ontology engineering framework

4.3.2 Initialisation of the cycle

If building an ontology from scratch, the whole process is initiated with the manual creation of a core ontology. This step already done in combination with resource selection, since parts of the resources such as subject experts and documents are needed to carry out this task. First the goal of the ontology regarding its intended use has to be specified and clearly stated. This determines the level of specificity and detail needed in the domain ontology.

Subject experts of the target domain then use several brainstorming sessions in order to initially identify the top-level main areas and concepts of the domain. This is considered to be a good initial approach for several reasons. First, it creates a rather unbiased starting point:

tool support and automatic extractions risk otherwise being channelled into one specific direction, leaving important aspects untouched. Unbiased brainstorming helps to first discover each individual's human knowledge resources to be then enhanced and triggered by tool support in later steps. Second, it helps untrained subject specialists to become familiar with the notions of ontology and third, it gives a good understanding of what shared conceptualisation means: to commonly agree on the formal specification being designed. In this prototype project, 3 food safety specialists extracted the major concepts in their area using their own subject knowledge supported by the Codex Alimentarius[18], a reference for food standards in food safety biosecurity. In further brainstorming sessions, they identified the main relationships linking these initial concepts.

Here and in further expert subject assessment throughout the process, other methodologies can be applied to address such issues as disambiguation, clarity, unity and rigidity as explained in [GW00] and [WG01]. Being not the main focus of this research work, I will not go into further detail of such methodologies here. It is even questionable if such rather complex methodologies are applicable with a reasonable time and cost effort in an environment, where deadlines have to be met.

The initial core ontology design thus created has to be captured in a machine-processable way. This is accomplished throughout the whole process, using the OIModeler, a graphical ontology editor that is part of the KAON environment. The tool supports the distributed engineering of ontologies by providing concurrent access on a database ontology representation. The ontologies are represented using the conceptual model defined in the previous chapter and can be imported from and exported to RDFS format. The tool is multilingual and supports all official FAO languages, hence meeting all requirements of an ontology editor needed in the context of this project.

The application of this initialisation in the prototype project resulted in a core ontology on food safety consisting of 102 concepts, 18 meta-properties and 91 property instances. I will introduce the modelling approach for properties, which has been used here in more detail later in section 4.5, when I present the conversion of the AGROVOC thesaurus into the KAON language. An extract of the RDFS representation of the initial core ontology is attached in Appendix A.

In the following, each successive phase of the ontology engineering cycle will be explained in detail.

[18] [http://www.codexalimentarius.net/, July 2002].

4.3.3 The 5 phases of the framework

1. Selection of resources:

Each cycle starts with the selection of the resources needed within a development cycle. Subject experts and ontology engineers have to be identified and allocated, so that the target domain is represented by the subject experts and an ontology engineer represents and co-ordinates the tool supported steps throughout the whole process. A second type of resources to be selected is documents describing the target domain. On the one hand, the subject experts will consult them as a source for brainstorming, refinement and extension. On the other, a set of electronically available documents has to be compiled out of the documents, covering all aspects of the domain to an approximately equal weight. Another set of generic documents - not representing knowledge of the target domain - has to be determined in this phase. These document sets will serve as input to several computer-supported steps as described later. I will explain more about both these document sets in chapter 5, when the most critical (regarding the choice of documents) usage application is discussed. Finally, already existing ontologies or controlled vocabularies containing conceptualisations of the target domain are valuable resources and have to be identified in this first step. They will serve as input in the following step of semi-automatic ontology acquisition. This process consists of two different approaches, explained in the following two sections:

2. Semiautomatic ontology acquisition

1st Acquisition approach: Manual creation of the core ontology

The input to this phase in each development cycle is the current version of the ontology, which is the brainstormed initial ontology in the first instance and successive ontology versions in further iterations. This ontology is then fed into a focused web crawler, developed within the KAON environment, explained and evaluated in detail in [Ehr02]. The Crawler takes a set of start URLs and domain ontology. It then crawls the web in search of other domain specific documents based on a large set of user specified parameters. The start URLs, consisting of well known main sites existing in the target domain, have been identified in the first step (Selection of Resources). The outcome of the crawling process, consists of a rated list of found domain specific documents and links as well as a list of most frequent terms found on these documents. The list of documents has to be assessed by subject specialists and can be used to revise the initially compiled document corpus. This document corpus can now be input into an automatic concept extractor. Text-to-onto is a component of the OIModeler

ontology editor and extracts concepts from a text corpus based on frequency computations. This is explained in more detail in [MS00]. The output is a list of frequent terms that can be used to directly extend the ontology. Only subject experts can accomplish this step, since the lists are only outputs based on frequency heuristics and still need assessment and evaluation.

The core ontology resulting from the initialisation step has been applied to the Focused Web Crawler in the first application of the framework. The result of this was a list of 257 web pages, which have been assessed by the 3 subject experts and grouped into 8 main sites. Figure 11 shows this step and the main groups. The extensive list of all the retrieved web sites together with their score is attached in Appendix B for completeness.

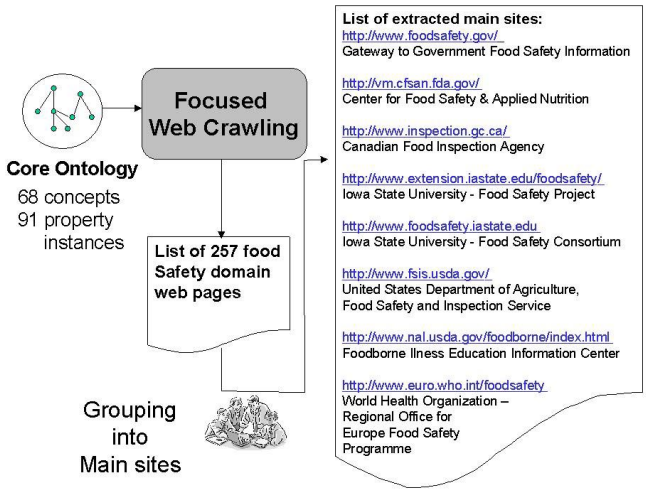


Figure 11: The Focused Web Crawler

Time restrictions and deadlines in the first application of the engineering cycle made it impossible to compile an extensive document set at this time. Some Codex Alimentarius documents and some documents from the 8 main sites identified by the crawling step have been input into Text-To-Onto to extract a list of 1632 frequent terms. This list will be used in later in the refinement step.

Since this has already been done elsewhere, detailed evaluations of the Focused Crawler and Text-To-Onto are not carried out in the context of this work.

2nd Acquisition approach: Pruning of already existing vocabularies

The second acquisition approach is completely automatic and only needs ontology engineer support. The input to the second step is an already existing vocabulary (ontology, thesaurus), presumably containing parts of conceptualisations of the target domain, which can be reused here. If the vocabulary is not already a KAON ontology, it first has to be converted into this

format. In this project, the AGROVOC thesaurus has been chosen as input resource. Since the representation of the AGROVOC as a KAON ontology is not only used for this step but also serves as a resource for the automatic text classification algorithm discussed later in chapter 6, the conversion will be discussed in more detail in a separate section later in this chapter. The aim of this acquisition step is to automatically extract possibly the whole subset of this ontology structure, relevant for the target domain. An ontology pruner, developed in [Volz00] and formerly applied in [KVM00] has been extended and adapted to accomplish this task. The approach is a heuristic, for which reason a detailed and extensive evaluation of the algorithm has been carried out. The algorithm, its application and evaluation will be discussed in detail in chapter 5. The input to the pruner is the converted vocabulary to be reused together with two sets of documents, a domain specific one and a generic one, both explained in detail in chapter 5. The output of this step is a subset of the initial ontology, i.e. the input ontology pruned to contain only the concepts relevant for this domain. The result has to be assessed and validated by subject experts and combined with the core ontology. This is done in the next phase of the engineering cycle.

In the prototype project the not yet adapted version of the pruner has been applied, again due to time restrictions and the need for some fast results. The set of domain specific documents used for Text-To-Onto in the previous step has been reused here. An ontology with 504 concepts has been extracted from the AGROVOC in this application. Since an extensive evaluation of the algorithm follows in Chapter 5, I will not go into more detail with this prototype result. The evaluation presented in chapter 5 is actually the main part of the second application of the engineering cycle.

3. Merging of ontologies

The above acquisition steps have created two ontologies, the manually created core ontology and the derived ontology, using thesaurus terms. These have to be assembled into a single ontology. Ontology merging is still more of an art than a well-defined and established process. Gangemi et al. describe a methodology for ontology merging and integration in the Fishery Domain in [Gan00]. Here, we use the opportunity given by the modular structure of the KAON conceptual model to merge the two ontologies or OIModels referring Definition 1 in chapter 3. The pruned OIModel is an included OIModel of the core model. Before this inclusion, however, the pruned OIModel has to be assessed by subject experts for usability. All concepts, which the ontology pruning algorithm mistakenly left in, should be deleted before the merging of the two ontologies. Merging itself can be accomplished simply by

including the assessed and cleaned up OIModel into the core ontology OIModel using the OIModeler of the KAON tool environment.

In the prototype, 23 concepts have been taken from the pruner output, only considering the most important ones at this stage for presentation purposes.

The merged ontology thus created is, however, not connected and linked yet, neither is it sure that the structures and hierarchies of the two parts are compatible. These issues are addressed in the next phase of the engineering cycle.

4. Refinements and Extension

The refinement step completes and extends the merging step in the sense that the merged ontology structure has to be extensively assessed by subject experts in order to resolve the following issues:

- Duplicate concepts have to be identified and resolved, i.e. two concepts, which are present both in the core ontology and in the included pruned ontology, have to be reduced to one and the relationships have to be resolved. Gangemi et al. use a different approach here. In their fishery ontology project ([Gan02]), when including different already existing vocabularies in the area, they keep them the way they are and provide the different views of a concept according to the source it has been defined in. Creating a specific domain ontology (and given the requirements that this ontology will also be used for indexing purposes), we want to provide one single view of each concept determining its definition within this specific domain.
- After the inclusion of the pruned ontology, all top-level concepts of the included ontology are connected directly to the Root of the OIModel. This hierarchy does not necessarily reflect the hierarchy of the core ontology and has to be assessed by subject experts. The concepts might be rearranged under the identified main concepts of the core ontology or new main concepts have to be introduced in order to properly incorporate the included ontology.
- Non-hierarchical relationships have to be established in order to connect the included ontology with the core ontology cross-hierarchically.

Upon completion of these steps, the pruned ontology is now fully integrated into the core ontology. Now the resulting ontology can still be extended using the list of frequent terms output during the first acquisition approach (web-crawling step and text-to-onto step). The

terms identified here have to be assessed for usability and arranged into the hierarchy of the ontology structure. Again, possibly new concepts have to be created in order to bridge gaps between newly included concepts and the already existing structure. As in the last step, non-hierarchical relationships have to be established between these newly included concepts and the already existing ones from the initial core ontology and the included one. The concepts included in this step extend the core OIModel respectively, in order to be able to always unambiguously identify the source of the concepts of the overall ontology structure (i.e. the pruned AGROVOC OIModel will not be changed: instead, all changes are applied in the core OIModel). More about this modelling approach is described in [MMV02].

The prototype has been extended with 12 more concepts from the list of frequent terms. The concepts have been linked applying 92 property instances, resulting in a final prototype of 102 concepts and 183 property instances.

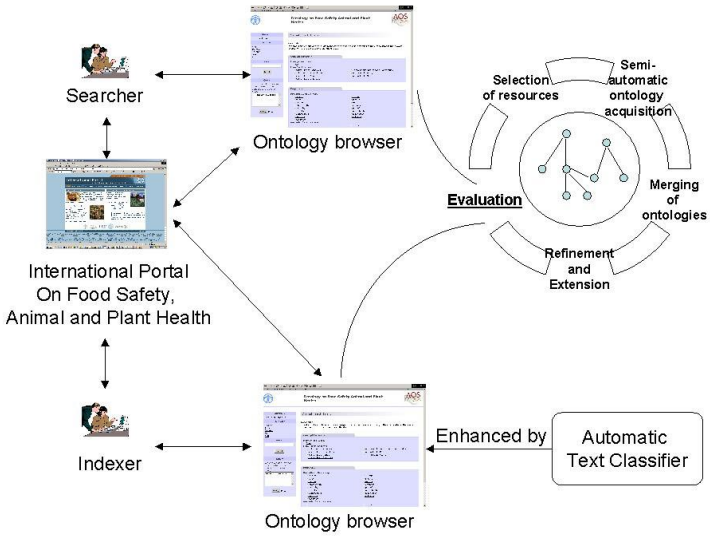


Figure 12: Evaluation of the ontology

5. Evaluation

The ontology thus created is now made subject to testing and evaluation by a broader audience. As discussed in the prior section, users can be split generally into indexers and information searchers. These serve as a good audience for evaluation, representing various levels of knowledge and different needs and ways for accessing the ontology. This framework does not provide guidelines or procedures for testing and evaluation. The focus within this project was to provide the means for accomplishing this task. A generic ontology web browser, providing easy access to all features of the ontology serves this purpose specifically supports the reusability of the framework, since the browser can be reused for the same purpose in other projects. Figure 12 shows the possibilities for evaluation given within this

framework. The incorporation of the automatic text classifier is only a suggestion at this stage, as mentioned before and has not been accomplished within this project.

The adaptation of an already existing ontology browser to meet the requirements discussed above, and its possible incorporation into the IP-FsAPH will be discussed in the next section.

4.4 The Ontology Browser

Figure 13 shows the possible communication between the IP-FsAPH and the ontology browsing application. As described in the above use cases, the ontology browser can be an external application to be called from within the portal system. The left half of the picture shows the CDS system, on which the IP-FsAPH portal will be built/will operate. In the section to the right, the ontology system is shown. The search or indexing application of the portal (or any other application in need of querying a String of terms retrieved by browsing the ontology) sends the following information to the portal:

- The current language(s) as a String (en – English; fr – French; es – Spanish; ar – Arabic; zh – Chinese)
- An initial search String (for example "risk").

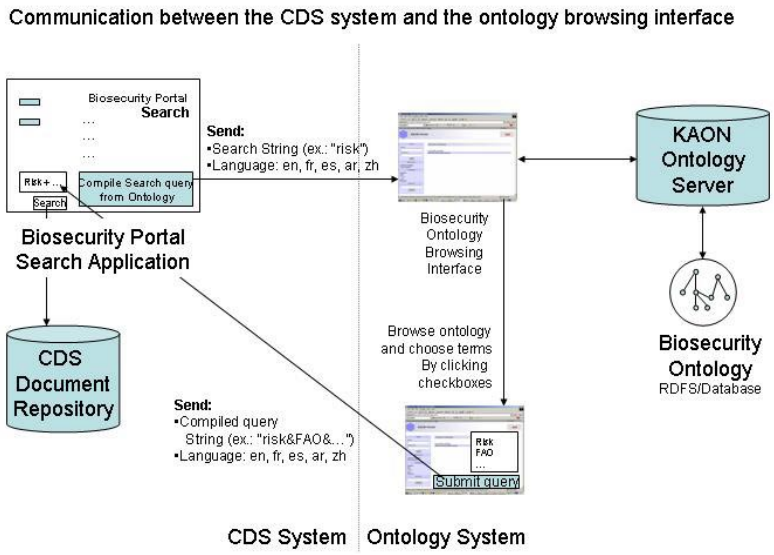


Figure 13: Communication between the CDS system and the ontology browsing interface

The interfacing ontology browser is an adapted and extended version of the KAON portal developed within the KAON environment[19]. The ontology is accessed through the KAON

[19] Refer to [http://kaon.semanticweb.org] for more details and a free version of the KAON Portal

API either from a database or an RDFS file. For more details on the overall architecture refer to the developer guide[20] on the KAON web site.

When called with the information above, a new browser session is invoked. The user is then able to browse and search the whole ontology structure in all available languages. At the current stage of the project the parameter handling is not implemented, since it is subject to application specific interfacing and therefore not part of the framework. The KAON Portal has been extended, however, with the functionality to mark terms while browsing the ontology. The user can browse the ontology in all the five FAO languages and compile a query string consisting of an arbitrary number of marked entities of the ontology. The lexicalizations of the entities are made visible to the user. The browsing session stores the entity URIs together with the lexicalizations (labels in the currently active language respectively) to be used by and backed up into the calling application (which would be the CDS system in this case). Figure 15 shows a screenshot of the adjusted KAON Portal for the OFsAPH.

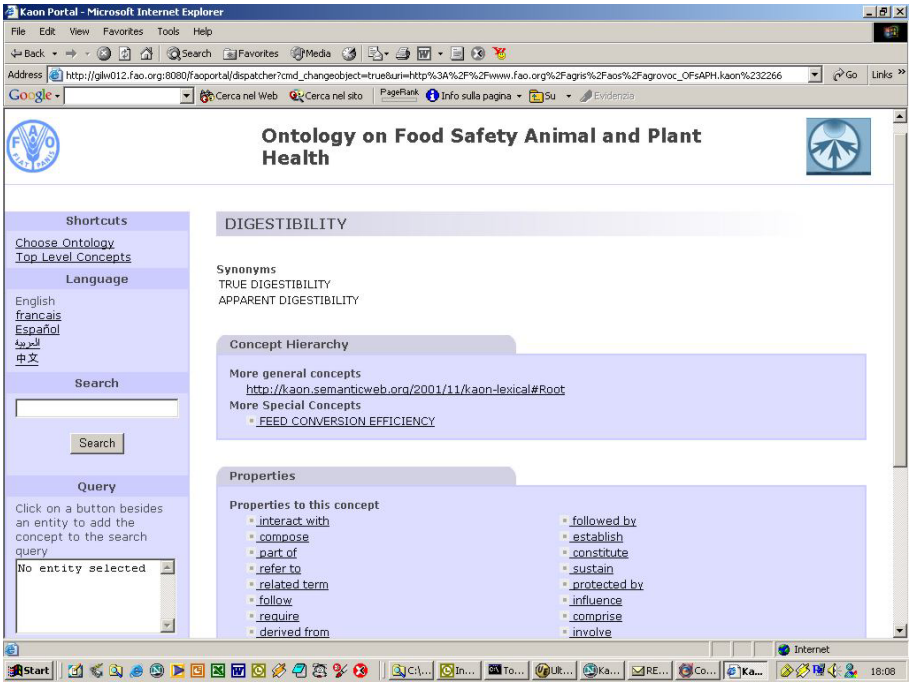


Figure 14: Screenshot of the adapted KAON portal

In the following section, I will explain in detail the conversion of the AGROVOC thesaurus into the KAON ontology structure, setting the basis for the evaluations in the remainder of this thesis. The presented modelling approach has also been applied to the food safety ontology.

[20] Freely available at [http://kaon.semanticweb.org/Members/rvo/KAON_Dev_Guide.pdf].

4.5 Representation of AGROVOC in KAON

Conversion of a thesaurus or another already existing vocabulary into an ontology is still more of an art than a well-defined process and has to be carefully considered in each specific case. One approach mapping an Art and Architecture Thesaurus to an ontology using Protégé[21], which is similar to the here chosen one is described in Wielinga et al. [WSWS01]. Another interesting methodology for integrating ontologies and thesauri to build RDFS schemas is given in [AF99]. The methodology discussed there is platform independent and describes a process of several steps on how to map thesaurus term to an ontology. In this approach, an ontology structure of super-concepts is first created, followed by the mapping of the thesaurus terms to fit into this high level structure of top concepts.

Here a more straightforward approach has been chosen, mapping terms directly to concepts and leaving the thesaurus hierarchy structure as is in the first instance. The resolving of this and establishment and restructuring using super-concepts is left for further assessment as explained before in the engineering framework.

The first problem in converting a thesaurus arises with the question of what is a concept. As described before in chapter 2, AGROVOC is a collection of terms, each term being either a descriptor or a non-descriptor. The reason for this distinction comes from the purpose of a thesaurus to be used solely for indexing purposes. An ontology, however, is somewhat more generic and can be used in different ways, therefore not reflecting this very specific differentiation. One possible solution would be to simply have a concept for each keyword and link the former non-descriptors with the same relations (use and used for relation) to the remainder of the mapped concepts. The disadvantage of this approach is, however, that in the core ontology these relationships are unlikely to exist and to be used, since the KAON ontology model provides different modelling constructs of reflecting such relationships. Basically, a descriptor and a non-descriptor together refer to the same concept, deriving from the indexing rule to use a descriptor instead of a non-descriptor for indexing. The synonym concept of the KAON Lexical OIModel reflects such a relationship. Experience has shown that a non-descriptor in most cases can be seen as a synonym of its descriptor. The option chosen here is therefore to map all descriptor terms to a concept with a label in each language provided in the AGROVOC. Each non-descriptor has been mapped to synonyms attached to its mapped descriptor concept. For each translation available in the AGROVOC, a lexical

entry (synonym or label) is attached to the concept. This cannot be as easily accomplished for the 'used for in combination with' related descriptors and non-descriptors, since neither of the two non-descriptors can be directly seen as a synonym of the descriptor. These non-descriptors have been mapped to concepts, linking to their respective descriptor concepts using the same relationships (use and used for in combination), and herewith not directly in the hierarchy structure as discussed in the following. The resulting AGROVOC ontology structure consists of 17506 concepts of which 899 result from this non-descriptor mapping. This minor number of concepts is subject for later expert assessment to be resolved in future refinement steps.

AGROVOC contains scope note references for terms such as definitions, comments and descriptions. These have simply been converted to a KAON Description of the Lexical OIModel in the respective languages and attached to the respective concept.

The next problem addresses the mapping of the hierarchy. Broader Term and Narrower Term relationships which constitute the hierarchy of a thesaurus not necessarily adhere to sub-class, super-class relationships as being required in ontologies. Being a sub-class of another class is a stronger type of relationship, since a sub-concept inherits all properties from its super-concept, whereas narrower term only says that a term is more specific than its broader term. In the case of AGROVOC and given the type of domain ontology to be built, most of the broader term – narrower term relationships however meet the more restrictive requirements and therefore this mapping has been chosen to build the super-class sub-class hierarchy H^c of the model.

Finally, the issue of modelling the thesaurus relationships in the ontology structure has to be considered. In case of the AGROVOC conversion, several options have been considered:

1. **Create properties on concept level for each relationship:**
   In this option, each pair of concepts related by for example a related term relationship will be related by a new relationship extending the set P of relationships of the ontology structure. The problem with this approach is the creation of much duplication of relationships. The most applied relationship is the related term relationship. Using this

[21] Protégé is an ontology environment project similar to the KAON approach initiated at Stanford University. Refer to [http://protege.stanford.edu] for more details.

approach, P would have lots of properties, all with the label 'related term' and semantically having the same meaning, however having all different URIs.

2. **Create super-class 'keyword' and create instances of it to map all keywords:**
   This approach would affect the whole modelling of the AGROVOC ontology described above. Here, an ontology structure based on the purpose of indexing and describing documents and resources would be constructed. Each descriptor of the AGROVOC thesaurus would be modelled as an instance of the generic concept 'keyword'. This concept represents both a relationship's domain and range. Instances are linked by instantiating these relationships according to the AGROVOC structure. This model could be included into broader resource description ontologies, for example to the subject field of the Dublin Core[22] element set. This would however also imply to model the broader term – narrower term relationships in the same way. A big part of the semantic power of the ontology would therefore be lost by using this modelling approach, reason for which it has been discarded.

3. **Creation of meta-properties:**
   The third modelling option, which has also been chosen in case of the AGROVOC conversion, makes use of the meta modelling feature of the KAON language and the Spanning object paradigm introduced in the previous chapter. All relationships of the AGROVOC are mapped to properties having the Root node as domain and range concept. This approach creates only a single property in the set P for each relationship and the properties can be instantiated by every concept. The instantiation of such a meta property is accomplished relating two concepts' spanning instances, making use of the two different views of a concept. This modelling approach has moreover been used throughout the creation of this domain ontology, i.e. also in the creation of the core ontology structure.

Figure 15 conceptually shows an example extract of the final mapping of the AGROVOC. The right side shows the mapped ontology structure. The dashed lines from the Root node mean that the concepts are connected to the Root through other concepts further up in the hierarchy left out here for simplification. The left side with the dark grey shaded concepts

[22] See http://www.dc.org. The Dublin Core element set is often referred to as ontology.

shows the concept and hierarchical view of the ontology structure, whereas the right side shows the instance view. It becomes clear that the concept created for a 'used for in combination with' (uf+) – relationship is out of the hierarchy, only connected instantiating the use and uf+ relationships on its spanning instance. The connection to 'Bacterial Pesticides' has been left out for simplicity of presentation.

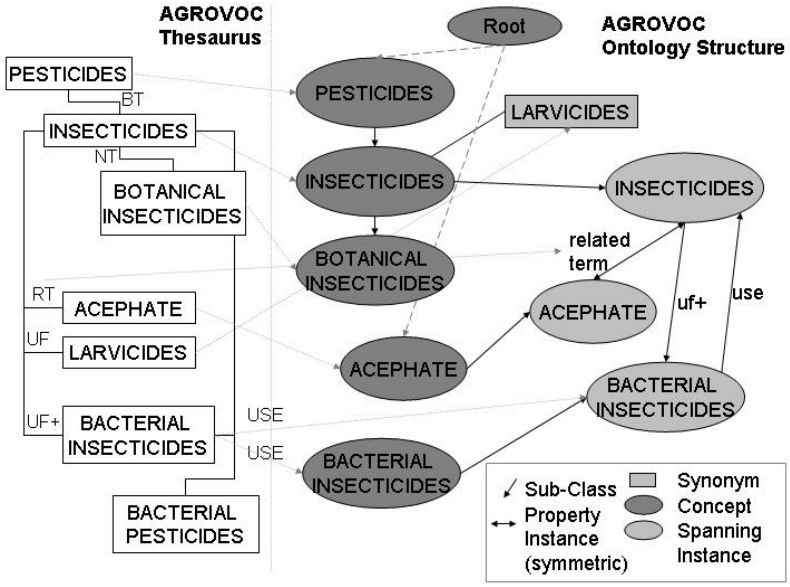


Figure 15: Mapping of AGROVOC thesaurus to ontology structure

As mentioned in chapter 2, the AGROVOC terms are further mapped to categories. At the current stage, the mapping of terms to categories is not finished completely yet. For this reason, the modelling of the categories has been done but not applied in the current version of the converted AGROVOC yet. Figure 16 shows the category modelling approach chosen here. The categories are separated into an extra OIModel, which can be included into the AGROVOC OIModel in order to make them reusable in other possible models. All categories listed in Appendix C are modelled instances of the Category concept. Each of the 115 subject category instances is connected to its respective main category (the matching is determined by the preceding capital letter, i.e. A01 is a subcategory of A) instantiating the subCategoryOf property. The assignment of a category to a concept is modelled in the same way as the AGROVOC relationships; hence a concept is assigned a category by instantiating the category property to relate its spanning instance to the respective category. The only difference is the range of the category property, which is the concept 'Category' of the Category OIModel. An RDFS excerpt of the category OIModel is attached in Appendix C.

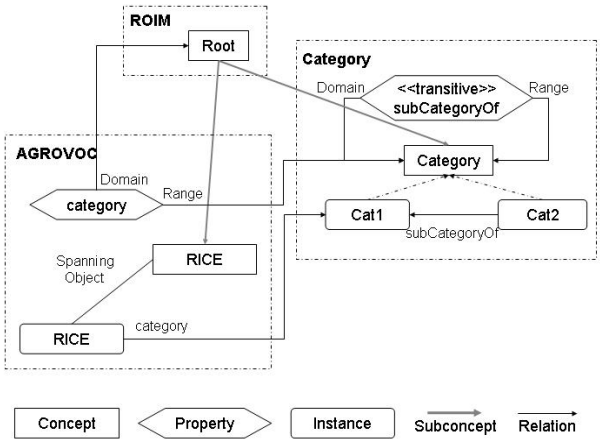


Figure 16: Modelling of AGROVOC categories

The multilingual issues discussed in chapter 2, i.e. the fact that mere translations of concepts are sometimes not possible and that another mapping would be necessary to resolve these issues has not been considered in this conversion. The possibility of taking these issues into account is currently not given, since they have not been considered in translation of the AGROVOC, which is the only source at this moment. Resolving these issues would entail extensive multilingual expert assessment: this is subject to future assessment and is not the main focus of this thesis. Translation efforts of newly created ontologies using the above framework should, however, consider such issues and make use of extended modelling opportunities provided in an ontology.

4.6 Related Work and positioning:

Various other ontology engineering frameworks exist, which I will briefly address in the following.

An ontology engineering effort also established within the context of the AOS suggests a formal ontological framework for semantic interoperability in the fishery domain and is discussed in [Gan02]. Here the methodology focuses mainly on conceptual reengineering, integration and merging of existing resources in the fishery domain. A detailed explanation of the different understandings and the here used view of the terms integration and merging is given in [PGM99]. The basis in this approach is the foundation ontology DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) described in [GGM+02]. This ontology is a very high level, generic ontology, containing concepts like perdurant or endurant and a set of high level relations, roles and descriptions. A core ontology is then built, inheriting from this foundation ontology and integrating different existing sources (thesauri,

DTDs, reference tables, etc.). In the merging process, the ONIONS methodology as introduced in [GSG96] and also briefly described in [PGM99] provides guidelines to analyse and merge existing ontologies, and addresses the reengineering of domain terminologies. This methodology and engineering approach is highly complex and focuses rather on problems of ontology acquisition itself than ontology representation. The whole approach is built in Loom [Bri93], a complex knowledge representation system using logic-based representation language and a built in inference engine supporting above methodologies. At this point no converting mechanism from Loom internal representation of ontologies to RDFS or OWL format exists. Therefore, web usability is limited at this point. The problem with this whole approach, i.e. using loom together with all the above mentioned methodologies is its high expressive power with no means for a regular non-expert in this area to understand and use it. A steep learning curve is needed and the overall process is extremely time consuming. In an environment with tight time and financial restrictions as given in the case of AOS and the FAO, this is rather unlikely to be applied on a large scale.

On-To-Knowledge is a project in the Information Society Technologies (IST) Program for Research, Technology Development & Demonstration under the 5th Framework Program of the European Community. The project targets are similar to the ones of the AOS project, in that it aims to maintain and organise knowledge in large organisations and provide intelligent search mechanisms. This shall be achieved by the development and application of an extensive took suite, accompanying and supporting each step within an ontology engineering life cycle. The tool environment and the architecture are briefly explained in [Fen00]. An ontology engineering methodology and life cycle presented within this project[23] consists of 5 phases: feasibility study, ontology kick-off, refinement, evaluation, refinement and evolution. All these phases are augmented by semi-automatic tool support. In order to create ontologies, text mining and extraction techniques are applied here as well. The core tool of the ontology building and engineering is OntoEdit [SSA02], an ontology editor and engineering tool based on Frame-Logic. It focuses on three steps for ontology development: requirements specification, refinement and evaluation. Each step is supported by extensive plug-ins to the tool, such as OntoKick, which enhances the requirements phase of the engineering by using competency questions [UK95] to define requirements for the ontology. Besides that, it provides a powerful inference engine for evaluation purposes. The here presented framework

[23] KMMethodology as presented on [http://www.ontoknowledge.org/downl/otk.presentation.ppt, Dec 2002].

approach is less developed and there is much less automatic integration of supporting methodologies. However, as opposed to OntoEdit, the here used framework is based on freely available open source software, completely Java based and therefore highly portable in a distributed system. It provides multilingual support in all 5 official FAO languages, extendable to cover each other possible language encoding. The OIModeler editor used within the framework provides an easy to use graph based interface, and supports concurrent ontology engineering. These facts seem to fit better the needs of the FAO and the AOS, where in a financially restricted environment subject experts of various areas with different technical backgrounds in different locations with unspecified systems will work on engineering ontologies.

The GETESS project[24] (introduced in [SBD+99]) aims on facilitating natural language query searches on available knowledge and establishing user dialogues. For resolving and enhancing these queries, an F-Logic ontology component is part of the system. The here used domain ontologies have been in a semi-automatic extraction framework using the natural language processing component SMES and the engineering environment OntoEdit and is described in [MNS02]. The approach is similar, to the here introduced framework in that it combines human experts and semi-automatic text extraction tools in order to create a domain ontology in iterative steps. In an online demo version, the entered search query is parsed and run against the ontology and the retrieved search results are augmented with relationships found in the ontology, giving the user the option to refine the query. The ontologies used here are not very detailed at this stage giving rather few refinement options. At this stage, there is no such application developed yet within the here presented framework. Given the depth and high degree of detail of the to be developed Biosecurity ontology, however, it will be interesting to evaluate in the future the integration of this ontology to provide such augmented searches.

4.7 Current status and Further Work:

At the stage of the assembling of this thesis, the project is currently in the second application cycle of the framework introduced in this section. This second cycle has been initiated with the prototype core ontology assembled and refined as discussed before. Resource selection has been reapplied, extending the set of subject experts to a number of 5 in

total, this time representing also the areas of animal health and plant health. Especially the recompilation of a profound domain specific document set has been considered in this second iteration. The new compilation, as well as the reapplication of the thesaurus pruning, and its detailed and extensive evaluation are covered in the next chapter. Currently, the newly pruned and assessed AGROVOC ontology has been merged into the core ontology. Next steps include the refinement and extension as described in the 4th process of the framework. Further expert subject assessment is needed to accomplish a first version to be incorporated into the IP-FsAPH. The evaluation application used for document indexing and enhanced search still has to be linked to the system and documents have to be indexed in order to provide searchable metadata. An ontology enhanced search scenario has to be implemented, taking ontology compiled search queries and returning ontology enhanced search results. This will finally enable the usability testing of the whole approach.

[24] Refer to [http://www.getess.de] for an overview of the project.

5 The ontology pruner

This chapter depicts the ontology pruning step in the ontology creation framework introduced in the previous chapter. I will reason its valuable application within the framework by presenting an extensive evaluation of the pruning algorithm. The evaluation focuses on applicability of the algorithm to the domain at hand and the influence of different parameter settings on the quality of the output. In the following section, I will give an introduction to tree pruning and explain in detail the algorithm used here, which has been used in earlier research studies on ontology acquisition in the insurance and finance domain in [Volz00]. 5.2 will explain the adaptations and improvements of the formerly used algorithm, in order to fit the environment of the project. I will explain the evaluation plan and the test set used for the evaluation. Finally, conclusions on the applicability of the algorithm within this project environment are drawn from the evaluation results.

5.1 Introduction to the pruning approach

Tree pruning algorithms are widely used in the field of automatic classification in order to prune decision trees. A wide variety of algorithms have been applied for this task and a comprehensive comparison of several algorithms can be found in [BrA97]. A promising approach based on the minimum description length principle has been applied in [MRA95]. Decision trees are, however, not quite equivalent with an ontology structure and therefore an adjusted approach has to be taken here. The problem to be solved here can be described as follows:

We are given an ontology structure (according to Definition 2) with a set of concepts C and a tree-like hierarchy order of these concepts $H^c \subseteq C \times C$. If $(c_1, c_2) \in H^c$, then c_1 is a sub-concept of c_2 and c_2 is a super-concept of c_1. The tree hierarchy contains no cycles, but is not necessarily a simple tree, meaning that each concept can have several super- and several sub-concepts. Additionally, we are given a set P of non-hierarchical relationships between these concepts as well as a function $P \times C \rightarrow 2^C$ assigning to each relationship-concept pair the related set of concepts. Moreover, we are given a lexical structure of labels and synonyms as described in Definition 7. The given ontology structure can describe a wide area, containing concepts of various different domains such as the interdisciplinary ontology built in this project.

Definition 9 (Ontology Pruning Problem). Let $D = \{d_1,\ldots, d_l\}$ be the set of all possible domains. Let $D^1 = \{d_i,\ldots, d_k\}$ be a larger subset of D and D^2 be a smaller subset of D and D^1 containing only the domains $d_{i+x}, \ldots, d_{k-y}$. D^2 is called the set of target domains. Assuming that a mapping m: $C \rightarrow 2^D$ exists, mapping each concept of an ontology to one or more domains, we can formulate the following: Let C^1 be a set of concepts describing an ontological structure, called the source ontology, so that

- for each $c \in C^1$ $m(c) \in D^1$,
- $n_{d1} = |C^1|$,
- $n_{d2} = |\{ c \text{ e } C^1 \mid m(c) \text{ e } D^2\}|$, and
- $n_{d2} <= n_{d1}$.

We are looking for the best possible transformation t: $C^1 \rightarrow C^2$, so that

- $C^2 \subseteq C^1$,
- for each $c_2 \in C^2$: $c_2 \in C^1$ and $m(c_2) \in D^2$, and
- $|C^2| \sim n_{d2}$.

Figure 17 shows the ideal transformation:

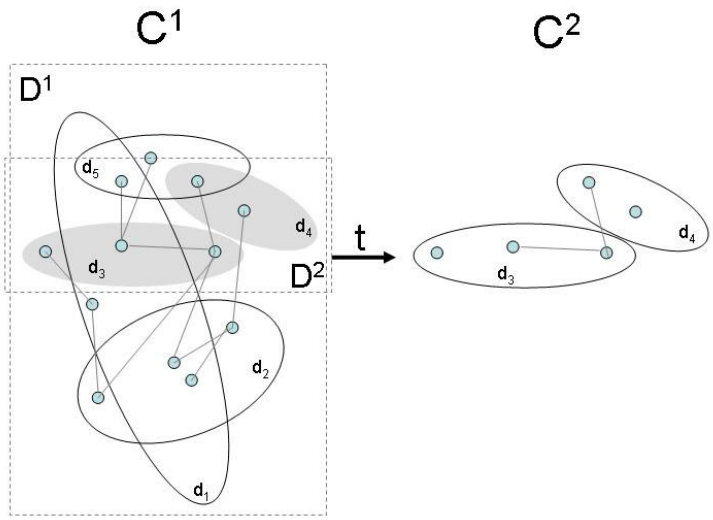


Figure 17: The ontology pruning problem

In other words, we are trying to extract all the concepts from the set C_1, which represent and describe the set of target domains D_2. The domain mapping is an assumption and not explicitly present in the ontological structure, i.e. no formal conceptualisation exists for it. We therefore have to take a heuristic approach, in order to get as close to the above ideal extraction as possible.

A representative set of documents specifying the domain of interest and therefore, one assumes, containing and defining all the concepts of that domain can accomplish the task of domain assumption. This domain specific document corpus (Δ) has to be carefully chosen by subject specialists in that area and should cover all aspects of the domain it represents.

The term pruning in the sense used here means the deletion of domain-unspecific concepts from the more generic source ontological structure C_1. In order to prune domain-unspecific concepts, their term frequencies are determined from the selected domain-specific documents (see also [Sal88] for this approach) and moreover from a second corpus (Γ) that contains generic documents (as found in reference corpora like CELEX[25] or public news archives). This second corpus serves as a reference against which frequencies are compared to filter out the unspecific concepts. All concept frequencies are propagated to super-concepts, by summarising the frequencies of sub-concepts. Then the frequencies of both corpora are compared using one of two different measures. The here used measure tf (term frequency) coupled with idf (inverse document frequency), first introduced in [Jom72], is widely applied today in the information retrieval community:

TF (term frequency)

Here, simply the frequencies are counted in absolute numbers

TFIDF (term frequency - inverted document frequency)

Here, a term-weighting factor (idf – inverted document frequency) is attached to the original TF, which punishes all terms that are frequent in all documents, using a collection frequency. Concepts occurring in almost all documents obviously accumulate high frequency counts, but can be assumed to be rather unspecific for that domain. The term weighing factor could be as easy as simply the number of documents, the concept occurred in. In that case, however, absolute size of the domain specific corpus largely influences the weighing factor. A better measure used here (following Salton's definition in [Sal88]) and relating the number of documents, the term occurred in (df) to the total number of documents is

$$TFIDF = TF * \ln\left(\frac{|\Delta| (bzw. |\Gamma|)}{df}\right)$$

[25] [http://europa.eu.int/celex/htm/celex_en.htm].

This measure gives a concept a higher weight (with decreasing inclination), the fewer the documents it appears in and a lower weight, eventually reaching 0, if it appears in all documents.

Another weighing factor widf (weighted inverse document frequency) is discussed in [TI94] and could be interesting to evaluate in future tests.

All existing concepts that are significantly more frequent within the domain-specific corpus than in the generic corpus remain in the ontology. The degree of significance can be varied using the **ratio** parameter r. A concept will be deleted, if its weighed term frequency is not at least r times higher than its counterpart in the generic set. Another parameter - **beat strategy** - determines the way, in which the frequencies of the concepts in the different document sets are compared. The beat strategy ALL compares only the overall frequencies, summarised over all documents in the respective sets. On the other hand, one might reason that it might be enough for a concept to be specific for a domain, if it occurs significantly more frequently in one document of the domain corpus than in another one of the generic corpus (beat strategy ONE).

5.2 Adaptation of the ontology pruner

An ontology pruner formerly applied in [Volz00] has been extended in several ways. First, the ontology access has been changed to read ontologies represented in the ontology language of KAON, version 1.2, as introduced in chapter 3. In KAON language, lexical entries are attached to instances. The important lexical entries for the ontology pruning task are labels and synonyms of a concept, since they describe and represent the concept in the respective language. So, in fact, every occurrence of either a synonym or a label in a document of a corpus increases the frequency count of this concept in this corpus.

The second applied change created a new version of the ontology pruner. The existing version was not able to recognise the occurrence of a concept in a document, if its label/synonym is a compound word (i.e. a label consisting of more than one word like ‘animal health’). In the new version, documents are pre-processed in a slightly different way. Figure 18 shows the process more clearly in comparison of the two versions.

The upper part shows the process in the old adapted version of the pruner, whereas the lower part shows the new version, now called Ontology Pruner Trie, in accordance to the here used data model. The file processor has been extended to read in HTML Files. In the old

version, documents have been pre-processed in a number of steps. First, any tags are removed to extract the content text only. Then a stop-word list is applied to filter out language specific fill words (such as 'and', 'in', etc.). In a next step, the remaining words are reduced to their word stems. After this pre-processing, each document is represented by its word vector, being a vector of tupelos (word, frequency of word in document). Finally, all the word vectors are compiled into one term-frequency table, containing the overall frequency of each word stem in the document set. From the ontology, the label-concept mapping is extracted into a hash table. In the following step, each term from the term-frequency table is checked for occurrence in the label-concept hash table, and if found, the concept's frequency count is increased. This imposes, that only labels with one word can be recognised using this approach.

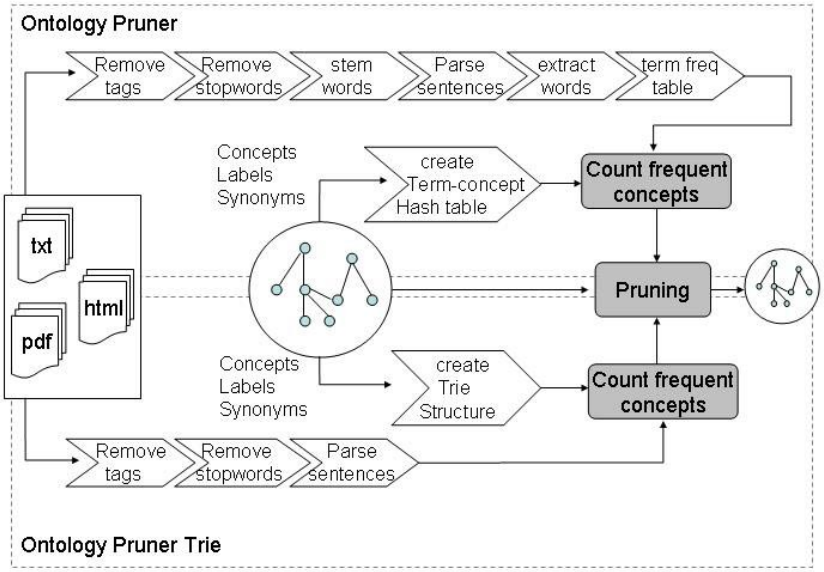


Figure 18: Pruning process – old vs. newly adapted version

The revised version of the pruner is now able to recognise compound words. This is achieved by changing the process of document pre-processing and frequency counting and by a different data representation of the term-concept mapping. Now documents are not represented by their word vectors anymore. The ontology labels and synonyms are first pre-processes to build a TRIE tree structure[26]. Refer to [NN95] for a detailed description of this data structure. The number of document pre-processing steps has been reduced to half, significantly improving the overall performance of the pruner. The pre-processed documents are now processed through the Trie tree structure one by one, counting the frequency of a

[26] See [http://www.nist.gov/dads/HTML/trie.html] for a short introduction on TRIE structures.

respective concept, whenever a leaf of the tree is reached (i.e. a label or synonym has been found in the text).

In the current implementation, each occurrence of a label or synonym increases the frequency count of a concept by 1, no matter, where the label or synonym appears in the text. It might be interesting for future evaluations to evaluate, whether it makes a difference to work with weighed frequency counts. For example a concept's lexical entry occurring in a heading or in the beginning of a paragraph might have a higher significance than its occurrence elsewhere in the text.

After the frequencies of all concepts are counted, the frequencies are then propagated to the super-concepts of each concept. This reflects the idea that if a concept occurs in a document, then also its broader super concepts are represented, even if not directly through their lexicalisations. These frequent concepts are finally checked against the initial source ontology and all infrequent concepts are deleted, resulting in the pruned ontology.

Another change to the existing ontology pruner, applied in both versions, is the output of a critical concept list. Based on the way of frequency counting and promoting to super-concepts, and later comparison between the domain-specific and the general corpus, it can happen, that a concept is more frequent in the domain corpus than in the generic one, but that its super-concept is less frequent. This can happen due to the fact that an ontology is not a simple tree. A short example shown in figure 19 shall explain this.

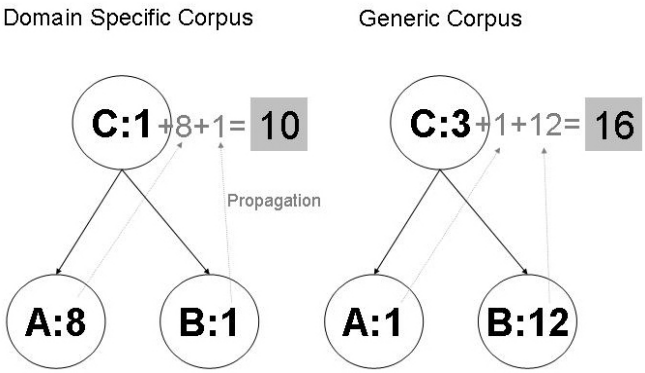


Figure 19: Frequency Propagation – frequent concept with infrequent super concept

Concept A occurs 8 times in the domain corpus but only 1 time in the generic corpus. Concept B occurs 1 time in the domain corpus, but 12 times in the generic one. The super-concept C of both concept A and B, however, only occurred once in the domain corpus, but 3 times in the generic one. Through propagation of the frequencies, C now gets the frequency

count 10 for the domain corpus and 16 for the generic corpus. Obviously, concept C is not frequent, but remains in the structure, since it has a frequent sub-concept A. These concepts are considered critical and output to a critical concepts list.

5.3 Evaluation

5.3.1 Resources: Document corpus and source ontology

Two different document sets are needed in order to evaluate the ontology pruner, a domain specific and a generic one. Concerning the size of the corpus, no significant requirements apply and research has not been done in order to find out, if the size of the document corpus actually affects the output of the pruning process. The only requirement is that the two used document sets should be approximately equal in size (amount of textual data). The number of documents in each set could moreover influence the results, if TF instead of TFIDF is used as frequency measure because absolute numbers are taken here. Three sets of documents have been compiled for evaluation purposes within the FAO:

Domain Document Set

According to the project environment described in chapter 4, this document set has been compiled from the areas of:

- Food Safety,
- Animal Health and
- Plant Health

The documents have been carefully chosen by subject specialists in the respective areas against the requirement of covering all aspects of each area. The total number of documents chosen is 90, of which 68 are plain ASCII text and 22 are HTML documents. Each of the above areas is represented in approximately equal weight. The total size of the documents is 9.73 MB.

Two different generic document sets have been compiled:

Generic Document Set 1 (Gen):

The first set of generic documents has been chosen randomly from generic news. It consists of a collection of 25 generic news texts, taken from various random English news web sites.

Additionally, 7 files have been taken from the Reuters 21578 test collection[27] described in [Lew99], each of which being approximately 1,3 MB in size. The Reuters collection contains news texts from a wide variety of different areas and can therefore be used to compile the generic document set.

In summary, the first generic document set consists of 32 documents accounting a total of 9.55 MB of data.

Generic Document Set 2 (AG):

The second generic document set has been chosen motivated by the hypothesis that evaluating a domain specific corpus against a corpus taken from a similar domain, but covering a wider selection of areas except the ones constituting the domain-specific corpus, might lead to even better filtering of the domain-specific concepts. Therefore, this second generic corpus has been compiled of 142 randomly chosen html news articles from the Agricultural Research Service of the United States Department of Agriculture, as well as a selection of 73 documents from different FAO research areas, covering a broad range of agricultural topics. This adds up to a collection of 215 documents at a size of about 4 MB. The size of this set is significantly smaller than the domain-specific set. This is, however, reasonable taking into account the significantly bigger number of documents and the fact, that some of the documents might as well cover concepts, also specific for the domain of interest.

Source Ontology:

For this evaluation, the converted AGROVOC serves as source ontology. Basically, every other thesaurus, ontology can be pruned, as long as it is represented in RDFS (KAON RDFS) format. The statistics of the converted AGROVOC ontology is shown in Table 1.

# concepts	# properties	# hierarchical relationships	# property instances	# Related Term Instances	# Used For in Combination instances	Maximum Taxonomic Depth
17506	3	17168	15285	13486	1799	8

Table 1: AGROVOC ontology statistics

The low number of properties is due to the meta-modelling approach explained in the preceding chapter. The richness of connections is represented by the number of property instances, which are instantiations of the 'related term'-property and the 'used for in

[27] The Reuters 21578 collection is available at [http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html].

combination'-property in the case of AGROVOC. The taxonomic depth of 8 indicates the AGROVOC to have a rather flat structure with respect to the high number of concepts. The converted AGROVOC obviously contains more concepts than the overall number of descriptors (16607), due to the 'used for in combination'-relationship, for which new concepts have been introduced as explained before in chapter 4.

The AGROVOC has been chosen here for reasons of availability. However, AGROVOC is not the best resource to be applied and pruned for this domain. According to subject specialists of the above areas, AGROVOC is not very expressive in the fields of Food Safety, Animal and Plant Health. On the one hand, it misses out on many of the main concepts representing the area; on the other, it is very specific in certain sub areas, somehow belonging to the above but not necessarily in the same sense and depth. This fact might have an influence on the overall evaluation here and has to be considered in the discussion and interpretation of the results.

5.3.2 Hypotheses for evaluation

From the above discussed adaptations of the ontology pruner and the used document sets, we can infer the following criteria and hypothesis for evaluating the ontology pruner:

The effect of the variation of the 3 main parameters frequency weighing measure (TF, TFIDF), beat strategy (One, All) and ratio on the size of the pruned ontology shall be evaluated.

The new version of the pruner should be able to recognise more concepts due to the ability of recognising compound words. Therefore, the pruned ontology should be bigger in size, leaving all other parameters constant. Above variations of parameters are therefore applied to both versions of the pruner.

Using the Generic Document Set 2 (AG), a more specific selection of concepts can be expected and hence a smaller pruned ontology. Since AGROVOC is used as source ontology, it contains concepts specific for the overall domain of agriculture. Evaluating the domain-specific corpus frequencies against generic corpus frequencies using the generic news set, lots of concepts might remain, specific for the overall domain, but not very specific for the domain of Food Safety, Animal and Plant Health. The evaluation against this second generic set of documents taken from the agricultural area might reduce this effect.

The above criteria are more statistically oriented, in a sense that the measures to evaluate the criteria refer to the size and statistics of the output ontology. Beyond these number

oriented measures, the relevance of the extracted concepts and of their descriptiveness towards the specified domain is of extreme importance. However, this is an empirical evaluation and needs human support in form of a subject specialist assessing the output. This is a very time intensive task and can therefore only be done exemplary with very few examples.

A further interesting aspect is the evaluation of the critical concepts list. Therefore another option is introduced, leaving the choice of deleting all infrequent concepts, no matter if they have frequent sub-concepts, or keeping them and outputting them to a critical concepts list to be assessed later by a human subject specialist. This is again a time consuming task and can only be done exemplary.

The following section presents the complete evaluation and its results and discusses the findings against the above statements.

5.3.3 Evaluation plan:

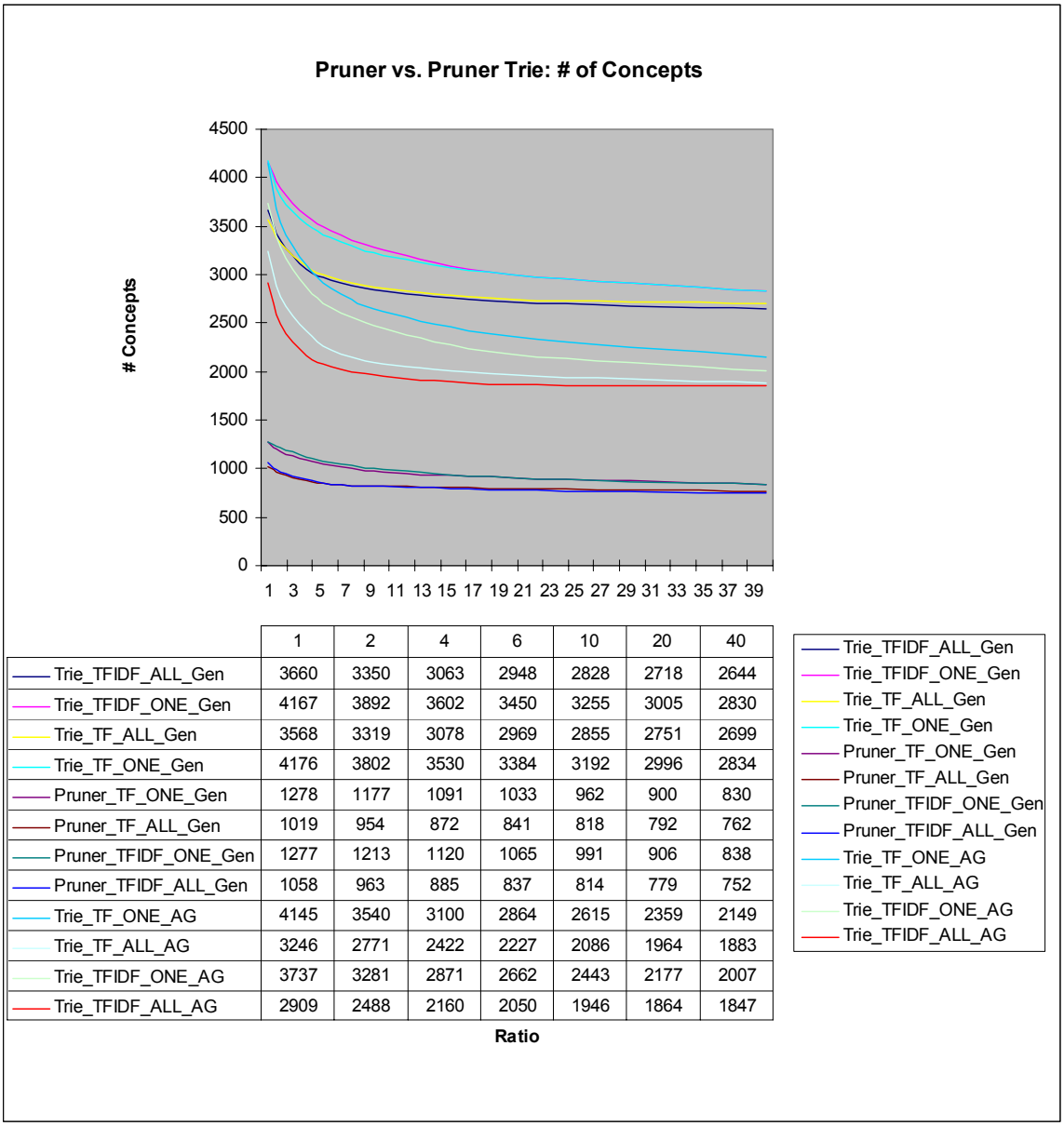
The Ontology Pruner, as well as the Ontology Pruner Trie have both been evaluated using the Generic Document Set 1 (Gen) and following parameter variation setting: The frequency weighs have been varied between TF and TFIDF, the beat strategy oscillated between ONE and ALL, while the ratio has been varied using the discrete values {1.0, 2.0, 4.0, 6.0, 10.0, 20.0, 40.0}. Moreover, the same parameter variation has been applied to the Ontology Pruner Trie using the Generic Document Set 2 (AG). These evaluation runs are all statistically evaluated using the same measures as used in Table 1 to represent the AGROVOC statistics.

The pruned ontology with the highest number of concepts has been chosen for empirical assessment and evaluation by subject specialists. Subject specialists deleted out all non-relevant terms (non-relevant towards the goal domain). The remaining ontology has then been tested against inclusion in the other pruned ontologies. This gives an idea about to which extend the other parameter settings could succeed in filtering the relevant and more specific concepts.

Moreover, another evaluation run has been conducted using the same parameter configuration, creating the list of critical concepts by not deleting infrequent concepts having frequent sub-concepts. To give an idea about the value of this option subject specialists as well assessed this list.

5.4 Results and Discussion:

Figure 20 shows the results of the evaluation of the old pruner version against the Pruner Trie version, recognising compound words. The graph shows the number of concepts of the pruned ontology structure in dependency on the chosen ratio value in each evaluation run. The different curves result from variation of the other 2 main parameters as well as the two different generic document sets and the pruning version used. Each curve belongs to one specific set of these other parameters.



	1	2	4	6	10	20	40
Trie_TFIDF_ALL_Gen	3660	3350	3063	2948	2828	2718	2644
Trie_TFIDF_ONE_Gen	4167	3892	3602	3450	3255	3005	2830
Trie_TF_ALL_Gen	3568	3319	3078	2969	2855	2751	2699
Trie_TF_ONE_Gen	4176	3802	3530	3384	3192	2996	2834
Pruner_TF_ONE_Gen	1278	1177	1091	1033	962	900	830
Pruner_TF_ALL_Gen	1019	954	872	841	818	792	762
Pruner_TFIDF_ONE_Gen	1277	1213	1120	1065	991	906	838
Pruner_TFIDF_ALL_Gen	1058	963	885	837	814	779	752
Trie_TF_ONE_AG	4145	3540	3100	2864	2615	2359	2149
Trie_TF_ALL_AG	3246	2771	2422	2227	2086	1964	1883
Trie_TFIDF_ONE_AG	3737	3281	2871	2662	2443	2177	2007
Trie_TFIDF_ALL_AG	2909	2488	2160	2050	1946	1864	1847

Figure 20: Pruner vs. Pruner Trie, evaluation results

Obviously, 3 clusters or groups of curves can be identified. The upper 4 curves represent the results of the Trie version of the pruner run against the Generic Document Set 1 (Gen).

The 4 curves situated in the middle belong to its application with the Generic Document Set 2(AG), whereas the lower 4 curves show the results of the old pruner version.

5.4.1 Pruner Trie vs. Pruner:

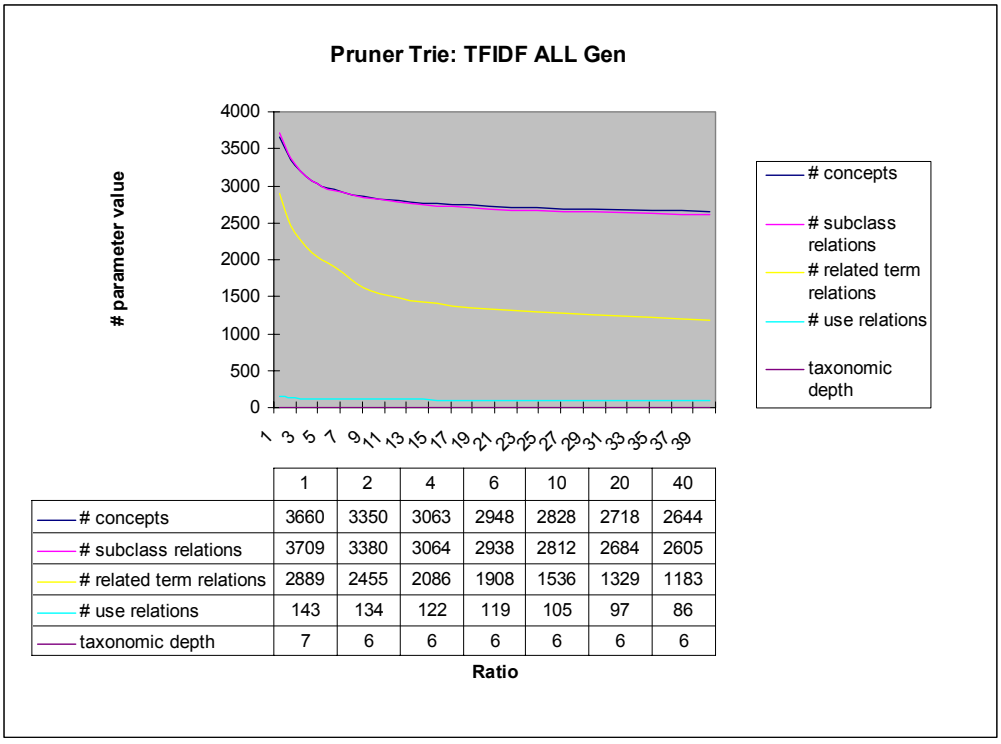
It can clearly be concluded, that the Trie version without exceptions created pruned ontologies significantly bigger (almost 4 times in case of using the Gen generic set) in size than the old version. Subset tests have shown, that each pruned ontology output from the old pruner is a total subset of its counterpart resulting from the Trie pruner version. The extension to work on compound words and include concept synonyms obviously recognises more terms, hence extracting more domain specific knowledge from the source ontology structure. It is however, not clear at this stage, if all this additional extraction is actually usable information in terms of relevance towards the domain. This issue can only be addressed by expert assessment and will be discussed in a few moments. Before this is outlined, some general conclusions concerning the base parameters can be drawn from above picture.

5.4.2 Dependency of the statistics on different parameter settings:

Within all three groups of curves, two sub groups can be identified. The two upper curves always belong to the evaluation runs using the beat strategy ONE. It can therefore be argued that the usage of ALL leads to a more specific filtering. This seems reasonable, since the frequency comparison is more restrictive in the latter case. The usage of TFIDF vs. TF on the other hand does not seem to make any significant difference. The effect of decreasing number of concepts due to the variation of the ratio parameter is declining with increasing values of ratio. In the older pruner version, a ratio value bigger than 7 could not bring any significant difference. In the Trie version, values bigger than 15 did not have any significant effect on reduction of the size of the pruned ontology.

I here focus on the number of concepts, since the other ontology statistics are more or less dependent on the development of this main variable. Figure 21 shows the dependency of all statistical ontology measures exemplary for a pruner series, leaving fixed the document set, the frequency measure (TFIDF) and the beat strategy (ALL), varying only the ratio. The graph shows, that the development of the hierarchical relationships and the ‘related terms’ relationships almost directly correlates with the number of concepts, whereas the ‘use’ relationship and the taxonomic depth do not vary significantly, in fact show very little decrease only. The development of the statistics is equivalent for the other parameter settings. Therefore, I abstain from showing all results here. From the fact that the ‘use’ relationship

doesn't vary significantly, we could assume, that the concepts which are linked by these relations are actually quite important for the domain, since they do not get pruned out, using more restrictive settings. As explained in the previous chapter, these concepts have been established from former AGROVOC non-descriptors. It might therefore be interesting to assess these concepts for inclusion into the ontology hierarchy, since they seem to be important for this specific domain! The little variation shown in the maximum taxonomic depth of the pruned ontology directly results from the frequency propagation approach explained in the last section.



	1	2	4	6	10	20	40
# concepts	3660	3350	3063	2948	2828	2718	2644
# subclass relations	3709	3380	3064	2938	2812	2684	2605
# related term relations	2889	2455	2086	1908	1536	1329	1183
# use relations	143	134	122	119	105	97	86
taxonomic depth	7	6	6	6	6	6	6

Ratio

Figure 21: Dependency of all statistical ontology parameters on variation of the ratio parameter (exemplary for the setting TFIDF ALL Gen with Ontology Pruner Trie)

5.4.3 Generic Document Set 1 (Gen) vs. Generic Document Set 2 (AG):

The use of the second generic document corpus (AG) could certainly lead to smaller ontologies as assumed. A pruned ontology resulting from the use of the AG corpus contains an average of 2565 concepts versus an average of 3234 concepts using the Gen corpus. Subset tests have now been conducted testing each output ontology - resulting from the use of the AG document corpus – for inclusion in its counterpart output using the Gen corpus (all other parameters are the same). None of the pruned ontologies resulting from the AG set is a complete subset of its Gen counterpart. On average, the AG outputs contain 213 concepts (with a standard deviation of 53), which are not found in the Gen output. This number is,

however quite constantly distributed amongst all the outputs, leading to the conclusion, that using the AG generic set, a subset of around 213 different concepts (not identified using the Gen set) could be extracted using the AG corpus. It would be interesting to evaluate the relevance of these additional concepts. Due to lack of subject expert availability, this step could not be conducted within this work. At this point, we can therefore only assume, that these additional concepts might be relevant for the domain.

On the other hand, the subset tests have revealed that an average of 2351 concepts has equally been identified in both the tested ontologies, hence an average of 883 (with a standard deviation of 235) concepts have been pruned out using the AG corpus instead of the Gen corpus. In other words, using the AG corpus, the pruner identified an average of 71% of the concepts identified using the Gen corpus to be relevant concepts. This obviously confirms above made hypothesis, that the use of the AG document set leads to a further specification and pruning of several concepts assumedly not that specific for the target domain. However, with this information, no statement can be made on the usability and relevance of this additional specification, i.e. if the 'right' concepts have been left out.

The discussion so far has been based on statistical evaluation and assessment only, hence not including any conclusion regarding the actual usability of the pruned ontologies and the value of the extracted knowledge with respect to the target domain. This can only be achieved by an empirical evaluation and subject expert assessment. The results of this are discussed in the following:

5.4.4 Empirical evaluation:

Obviously the largest pruned ontology has been output from the Ontology Pruner Trie, using the Generic Document Set 1 (Gen), TF as weighed term frequency, ONE as beat strategy and the ratio 1.0, meaning that all concepts have been deleted whose weighted term frequency was less than their counterpart's in the generic document set. Let this ontology be O_{Pruned}.

The ontology O_{Pruned}, pruned based on these parameters, has been assessed by 3 subject experts specialised in the food safety domain area. This choice has been made for reasons of availability of resources and time. It was not possible at the time of evaluation to let the ontology be assessed by subject experts perfectly representing all target domain areas. The output is certainly influenced by this circumstance and an uncertainty factor is added. Obviously, the constitution of the expert group might lead to unnecessary deletion of relevant concepts as well as subjectivity of the subject specialists might lead to different and

inconsistent decisions, compared with others in that field. This fact, however, always holds in real working environments, also when building an ontology without computer aided support. We have to measure against what we have and the perfect solution for such modelling decisions does not exist. In that context, we can even view the ontology pruner as just another human being, trying to do the same as the group of experts and evaluate his performance against that of the expert group. I will take the lack of an objective perfect basis against which to measure as a given weakness in the further discussion. Table 2 shows the statistics of the resulting, assessed and further pruned ontology in comparison to the ontology pruner output. Let the assessed ontology be $O_{Assessed}$.

statistics	# concepts	# properties	# hierarchical relationships	# property instances	# 'Related Term' Instances	# 'Used For in Combi-nation' instances	Max. Taxo-nomic Depth
Ontology Pruner Output (O_{Pruned})	4176	3	4269	3772	3619	153	7
After expert assessment ($O_{Assessed}$)	3127	3	3161	2393	2262	131	6

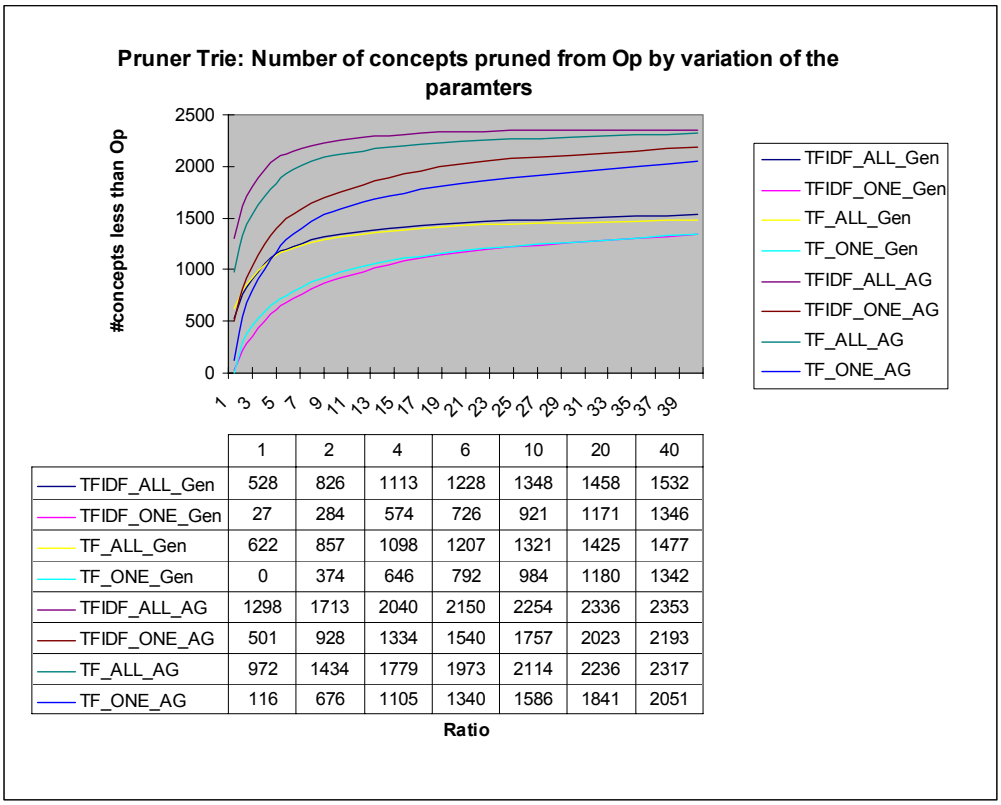
Table 2: Ontology Pruner output vs. subject assessment of this output

Since the number of concepts basically represents the main variable of the pruning process and the other parameters are all directly affected by the variation of the number of concepts, I will focus on the number of concepts in the further discussion. Taking the subject expert's judgement, 74,88 ~ 75% of the concepts extracted by this pruning parameter setting were valuable concepts (success rate). A rate of 25% (fault rate) was mistakenly identified domain specific by the pruner. At this stage we can already conclude without further testing that the new version of the ontology pruner has been able to recognise a bigger set of domain relevant concepts than the old pruning algorithm (cf. Figure 20). The highest number of concepts identified by the old pruner was 1278 hardly exceeding only a third of the here identified relevant set. Subset tests have shown that all of the output ontologies of the old pruning algorithm are complete subsets of O_{Pruned}. Clearly, the adaptations of the algorithm show significant improvement. In the further discussion, I will only focus on the Trie pruner version.

The interesting question is now: which of the other parameter settings might actually be able to increase the success rate to exceed 75% and decrease the fault rate of 25%? In other

words: Which parameter settings are able to filter out more of the 1049 unspecific concepts while keeping the subset of the identified 3127 relevant concepts?

In order to test this, several subset tests have been conducted. First, each resulting ontology of each pruning step has been tested for inclusion against O_{Pruned}. The result showed, that almost all the pruned ontologies using the Gen corpus are complete subsets of O_{Pruned}, except the ones with ratio 1.0, in which on average 14 other concepts appeared. The ontologies output using the AG corpus contain an average of 30 concepts, which do not appear in O_{Pruned}. Given the overall size of the ontologies, this effect is minor, and I will therefore not consider these concepts in the further discussion and consider only the parts of each ontology, which is fully included in O_{Pruned}. Assuming this, we are now given a set of 27 ontologies (output using the Gen generic corpus) and a set of 28 ontologies (output using the AG generic corpus), all of those being subsets of O_{Pruned}. Figure 22 shows the number of concepts less than in O_{Pruned} in dependency on the different parameter settings.



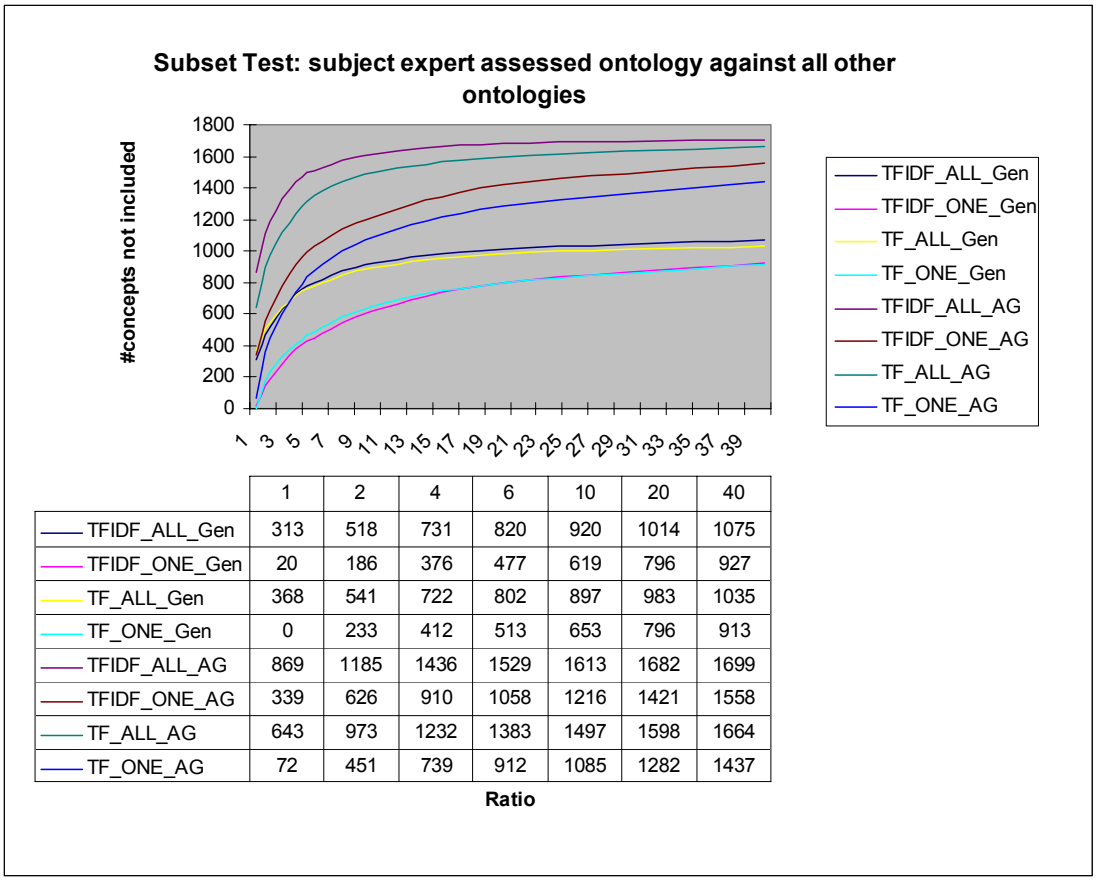
	1	2	4	6	10	20	40
TFIDF_ALL_Gen	528	826	1113	1228	1348	1458	1532
TFIDF_ONE_Gen	27	284	574	726	921	1171	1346
TF_ALL_Gen	622	857	1098	1207	1321	1425	1477
TF_ONE_Gen	0	374	646	792	984	1180	1342
TFIDF_ALL_AG	1298	1713	2040	2150	2254	2336	2353
TFIDF_ONE_AG	501	928	1334	1540	1757	2023	2193
TF_ALL_AG	972	1434	1779	1973	2114	2236	2317
TF_ONE_AG	116	676	1105	1340	1586	1841	2051

Ratio

Figure 22: Differences in size between largest pruned ontology and all others (Pruner Trie)

Given this basis, subset tests have been concluded testing $O_{Assessed}$ against each of the above 55 ontologies, in order to find out, to which degree the other parameters of the pruner could accomplish the same job than the subject specialists by assessing O_{Pruned}. I will refer to any of these 55 ontologies as O_x in the following definitions. Figure 23 shows (for each

parameter setting) how many of the concepts of $O_{Assessed}$ are not included any more in the respective automatically pruned ontology.



	1	2	4	6	10	20	40
TFIDF_ALL_Gen	313	518	731	820	920	1014	1075
TFIDF_ONE_Gen	20	186	376	477	619	796	927
TF_ALL_Gen	368	541	722	802	897	983	1035
TF_ONE_Gen	0	233	412	513	653	796	913
TFIDF_ALL_AG	869	1185	1436	1529	1613	1682	1699
TFIDF_ONE_AG	339	626	910	1058	1216	1421	1558
TF_ALL_AG	643	973	1232	1383	1497	1598	1664
TF_ONE_AG	72	451	739	912	1085	1282	1437

Figure 23: Number of domain specific concepts, which have not been identified by the automatic ontology pruner

Comparing this graph with Figure 22, there is obviously a strong correlation between this number and the number of additionally pruned concepts, using more restrictive parameter sets. With this information we can now define two measures:

Definition 10 (specification correctness). Let n_{pruned} be the number of concepts less in any ontology O_x than in O_{Pruned}. Let $n_{assessed}$ be the number of concepts, which are in $O_{Assessed}$ but not included in O_x. The number n_{corr} of correctly pruned concepts is herewith $n_{corr} = n_{pruned} - n_{assessed}$. The specification correctness is now computed as:

$$\text{Specification Correctness: } S_{corr} = \frac{n_{corr}}{n_{pruned}}$$

Definition 11 (specification recall). Let further $n_{CorrAss}$ be the number of correctly pruned concepts by the subject experts, i.e. $n_{CorrAss} = | O_{Pruned} \setminus O_{Assessed} |$. The specification recall is defined as:

Specification Recall: $S_{rec} = \frac{n_{corr}}{n_{CorrAss}}$

The specification correctness computes the ratio of the correctly automatically pruned concepts to all the automatically pruned concepts, i.e. it gives us an idea of how correct the further automatic pruning was. The specification recall on the other hand gives us an idea of how many of the concepts identified by the human experts have also been identified by the automatic pruning using different parameter settings.

	S_{corr}	STDEV	S_{rec}	STDEV
Gen	0.34	0.03	0.31	0.11
AG	0.31	0.02	0.47	0.15

Table 3: Specification correctness and specification recall for automatically pruned ontologies

The computed measures for the conducted subset tests are shown in Table 3. The specification correctness is rather constant for all automatically pruned ontologies at around 33% with a very low deviation. This means that throughout all parameter variations, only around a third of the further automatically pruned concepts have been correctly pruned, the remaining two thirds have been mistakenly left out (i.e. they have been deleted by the automatic pruning algorithm, whereas the subject experts defined them to be relevant for the target domain). Regarding the recall, the use of the Gen set vs. the use of the AG set showed a significant difference. Using the generic corpus, only around a third of the concepts pruned out by the subject experts could be identified, whereas using the AG corpus, almost half of the irrelevant concepts could be identified. This fact now supports our formerly made assumption that using the AG corpus can lead to a more specific extraction of concepts from the source ontology. However, this number has to be interpreted in conjunction with the correctness. The higher specification recall using the AG set is achieved on a higher total cost of loss of information, since the precision is even less than in the Gen set. In other words, the higher recall could only be achieved, deleting a much higher number of concepts in total, hence also deleting out a higher number of relevant concepts.

5.5 Summary

The ontology pruner has been adapted to recognise compound words. It could be shown, that this adaptation better approximates the transformation t introduced in Definition 9. Hence, this adaptation better succeeds in solving the pruning problem. The evaluation has been based on the largest resulting ontology, which has been automatically extracted from the ontology, given the used parameter variations. Due to before discussed limitation of

availability of human resources another necessary step in order to draw a better conclusion on the value of the pruner extraction could not be conducted: the up-front human assessment of the whole AGROVOC in order to manually identify the whole set of concepts relevant for the target domain. This manually pruned ontology would be a better base source to evaluate against. It would be interesting to see, if the largest pruned ontology actually contains all the concepts identified by that exhaustive manual assessment. Given the restrictions, however, it had to be assumed, that the largest automatic extraction contains at least most of the relevant concepts. Given that assumption, we found that none of the variations of the pruner parameters could succeed in loss-less further pruning, i.e. the largest extracted resulting ontology could not be further automatically pruned without losing a significant amount of relevant concepts. It has been shown, however, that using a generic document set, which represents the surrounding area of the target domain (here the AG set), succeeded in identifying more of the non-relevant concepts. This higher rate can on the other hand only be achieved on a higher total cost of losing a larger set of domain relevant concepts.

In conclusion, no clear statement can be derived concerning an optimal parameter setting. If the aim is to extract possibly all relevant information from the source ontology, then the best approach is to apply the pruner with the least restrictive parameter setting and then further assess the result by subject experts. If, however, subject experts are not available and the goal is to rather retrieve a subset of the source ontology, which includes the least possible amount of irrelevant concepts, even on risk of loosing valuable concepts, then a more restrictive set of parameters should be chosen.

Moreover, the here drawn conclusions and findings are highly depending on the used source ontology and the compilations of the document sets. A slightly different compilation of the document sets might have lead to different results. It might be interesting in our case to identify 3 different domain document sets representing the sub areas of food safety, animal health and plant health separately and apply them to the pruner in separate evaluation runs, later merging the resulting ontologies. In further work, this evaluation should be applied in different domains, in order to see if the statements and conclusions derived above still hold.

6 Automatic Classification

6.1 Introduction

6.1.1 What is text categorisation?

Text categorisation[28] is the process of algorithmically analysing a document to assign a set of categories (or index terms) that succinctly describe the content of the document [RS01]. This process is performed quite naturally by human beings. The origins of automatic text categorisation date back to the early 1960's when also the term "automatic document classification" was introduced to name different tasks in this field: the automatic assignment of documents to categories, the automatic definition of categories (also known as clustering), the automatic assignment of uncontrolled vocabulary (extracted from the free text of the document instead of taken from a predefined set) to documents. In this thesis, I refer to automatic text categorisation as 'the use of statistical patterns of word occurrences in documents to select predefined categories for indexing documents' [Luh58]. Documents can be deterministically assigned to one category (single-label case) or to an undefined number of multiple categories (multi-label case).

One application of text categorisation is document indexing, in which documents are assigned an arbitrary number of keywords, which describe the content of the document. These keywords can be taken from a controlled vocabulary like the AGROVOC in the FAO as I already described in section 2.2.3. In this application, keywords are viewed as categories and hence document indexing becomes a multi-label text categorisation problem. Librarians traditionally carry out the indexing process manually as a costly effort. The motivation for automatic support with today's exponential growth in electronically available documents becomes evident.

6.1.2 Motivation within the project context

The FAO manages a vast amount of documents and information on agriculture. Professional librarians and indexers using the AGROVOC as a controlled vocabulary for keywords manually index all documents and resources managed by FAO's information management system. Each resource is assigned an arbitrary number of keywords from the AGROVOC, describing the content of the document. This process is applied to resources in

all the official FAO languages and herewith constitutes a multilingual problem. The cost of labour and the fast growth in available electronic resources keep the system from being filled with the adequate amount of available resources. Automatic document indexing could be particularly useful in digital libraries as the ones maintained at the FAO in order to make more resources available through the system. In the following, I will reason the adaptation of a support vector machine classifier to be a promising approach towards this multilingual, multi-class, multi-label classifying problem.

6.2 Basic definitions

6.2.1 Using Support Vector Machines for Multi-label Document Indexing

Various methods have been applied in text categorisation approaches and can be classified into:

- Classical IR based classifiers
- Statistical learning classifiers
- Linear Classifiers
- Instance-Based Classifiers
- Decision Trees
- Inductive rule learning
- Expert systems
- Neural networks
- Support vector machines

It is not my intention to give an extensive overview about all available methods, but rather focus on the approach used here throughout the remainder of this chapter. A comprehensive survey of machine learning algorithms is given in [RS01] and [AE99].

Definition 12 (Multi-Class, Multi-Label Classification Problem). We are given a set of training documents $x_i \in X$ and a set of possible classes $C = \{c_1,\ldots,c_n\}$. Each document is assigned a subset $C_1 \in C$ ($|C_1| = m$) of relevant topics. The task is (according to [Seb99]) to approximate the unknown target function $\overline{\Phi} : X \times C \rightarrow \{true, false\}$ (that describes how

[28] Throughout this chapter, I use the terms categorisation and classification as well as categories and classes as synonyms.

documents ought to be classified) by means of a function $\Phi : X \times C \rightarrow \{true, false\}$ called the classifier (aka rule or hypothesis or model), such that they coincide as much as possible.

In a multi-class, multi-label classification problem, each document can be assigned an arbitrary number m (multiple labels) of n (multiple classes) possible classes. In the single-label case, only one class is assigned. The binary classification problem is a special case of the single-label problem and can be described as follows:

Definition 13 (Binary Classification Problem). Given a set of documents $x_i \in X$, each of the documents will be assigned to one of two possible classes c_i or its complement $\hat{c}_i$.

Obviously, document indexing as introduced in the previous section and applied within the FAO is a multi-label, multi-class problem. There are different alternatives towards this approach. The one I adopt in this research (another approach is mentioned later when I discuss related work) is to transform a multi-classification problem into $|C|$ independent problems of binary classification. This requires that categories be stochastically independent, that is, for any c', c'' the value of $\Phi(x_i, c')$ does not depend on the value of $\Phi(x_i, c'')$ and vice versa. In the case of document indexing in the FAO, this is a reasonable assumption. Consequently, the research carried out in the remainder of this thesis builds on an approach, using binary support vector machines with background knowledge integration, formerly applied in [Pac02].

Vapnik first introduced support vector machines (SVM) in 1995 [CV95]. In support vector machines, documents are represented using the vector space model:

Definition 14 (Vector Space Model). A document x is transformed into an n-dimensional feature space $\Phi(x) \in IR^n$. Each dimension corresponds to a word/term (also referred to as feature). The values are the weighed frequencies of the words in the document. A document is represented by its vector of term weights,

$$\text{word-vector } \vec{x} = (w_1, ..., w_{|T|}),$$

where T is the set of terms (features) that occur at least once in at least one document in the whole set and the w_k represent the term weigh, i.e. the semantics of how term k contributes to the semantics of a document.

A wide variety of weights exist (as indicated in section 5.1) and different ways of what to choose as a term/feature. A more detailed discussion can be found in [Seb99]. Here, words are chosen as features and the standard tfidf (Term Frequency Inverse Document Frequency) measure is used as term weight, calculated slightly different from the definition given in section 5.1 as:

$$tfidf = \ln(tf + 1) * \ln(\frac{N}{df}),$$

where N is the total number of documents and df (document frequency) is the number of documents, a term occurred in.

A binary SVM tries to separate all the word vectors of the training document examples into two classes by a hyper plane, maximising the distance of the nearest training examples. Therefore, it is also referred to as the maximum margin hyper plane. A test document is then predicted by the SVM by determining, on which side of the hyper plane its word vector is. A good and detailed introduction to SVM and also to the document representations is given in [WF99].

Using all words of the documents for building the feature space usually results in very high dimensionality causing problems like over-fitting for many classifiers. Over-fitting is the effect that a classifier is trained too well on already pre-classified data and adjusts to too many details given by the large feature space; hence performs worse on unknown data, which has not been used for training the classifier. Therefore, a technique called feature selection is often applied to reduce dimensionality. This is explained in more detail in [Seb99]. However, research has shown that there is still a substantial amount of information in such words [Joa98] and therefore omission could result in loss of this information. Support vector machines distinguish themselves by over-fitting protection and their ability to deal with large feature spaces. Document pre-processing in terms of feature selection is therefore not required in case of support vector machine classifiers. The only pre-processing applied here is the filtering of unspecific words (and, or, a, etc.) using language specific stop-word lists. Such rather unspecific words also cause many classifiers to over-fit, since they occur frequently in every document and therefore outweigh other words in the word vector. Support vector machines seem to behave robust towards a multilingual environment, since no language

specific pre-processing (other than applying stop-word lists, which are available in several languages[29]) has to be performed.

In the work of Pache discussed in [Pac02], support vector machines have been applied to the multi-class, single-label case, where each document is assigned one out of n categories. The binary SVM approach has been adapted to this case by reducing the multi-class problem to binary problems, which can be solved by binary support vector machines. Basically, there are two possible versions how this adaptation can be performed, called One-Versus-All and One-Versus-One approach. In One-Versus-All, the n-dimensional multi-class problem is split into n binary classifiers, each deciding between one category versus all others. In One-Versus-One, one SVM is trained for each unordered pair of categories, resulting in $\frac{m*(m-1)}{2}$ support vector machines. Former research has shown, that this approach leads to better results (according to [WW99]) and has therefore also been applied in Pache's research. In order to assign a single category to a test document, the word vector of this document will be evaluated with all trained binary classifiers (support vector machines). Each binary classifier outputs a decision function value voting for one class or the other. The class, which has been decided for in most of the cases will be assigned. The extension to multiple labels can now simply be accomplished by creating a ranking of the binary classifier results and assign the categories with the highest rankings.

In addition to other evaluations, Pache evaluated the integration of generic background knowledge. Here, the terms of a word vector are extended with broader terms given by a generic controlled vocabulary. He reasoned even an improvement in quality using domain specific background knowledge, like the one provided by AGROVOC.

Support vector machines in general and particularly the One-Versus-One approach - since a large number of classifiers have to be trained - show bad time performance. Accuracy of prediction is, however, more critical in our case, since well-indexed documents provide the basis for good quality document retrieval. Support vector machines have shown to outperform other approaches regarding the quality of prediction as shown in [AE99].

Overall, SVMs and especially Pache's approach seem to be promising within the FAO environment. In the next section I will first define several measures needed for later

[29] See http://www.unine.ch/Info/clef/ for a listing of available stop-word lists in different languages.

performance evaluation before discussing the already indicated adaptations of Pache's classifier.

6.2.2 Evaluation measures:

The experimental evaluation of a classifier usually measures its effectiveness; i.e. its ability to take the right classification decision. Obviously, the most common strategy is to evaluate the performance of an automatic indexer against that of a human indexer. This is an error prone approach, since the assignment of categories is a subjective task and opinions of human indexers can substantially differ. The phenomenon also known as inter-indexer-inconsistency has been recognised in [Cle84]. Discrepancies can arise in the choice of categories as well as in the quantity of categories in case of multiple assignments of indexing terms (multi-label indexing). Nevertheless, since the only existing reference is the human indexing approach, the automatic output is evaluated against it.

In order to evaluate performance of a classifier, the initial document set X (pre-classified by human indexers) is split into a Training Document Set X_{Tr} and a Test Document Set X_{Te}, so that $X = X_{Te} + X_{Tr}$. The corpus of documents is pre-classified, i.e. the values of the function $\Phi : X \times C \rightarrow \{true, false\}$ are known for every pair (x_i, c_j). The classifier (in our case the Support Vector Machines) is built on the training document set and evaluated on the test document set.

The effectiveness of a classifier is usually measured using to common IR notions precision and recall. Precision and recall can be measured on three levels, document, class and global level respectively. They are defined slightly different on each level as done in the following according to [Seb99]:

Definition 15 (precision and recall, document level). On the **document level**, performance of the classification of a single document is measured. Table 4 shows the contingency table on the document level. Here, TP_i (true positives) is the number of classes/labels correctly assigned to the document wrt the set of all pairs (x_i, c_j) for a test document $x_i \in X_{Te}$. FP_i (false positives), FN_i (false negatives) and TN_i (true negatives) are defined accordingly.

Document x_i		Expert judgements	
		YES	NO
Classifier judgements	YES	TP_i	FP_i
	NO	FN_i	TN_i

Table 4: Contingency table for document x_i

Precision and recall can now be defined as follows:

$$precision = \frac{TP_i}{TP_i + FP_i} \qquad recall = \frac{TP_i}{TP_i + FN_i}$$

Precision is the probability, that if a label is assigned to a random document, this decision is correct whereas recall is the probability that if a label ought to be assigned to a document (according to the pre-classification), it will be assigned by the classifier. In other words, precision is calculated as the fraction of categories which have been correctly predicted divided by the total number of predicted categories and recall is the contingent of correctly predicted categories of the total number of pre-classified categories. Precision is hence a measure for how precisely the categories have been predicted by the classifier, whereas recall is a measure of how many of the pre-classified categories have been 'discovered' by the classifier.

Here and in the measures that will be further discussed, the pre-classified categories are the labels assigned by the human indexer. The measures precision and recall obviously take multi-label indexing into account. However, they do not take into account the above-mentioned discrepancy in the quantity of assigned labels. [Yan99] presents a more sophisticated measure called interpolated 11-point average precision addressing this problem. It would be interesting for future work to explore, if the results would differ substantially using this measure. Since this is not the main focus of this work, the measure has not been considered here.

The normal way to measure multi-class problems on the **class level**, is to break the problem down into disjoint binary problems:

Definition 16 (precision and recall, class level). On the class level, FP_i (false positives) are the number of documents incorrectly classified wrt class c_i. The other numbers are defined accordingly. Table 5 shows the contingency table for class c_i.

Class c_i		Expert judgements	
		YES	NO
Classifier judgements	YES	TP_i	FP_i
	NO	FN_i	TN_i

Table 5: Contingency table for class c_i

Precision and recall are calculated according to above formulas given in Definition 12. This time precision denotes the fraction of correctly predicted documents divided by the total number of documents predicted to that class. Recall measures how many of the documents, which have been pre-classified to that class have actually been predicted to it. These values are calculated for each class to measure the performance of the classifier on the class level. In [AE99] several other performance measures are discussed. Again, the above measures might vary significantly with the decision about the quantity of predictions for a document and therefore, measures taking this into account might be interesting for further evaluations.

Definition 17 (precision and recall, global level). On the global level, we have two different choices of summing up the lower level measures as discussed in [AE99]. In an approach called **macro-averaging**, the average mean of the class performances constitute the overall performance measure. **Micro-averaging** on the other hand sums up all the respective values first over all classes and then calculates above measures.

Category set $C = \{c_1,...,c_n\}$		Expert judgements	
		YES	NO
Classifier judgements	YES	$TP = \sum_{i=1}^{\|C\|} TP_i$	$FP = \sum_{i=1}^{\|C\|} FP_i$
	NO	$FN = \sum_{i=1}^{\|C\|} FN_i$	$TN = \sum_{i=1}^{\|C\|} TN_i$

Table 6: Global contingency table

The global contingency table (Table 6) is thus obtained by summing up over all category specific contingency tables (this can also be done by summing up over all documents). Formally, precision and recall can be calculated as follows:

Micro-averaging:

$$precision_{micro} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FP_i)} \qquad recall_{micro} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FN_i)}$$

Macro-averaging:

$$precision_{macro} = \frac{\sum_{i=1}^{|C|} precision_i}{|C|} \qquad recall_{macro} = \frac{\sum_{i=1}^{|C|} recall_i}{|C|}$$

The micro-averaging approach gives equal weight to each document, whereas macro-averaging gives equal weight to each class. A simple example shown in Figure 24 illustrates this:

> We are given 2 classes A and B. Figure 24 shows the number of correctly predicted documents and the overall number of predicted documents per class. The precision is calculated for each class separately. On the bottom, the two global level precision measures are calculated. It is obvious that the rather low precision of class two influences the overall measure much more in macro-averaging, since it gives equal weight to each class. However, micro-averaging takes into consideration the much larger number of documents in class A and better reflects the overall performance of the documents.

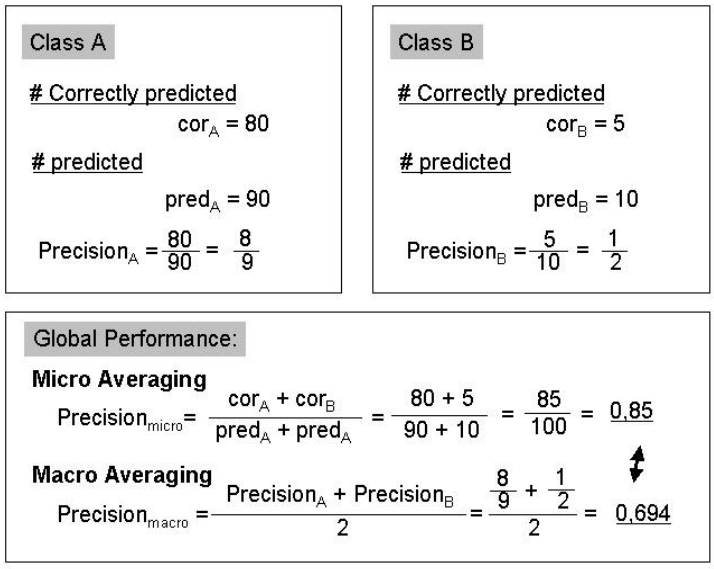


Figure 24: Example micro-averaging vs. macro-averaging

In our case, where each class could have a very varying number of pre-classified and predicted documents, the micro-averaging approach seems to lead to more objective and realistic results.

In addition to the above-mentioned discrepancy in quantity of assignment, the discrepancy in the choice of categories between indexers implies that a prediction of a category very similar to the one pre-assigned might not be completely wrong. The performance measures above do not take this into consideration, since it would be simply counted as a wrong prediction. An additionally calculated weighed precision and recall measure could deal with this problem. Here, the distance between a predicted category and all the pre-classified categories is calculated! This can only be done with the integration of background knowledge, providing the means for calculating a distance measure between categories and could be an interesting aspect for future evaluations.

In his work, Pache implemented the document and class level performance measures as well as micro-averaging on the global level, which will be used also in this evaluation. In the following, I will present the adaptations made to the system in order to allow the solving of the multi-class, multi-label indexing problem as introduced in Definition 11 and to allow the integration of domain specific background knowledge represented in the KAON language.

6.3 Adaptation of the classifier

6.3.1 Multi-label vs. single-label Indexing

The classifier has been extended to read in and process multi-label indexed documents and predict multiple labels. Training- and test-document registration is performed class by class, i.e. in each class, the documents pre-classified to that class are first separated into training documents and test documents. Then each document is registered as a training or test document in all the classes it is pre-classified to. This implies that one document can be training or test document for more than one class. In the training case, it can therefore happen that two classes are trained against each other with a partly overlapping document set, assuming that all the other documents provide enough necessary information in order to separate these classes. Several test runs could not confute this assumption. An additional mechanism has been implemented in order to assure that an equal number of training documents are registered for each class. In the test case, a mechanism has been added, allowing each unique document to be predicted only once. However, due to the multi-class assignments, the number of test documents slightly differs from class to class. Furthermore, in order to prevent that documents are always registered in the same order, the classes as well as the documents in the classes are shuffled in each evaluation run.

In Pache's former work, the best category is assigned to a test document (single-label case). This implies a ranking among the categories. This ranking is achieved by associating a score with each category regarding a certain test document. As explained above there are $\frac{m*(m-1)}{2}$ support vector machines, one for each unordered pair of classes. When testing a document, it is evaluated with each SVM. The SVM votes for the better class amongst the two it can choose from. A score value is then calculated for each class based on the positive votes for that class. The score is > 0 for a class, if more than 50% of its binary classifiers decided for it. It achieves the highest value, if all of a class's SVM vote for it. In the single-label case, the class with the highest score is assigned to the document.

The multi-label case is slightly more complex, since the decision is not only on which classes, but also on how many classes have to be assigned. The decision can unfortunately directly affect performance measures as already mentioned above. This becomes especially evident in deciding the number of indexing terms to assign to a document. Always assigning a too large number of terms to a document obviously results in a high recall, since every pre-classified term might be included. Precision will be presumably low in this case. On the other hand, assigning only the label with the highest probability will most probably result in high precision, but low recall, since most of the pre-classified terms are not predicted. Basically, two options exist to decide the number of labels to predict for a test document:

- Always predict a fixed number of terms for a document (i.e. the x terms with the biggest score)
- Predict an arbitrary number of terms for a document, based on a threshold value, (i.e. all terms with a score > x).

The choice depends on the training and test document set. If the indexers always picked a rather constant number of keywords, the first approach might lead to better results. If the number of keywords varies widely among the documents, the second approach might be more promising.

In the case of FAO documents, the number of keywords a document is indexed with varies substantially, ranging from as low as 3 up to as many as 20 (I will explain the compilation of the training and test document set in detail in the next section). Therefore, the second option is chosen in this adaptation. The score for a class is > 0, if more than 50% of the SVMs voted for this class. Therefore, 0 seems to be a reasonable starting threshold value, subject to variation depending on the achieved results. In order to prevent the assignment of too many labels, a maximum value, limiting the number of assigned labels, could be determined. A

straightforward approach for this would be to take the maximum number of labels having been assigned to any one document of the set of training documents. Another option, which has been chosen in this work, is to vary with the score threshold, until acceptable performance measures are reached. A score threshold of 0 presumably tends to assign too many labels (compared to the manually assigned number of labels), resulting in lower precision and higher recall than in the single-label case.

The first part of the evaluation of the adapted classifier aims on comparing the quality of multi-label classification against the single-label case. This is a difficult task due to the higher complexity of the multi-label case. The performance measures of the two cases are not completely comparable, hence only an attempt can be made here, which is subject to further testing and careful interpretation.

6.3.2 Multiple Languages

As reasoned in section 6.2.1, support vector machines are basically not dependent on extensive pre-processing in terms of feature selection and dimensionality reduction and could even perform worse, if pre-processing is applied (as outlined in [Joa01]). The classification of documents in languages different from English seems to be a promising option of support vector machines in the context of the FAO. Assuming, that the usage of a language specific stop-word list is the only adaptation, which has to be made in order to classify documents different from the English language, the performance should stay the same. The second focus of this work is therefore the evaluation of the classification of non-English language against English language documents.

6.3.3 Integration of background knowledge

Pache's central part of his work focused on the integration of background knowledge for classification. Integration of background knowledge in this context means the use of a-priori knowledge about statistical probabilities of word correlation. Two types of background knowledge serve for integration:

Generic Background knowledge

Generic background knowledge contains generic, lexical knowledge, which is independent from the domain of the classification problem. The publicly available thesaurus WordNet[30],

[30] Refer to http://www.cogsci.princeton.edu/~wn/ to obtain a full version of the WordNet.

which has been used in Pache's work, is an example of generic background knowledge. It contains synonym, hypernym and hyponym[31] relations of the generic English language.

Domain specific background knowledge

One problem of generic background knowledge is the ambiguity of the senses in which a word might be used in a certain context. Domain specific background knowledge might lower this problem, since it contains words specifically representing this domain. Words are usually used and referred to in a certain sense in this domain. Domain specific ontologies as created within the scope of this project represent such domain specific background knowledge.

In this work, the classifier has been enhanced to be able to integrate ontologies represented in the KAON language, such as the converted AGROVOC ontology. The integration of background knowledge into the classification problem can be accomplished in two ways. In the first one, the class hierarchy structure is built using the taxonomy of the ontology. This option is disregarded here. I will explain this in more detail in the following section. The second integration option is given by extending the word vector of a document with related concepts, extracted from the background knowledge by using word-concept mappings and exploring concept relationships. In [Pac02] only hypernym relations of the WordNet dictionary have been used to accomplish this. Since WordNet is a generic dictionary of the English language, the extension of the word vectors presumably has been performed with many words, relatively unspecific for this document.

The integration of domain specific background knowledge bares a certain potential, in that it only extends the word vector of a document with domain specific concepts, relevant to the classification domain. Very specific words obviously occur less frequent in documents. The inclusion of related concepts to such words seems promising, since such word vector extensions are not producing noise for the SVM and should therefore move the word vector towards the direction of the right classification.

Hypernym Disambiguation

When integrating generic background knowledge, a term occurring in a document might be represented in many different senses in the generic vocabulary. Consider for example the term 'branch'. Branch can refer to the branch of a tree or the branch of a company. Additional

[31] The terms hypernym and hyponym are in accordance with super-concept and sub-concept.

algorithms, as implemented in case of integration of WordNet in Pache's former work, are needed in order to resolve this ambiguity. In case of integrating domain specific background knowledge as given for example by the AGROVOC ontology, we do not face this problem, since the vocabulary here is very specific and each term is used in a particular sense only. I therefore abstained from an implementation of word disambiguation in case of integrating specific background knowledge.

Using an ontology as background knowledge, several levels of word-vector extension are possible:

- Inclusion of super- and sub-concept hierarchy (up to a maximum depth)
- Inclusion of arbitrary related concepts (up to a maximum depth)
- Inclusion with variations in depth

Differentiated treatment of the kinds of relationships provides another dimension, which could be evaluated. In this adaptation and application, all relationships are treated the same way, since the AGROVOC with mainly related term relationships is used as a domain specific ontology. However, in other ontologies, there might be many different relationships with varying semantics and therefore weighs could be introduced in order to vary the depth of inclusion. Here, super-concepts and arbitrary related concepts can be included up to a maximum depth, which can be chosen.

Extending a document's word vector with background knowledge means to extend it with the concepts found in the document and related concepts of those. The word-vector's dimension rises by the number of concepts included. The values are the weighed frequencies of occurrence of the concept and related concepts. An important question therefore concerns the lexicalisation of concepts, i.e. the word-concept mapping. Which level of matching of the lexicalisation of a concept with a given word in a document is needed in order to include a concept to enhance the word vector with? Basically, two options exist for this decision:

- Include and count concept, if part of its lexicalisation (label or synonym) is found (chosen here)
- Include and count concept, only if an exact match of a concept's lexicalisation (label or synonym) appears in the document

The first option might easily count a concept too many times. Consider the concept with the label 'animal health'. The concept will be included and counted, if either the word 'animal' or

the word ‘health’ occurs in the document. Obviously, it is counted twice, if the complete lexicalisation occurs in the document. The second option prevents this ‘over counting’ but is however more complicated to implement and degrades the performance of the system. Moreover, it might ‘forget’ to include a concept. A compound word might be written slightly different in a document than stored in the background knowledge. The check for an exact match is therefore likely not to integrate that concept, which would be found in the first case. Taken into account, that the integrated background knowledge is domain specific and that very specific vocabulary does not occur extremely often in a document, we can well ignore the fact of ‘over counting’. It is even possible, that this approach supports the SVM performance, since the domain specific words are given an even higher weight. Therefore, the first option has been implemented here.

The integration, i.e. the word-vector extension, is done for training and test documents respectively. Three different options for concept integration are possible (the words in brackets are the abbreviations for the concept integration mode used in the following):

- Adding all concepts to the word vector (add)
- Substituting the words with their matching concepts (replace)
- Consider only the related concepts (deleting out all other words from the word vector) (only).

When testing a document, only related concepts of words constituting the dictionary of the words of the training documents are taken in order to enhance the word vector of the test document. As an additional option, related concepts of words, which are not found in the dictionary, can be added to the word vector. The consideration of this option originates from the possibility that the synonym of a word might be in one of the training documents and therefore can only be realised through the common concept!

In [Pac02], no better performance of the SVM could be achieved in the single-label case by integrating generic background knowledge. The 3rd focus of this part of the thesis therefore evaluates the hypothesis that the integration of domain specific background knowledge is more promising and could lead to overall better classification results.

6.3.4 Multi-class problem and class hierarchy

In a multi-class problem, document classes can be arranged in 3 ways:

- Flat:

  The document classes build a flat structure and each class is trained against all other classes

- Simple Hierarchical Tree Structure:

  The document classes build a tree structure and on each level of the tree, a class is trained against all subclasses of this class, beginning at the root of the tree

- Multiple Inheritance Tree Structure:

  The document classes can build not only a simple tree, but any graph without circles and one root node

Former evaluations have shown, that the hierarchical ordering of the classes can even lead to worse results [Pac02]. This can easily be explained: Once, a document is in a branch of a tree, there is no way to come back to the other branch of the tree again. The only advantage of the tree structure is actually an expected performance gain, since in the best case of a binary tree, the effort can be reduced by the factor log(n) [Pac02]. However, evaluations have shown that the gain in reality is much less, and since performance is not the main goal of this work but rather exact predictions, only the flat hierarchy for document classes has been implemented for the multi-label case.

Training can be performed according to above structure in an either flat mode or hierarchical mode. In the hierarchical mode, each class is trained against its subclasses respectively, whereas in the flat mode, each class is trained against all others in One-Versus-One mode. That means that in the flat mode all classes are trained against each other.

Testing, i.e. prediction of a class can similarly be accomplished in a hierarchical and non-hierarchical way. Three different options are possible and implemented in the classifier. In case of the flat mode, the best class is chosen among all the classes. In case of a hierarchical ordering, two cases are possible: On each level, always choose the best class and then recursively apply this, until no better score is achieved or a leaf is reached (ONEPATH). The second option (MULTIPATH) evaluates the whole tree (not only the best path on each level) and then chooses the overall best class.

The integration of background knowledge, as discussed above, could serve to provide the hierarchical ordering for hierarchical classification. In our case, the AGROVOC constitutes the hierarchical order of the categories, with which the documents are indexed. Since this

thesis' main aim is not on performance issues, the adaptation of hierarchical classification to incorporate an ontology hierarchy has not been considered here.

6.4 Set of training and test documents

All evaluations of the adapted classifier are conducted on documents of the agriculture sector, pre-classified by the FAO. The FAODOC database, as discussed before in chapter 2, provides the most reliable set of metadata on documents and the indexing done here can be considered most consistent and correct apart from the indexer-indexer inconsistency. Consequentially, the part of FAODOC resources, which is available in the document repository, is a good source for compiling a training and test document set.

Resources are indexed on two different levels in the FAODOC database: the analytical level and the monographic level. On the analytical level, each single article is indexed. Resources on this level are full text documents associated with their index labels. On the monographic level, journals, proceedings and similar resources (containing a set of articles) are indexed. Resources on this level usually contain a brief description, i.e. editorial, of the whole journal together with the table of contents. A result of this two-levelled indexing is obviously an extremely heterogeneous test set, differing substantially in size and descriptive content of its documents.

The metadata elements of the resources furthermore contain a subject element and a category element besides others like title, URL, etc. Subject indexing is carried out using keywords from the AGROVOC thesaurus, hence over 16607 potential labels can be chosen from. At maximum 6 primary descriptors can be used to index a document, describing the most important concepts. Additionally, an indefinite number of secondary descriptors can be chosen, as well as geographic descriptors (for example country information). Only the primary descriptor associations have been considered in this evaluation. Besides AGROVOC descriptors, each resource belongs to a maximum of 3 categories. The categories are chosen from the set of 115 subject categories as described before in Chapter 4. A complete listing of all 115 categories is attached in Appendix C. All this information is stored in any of the three languages English, French and Spanish.

Given this organisation, 6 different test document sets can be created, associating documents with AGROVOC descriptors or categories in any of the three languages. The test set has been retrieved as follows: First, a mapping of document titles to their URLs into the Document Repository and their AGROVOC descriptors has been retrieved for all electronically available resources (analytical and monographic level) separately in English,

French and Spanish. The same has been done for the category assignments. Problems with broken and badly maintained links as well as way the links are stored forbid the retrieval of an extensive and 100% correct test set. Time restrictions and the large amount of documents did not allow for manually checking each link for correctness and hence there might be some wrong document classification label associations in the final test sets. Table 7 shows the statistics about the 6 retrieved document sets after this first step in the compilation process.

Test Set X_{raw} Statistics		**Language**					
		English (en)		**French (fr)**		**Spanish (es)**	
		Desc	**Cat**	**Desc**	**Cat**	**Desc**	**Cat**
Total	# Documents	1708	1879	481	897	519	769
	# Classes	1185	118	503	86	511	93
	# Labels	5072	3328	1494	1620	1574	1434
Class level	Max ($\frac{\#documents}{class}$)	96	315	67	214	71	179
	Min ($\frac{\#documents}{class}$)	1	1	1	1	1	1
	Avg ($\frac{\#documents}{class}$)	1,44	15,92	0,95	10,43	1,02	8,27
Docu-ment level	Max ($\frac{\#labels}{document}$)	8	3	7	4	7	7
	Min ($\frac{\#labels}{document}$)	0	1	1	1	1	1
	Avg ($\frac{\#labels}{document}$)	2,97	1,77	3,11	1,81	3,03	1,86

Table 7: Raw test document set for automatic text classification, X_{raw}

The retrieved documents differ widely in length and style, caused by the two different levels of indexing as well as by the content of the articles themselves. The size of the documents ranges from as little as 1.5 KB up to sizes of over 600 KB, creating a challenging test set for an automatic classifier. In this sense, the here used data set differs substantially from the widely used Reuters[32] data set ([Lew99]), which contains rather standard length documents written in similar styles. Especially the amount of possible categories in case of the AGROVOC descriptors significantly exceeds the amount of categories used in the Reuter's data set, where only 103 topic codes served for categorisation in the big RCV1 Reuters collection described in [RSW02].

[32] The Reuters-21578 test set is available at: http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html.

The data in Table 7 shows clearly, that that further compilation of the retrieved raw document sets is necessary. In his former work, Pache varied between 5 up to 100 documents per class in order to train it and left the test document set X_{Te} constantly at a number of 50 test documents per class. In order to achieve comparable results and to avoid effects resulting from unequal distribution, the training set X_{Tr} of each class should be approximately equal in size. None of the above preliminary sets fulfils this requirement.

The raw document sets have been compiled with the requirement to have at least 50 documents per class, where possible. All classes, which could not fulfil this requirement, were deleted from the above sets along with the documents belonging to those. Due to the small number of available documents in French and Spanish, which are indexed with AGROVOC keywords, this was not achievable in those 2 cases. Only one document set could be created out of the English language document category associations, providing a minimum of 100 documents per class. Table 8 shows the overview of the so compiled test set.

Test Set X_{multi} Statistics		**Language**					
		English (en)		**French (fr)**		**Spanish (es)**	
		Desc	**Cat**	**Desc**	**Cat**	**Desc**	**Cat**
Total	# Documents	464	1016	186	698	230	563
	# Classes	8	7	6	9	9	7
	# Labels	541	1272	235	979	309	797
Class Level	Max ($\frac{\#documents}{class}$)	96	315	67	214	71	179
	Min ($\frac{\#documents}{class}$)	51	108	30	58	20	58
	Avg ($\frac{\#documents}{class}$)	58	145,14	31	77,56	25,56	80,43
Docu-ment level	Max ($\frac{\#labels}{document}$)	4	3	3	3	4	3
	Min ($\frac{\#labels}{document}$)	1	1	1	1	1	1
	Avg ($\frac{\#labels}{document}$)	1,17	1,25	1,26	1,40	1,34	1,42

Table 8: Compiled test document set X_{multi} (multi-label)

In order to compare single-label vs. multi-label classification as discussed in the previous section, we still need a test set with documents only pre-classified to one class respectively. In order to compare the results, the documents and classes used for both evaluations should be the same. Therefore, another test set has been compiled from X_{multi} deleting from each

document all except the first pre-assigned label (AGROVOC descriptor or category). The same requirements regarding the minimum number of documents per class have been applied to compile the set shown in Table 9.

Test Set X_{single} Statistics		Language					
		English (en)		French (fr)		Spanish (es)	
		Desc	Cat	Desc	Cat	Desc	Cat
Total	# Documents	374	1016	117	612	188	563
	# Classes	6	7	3	7	6	7
	# Labels	374	1016	117	612	188	563
Class Level	Max ($\frac{\#documents}{class}$)	86	271	55	171	56	158
	Min ($\frac{\#documents}{class}$)	51	102	30	50	21	50
	Avg ($\frac{\#documents}{class}$)	62,33	145,14	39,0	87,43	31,33	80,43
Docu-ment level	Max ($\frac{\#labels}{document}$)	1	1	1	1	1	1
	Min ($\frac{\#labels}{document}$)	1	1	1	1	1	1
	Avg ($\frac{\#labels}{document}$)	1	1	1	1	1	1

Table 9: Compiled test document set X_{single} (single-label)

Due to the small number of documents in the French and Spanish sets compiled from the AGROVOC descriptor assignments, these four sets will not be considered in any of the following evaluations, reducing the number of possible test sets to 8.

Table 10 shows an overview of the respective descriptors and categories, the documents in the different sets have been indexed with. Obviously, there is a certain tendency towards similarity between the labels, which have been preferentially assigned to documents. In the descriptor set there seem to be lots of documents of the forestry sector, whereas in the category sets, many documents have been assigned categories from the E or K main categories (see Appendix C for a complete category listing).

	Language					
	English (en)		French (fr)		Spanish (es)	
	Desc	Cat	Desc	Cat	Desc	Cat
X_{single}	EXTENSION ACTIVITIES, FOREST MANAGEMENT, FOREST RESOURCES, FORESTRY DEVELOPMENT, SUSTAINABILITY, TRIFOLIUM REPENS	E14, E50, E70, K01, K10, M11, P01	FOREST MANAGEMENT, FORESTRY, FORESTRY DEVELOPMENT	E10, E14, E50, M11, K01, K10, P01,	FOREST MANAGEMENT, FOREST RESOURCES, FORESTRY, FORESTRY DEVELOPMENT, FORESTRY POLICIES, NONWOOD FOREST PRODUCTS	E10, E14, E50, E71, K01, K10, P01,
X_{multi}	EXTENSION ACTIVITIES, FAO, FOOD SECURITY, FOREST MANAGEMENT, FOREST RESOURCES, FORESTRY DEVELOPMENT, SUSTAINABILITY, TRIFOLIUM REPENS	E14, E70, E50, K01, K10, M11, P01	FOREST MANAGEMENT, FORESTRY, FORESTRY DEVELOPMENT , FORESTRY POLICIES, NONWOOD FOREST PRODUCTS, STAINABILITY,	E10, E14, E50, E70, E71, M11, K10, K01, P01,	FAO, FOOD SECURITY, FOREST MANAGEMENT, FOREST RESOURCES, FORESTRY, FORESTRY DEVELOPMENT, FORESTRY POLICIES, NONWOOD FOREST PRODUCTS, SUSTAINABILITY	E10, E14, E50, E71, K01, K10, P01

Table 10: Overview about the classes of the test document sets

A number of evaluation runs have been applied to these 12 final test sets. The settings and results will be discussed in the following.

6.5 Evaluation

6.5.1 Single-label vs. multi-label classification

The first evaluation aims at comparing the newly implemented multi-label classification against the formerly evaluated single-label case. Performance is measured on the global level for all of the following evaluations. Both precision and recall are measured and calculated using micro-averaging. Only the English document sets $X_{single\_en\_Cat}$, $X_{single\_en\_Desc}$, $X_{multi\_en\_Cat}$ and $X_{multi\_en\_Desc}$ have been chosen for this first evaluation, since they provide the most extensive test sets. The number of training examples per class has been varied from 5 up to 50 for the sets compiled from the AGROVOC descriptor assignments and up to 100 for the sets resulting from the categories. The number of test examples has been held at a constant rate of 30 (for the sets indexed with AGROVOC descriptors) and 50 (for the sets indexed with categories) test documents per class. A word pruning threshold deletes out all words from a document's word vector, which appear less than x times in all documents. This value has been set to 3 and 10 for the descriptor sets and additionally to 50 for the category sets (due to the

larger document volume of the corpus). Each parameter setting has been applied in 15 test runs. In each test run, the documents in each class are shuffled and therefore split into different disjoint sets of training and test documents in each run. Additionally, the classes are shuffled in the multi-label case as described in section 6.3.1.

Table 11 shows the results of the single label document sets. The values are the averaged precision over 15 test runs respectively. In the single-label case, precision and recall do not differ; hence recall is not displayed here.

Avg(Precision)		**TrainingEx**									
TestSet	P	**5**	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**100**	**Total**
Cat	**3**	0,4297	0,5143	0,5859	0,6270	0,6400	0,6676	0,6626	0,6716	0,6676	0,6074
	10	0,4406	0,5337	0,6002	0,6217	0,6484	0,6551	0,6591	0,6596	0,6800	0,6082
	50	0,4749	0,5530	0,5966	0,6253	0,6389	0,6579	0,6517	0,6665	0,6691	0,6148
Cat total		0,4484	0,5337	0,5942	0,6247	0,6424	0,6613	0,6578	0,6659	0,6722	0,6102
Desc	**3**	0,5559	0,6304	0,6744	0,6907	0,7033	0,7041				0,6598
	10	0,5763	0,6281	0,6752	0,6970	0,7002	0,7163				0,6655
Desc total		0,5661	0,6293	0,6748	0,6939	0,7018	0,7102				0,6627

Table 11: Single-label classification on English documents sets; word pruning threshold vs. variation of training examples per class; average precision over 15 test runs for each configuration

In order to test whether the variation of the word pruning threshold (P) creates significantly different results, several Student's T-Tests have been conducted against the usual confidence interval 5%. Refer to [Tro02] for an introduction to T-Tests. The 0 hypothesis against which is tested is that the means of the data samples are equal. Given the data above, the hypothesis could only be rejected for low training example counts. In other words, it could only be shown that with a probability of 95% the variation of the pruning value results in a significant improvement of the average precision, in case of variation of the training examples between 5 and 20. The same results have been received, applying the T-Test to the single label test sets in French and Spanish (the evaluation of these is discussed in the next section).

Since the variation in the number of training examples obviously creates a bigger effect on improving the average precision, I will not specifically focus on variations of the word pruning value in the further discussion.

Interesting is, however, the difference of performance between the category set and the descriptor set. The descriptor set shows significantly higher performance. Obviously, documents can be more clearly separated by their AGROVOC descriptors than by their categories. Taking into consideration, however, that a category is broader than a descriptor in the sense that many descriptors can be mapped to one category, this result seems very reasonable.

In the next test setting I therefore first considered the set $X_{multi\_en\_Desc}$ in order to evaluate the performance of the multi-label classification approach. In addition to above variations, the score threshold has been varied between 0 and 0,6 in order to vary the number of automatically assigned labels to a document. Table 12 shows the results of this evaluation.

Average values		Training Examples						
Score Threshold	**Measure**	**5**	**10**	**20**	**30**	**40**	**50**	**Total**
0,0	Precision	0,2592	0,2658	0,2715	0,2745	0,2728	0,2727	0,2694
	Recall	0,8401	0,8707	0,9056	0,9226	0,9323	0,9329	0,9007
	Breakeven	0,5497	0,5683	0,5886	0,5986	0,6026	0,6028	0,5851
0,1	Precision	0,2614	0,2693	0,2737	0,2748	0,2733	0,2754	0,2713
	Recall	0,8525	0,8826	0,9122	0,9249	0,9291	0,9350	0,9060
	Breakeven	0,5569	0,5759	0,5929	0,5999	0,6012	0,6052	0,5887
0,3	Precision	0,3039	0,3201	0,3329	0,3423	0,3408	0,3412	0,3302
	Recall	0,7402	0,7896	0,8327	0,8604	0,8622	0,8721	0,8262
	Breakeven	0,5221	0,5549	0,5828	0,6014	0,6015	0,6066	0,5782
0,5	Precision	0,3841	0,4160	0,4353	0,4476	0,4501	0,4492	0,4304
	Recall	0,6194	0,6759	0,7188	0,7504	0,7581	0,7618	0,7141
	Breakeven	0,5018	0,5460	0,5770	0,5990	0,6041	0,6055	0,5722
0,6	Precision	0,3839	0,4121	0,4369	0,4444	0,4475	0,4539	0,4298
	Recall	0,6197	0,6713	0,7230	0,7399	0,7512	0,7702	0,7126
	Breakeven	0,5018	0,5417	0,5799	0,5922	0,5994	0,6121	0,5712
Total: Precision		0,3185	0,3367	0,3501	0,3567	0,3569	0,3585	0,3462
Total: Recall		0,7344	0,7780	0,8185	0,8397	0,8466	0,8544	0,8119
Total: Breakeven		0,5264	0,5573	0,5843	0,5982	0,6017	0,6064	0,5791

Table 12: Performance of multi-label classification with English document set $X_{multi\_en\_Desc}$, average performance measures over 30 test runs

This time, precision and recall are both displayed, since they differ from each other in the multi-label case as opposed to the single-label case. The values have been computed as the average mean of 30 test run results (15 for each word pruning value of 3 and 10 respectively). As expected, the precision of the prediction is low in the beginning, since the classifier predicts multiple labels for each document but the numbers assigned by the classifier might be significantly higher than those assigned by the human indexer in many documents. Thus only a small part of the automatic classifications is correct. The recall on the other hand is high in the beginning, since a low score threshold tends to predict too many labels for a document. The probability that the right labels are among the assigned is therefore high. Increasing the score threshold resulted in increasing precision and decreasing recall. This implies that by assigning fewer labels to a document, some of the incorrectly assigned labels are obviously filtered out; hence increasing precision. However, some correct labels are also not predicted anymore, hence decreasing the overall recall.

In order to compare these contradictory effects, the breakeven value displayed in Table 12 is computed as the average mean of precision and recall (this calculation has been adopted from [Pac02]). Considering only this total measure, no significant change in the overall performance could be achieved, varying the score threshold (the total values are all very close to 0,58). This effect is visualised more concisely in Figure 25, where the development of the breakeven values is displayed for each different score threshold.

Figure 25: Development of breakeven for test set $X_{multi\_en\_Desc}$

An interesting observation is a positive effect on both precision and recall and consequently on the overall breakeven resulting from increasing the threshold from 0,0 to 0,1. In case of the document set $X_{multi\_en\_Desc}$, this score threshold produces the best overall results. In Figure 26 the significant increase in precision (and herewith decrease in recall) becomes evident more clearly.

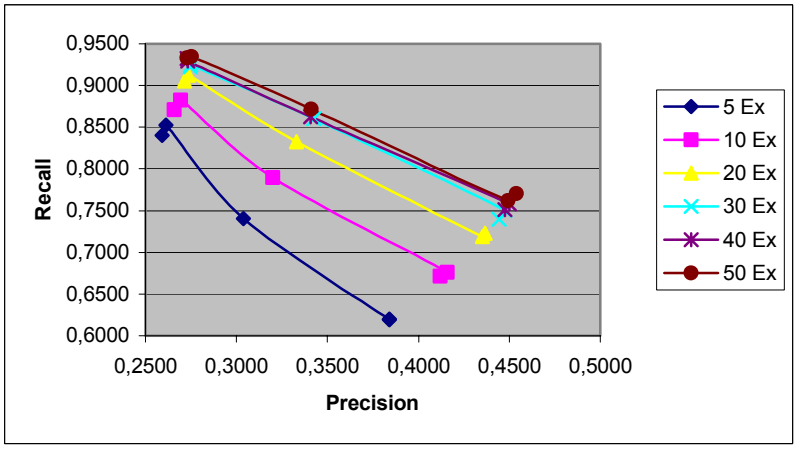


Figure 26: Precision vs. Recall for test set $X_{multi\_en\_Desc}$

The figure shows the development of precision and recall depending on variation of the score threshold for each number of training examples. The overall superior configuration of the threshold 0,1 can be seen in both figures. The precision-recall graph shows again the positive correlation between number of training examples and overall performance. The precision-recall curves elevate to higher levels, increasing the number of training examples, however, with decreasing margins. If compared to the results achieved with the single label test set $X_{single\_en\_Desc}$ in Table 11, the overall performance is worse. However, this could be expected since the multi-label classification problem is more complex than the single label case, even for a human indexer!

The evaluation runs on the multi-label test set $X_{multi\_en\_Cat}$ take longer time due to the higher number of tested documents and the different document pre-processing than in the single-label case. Therefore, this test set has only been considered exemplary with 5 training examples per class varying the score threshold as before from 0,1 to 0,6. Table 13 shows that the findings of above are equally supported in this case. Especially the increase of the score threshold from 0,0 to 0,1 again causes better overall performance. This time, however, the overall performance significantly decreases with each further increase of the score threshold. A significantly decreasing recall value causes this effect.

Average values		**Training Examples**
Score Threshold	**Measure**	**5**
0,1	Precision	0,3308
	Recall	0,7289
	Breakeven	0,5299
0,1	Precision	0,3333
	Recall	0,7294
	Breakeven	0,5314
0,3	Precision	0,4072
	Recall	0,5981
	Breakeven	0,5026
0,5	Precision	0,4022
	Recall	0,5871
	Breakeven	0,4947
0,6	Precision	0,5015
	Recall	0,3681
	Breakeven	0,4348
Total: Precision		0,3950
Total: Recall		0,6023
Total: Breakeven		0,4987

Table 13: Performance of multi-label classification with English document set $X_{multi\_en\_Cat}$, average performance measures over 15 test runs

The same test settings have finally been applied to the French and Spanish test sets $X_{multi\_fr\_Cat}$ and $X_{multi\_es\_Cat}$ aiming on approval of the conclusions made on variation of the score threshold. A comparison of performance between the different languages is addressed in a separate evaluation in the following section. The results of the multi-label evaluation with the French and the Spanish sets are shown in Tables 14 and 15. Here and in the following, a value with the superscript * means, that the respective result has been computed from less than 15 values, due to test run aborts or file corruption.

Average values		Training Examples						
Score Threshold	**Measure**	**5**	**10**	**20**	**30**	**40**	**50**	**Total**
0,0	Precision	0,2920	0,3084	0,3231	0,3314	0,3321	0,3314	0,3197
	Recall	0,7617	0,8033	0,8518	0,8801	0,8887	0,8931	0,8464
	Breakeven	0,5268	0,5559	0,5874	0,6058	0,6104	0,6123	0,5831
0,1	Precision	0,2981	0,3081	0,3261	0,3307	0,3323	0,3373	0,3221
	Recall	0,7721	0,8053	0,8585	0,8693	0,8814	0,8949	0,8469
	Breakeven	0,5351	0,5567	0,5923	0,6000	0,6069	0,6161	0,5845
0,3	Precision	0,3196*	0,3651	0,3923	0,4009	0,4051	0,4095*	0,3906
	Recall	0,6161*	0,7044	0,7697	0,7986	0,8044	0,8109*	0,7689
	Breakeven	0,4679*	0,5348	0,5810	0,5998	0,6048	0,6102*	0,5797
0,5	Precision	0,4099	0,4572	0,4756	0,5035*	0,5163	0,5192*	0,4762
	Recall	0,5230	0,5926	0,5947	0,6552*	0,6836	0,6493*	0,6127
	Breakeven	0,4665	0,5249	0,5352	0,5794*	0,6000	0,5842*	0,5444
0,6	Precision	0,4003	0,4521	0,4953*	0,4721*	0,5199	0,5008*	0,4669
	Recall	0,5124	0,5783	0,6271*	0,5730*	0,6791	0,6167*	0,5970
	Breakeven	0,4563	0,5152	0,5612*	0,5225*	0,5995	0,5587*	0,5320
Total: Precision		0,3482	0,3782	0,3973	0,3937	0,4212	0,3870	0,3884
Total: Recall		0,6407	0,6968	0,7468	0,7921	0,7874	0,8253	0,7457
Total: Breakeven		0,4944	0,5375	0,5720	0,5929	0,6043	0,6062	0,5670

Table 14: Performance of multi-label classification with Spanish document set $X_{multi\_fr\_Desc}$, average performance measures over 30 test runs

The French set completely supports all of the above findings, whereas at first glance the Spanish doesn't looking at the total numbers. Taking, however, into consideration that some of the values in Table 15 for the score threshold 0,0 were computed from less than 15 test runs, the same statements can be derived here. Especially in the case of 5 training examples, only 2 values were used, giving this value much lower weight in the total.

Average values		Training Examples						
Score Threshold	**Measure**	**5**	**10**	**20**	**30**	**40**	**50**	**Total**
0,0	Precision	0,3281*	0,4027	0,4162*	0,4227*	0,4209*	0,4413	0,4195
	Recall	0,6267*	0,7576	0,8022*	0,8243*	0,8261*	0,8532	0,8094
	Breakeven	0,4774*	0,5802	0,6092*	0,6235*	0,6235*	0,6473	0,6144
0,1	Precision	0,3869	0,4054	0,4190	0,4254	0,4289	0,4375	0,4172
	Recall	0,7283	0,7696	0,8024	0,8215	0,8337	0,8593	0,8025
	Breakeven	0,5576	0,5875	0,6107	0,6235	0,6313	0,6484	0,6098
0,3	Precision	0,4302	0,4845	0,5200	0,5381	0,5314	0,5383	0,5071
	Recall	0,5329	0,6123	0,6700	0,6768	0,6923	0,7038	0,6480
	Breakeven	0,4815	0,5484	0,5950	0,6074	0,6119	0,6211	0,5775
0,5	Precision	0,4278	0,4830	0,5081*	0,5182*	0,5378	0,5421*	0,4980
	Recall	0,5371	0,6078	0,6319*	0,6503*	0,6761	0,6938*	0,6263
	Breakeven	0,4824	0,5454	0,5700*	0,5843*	0,6069	0,6179*	0,5621
0,6	Precision	0,5160	0,5905	0,6677	0,6763	0,6874	0,6958	0,6389
	Recall	0,3249	0,3742	0,4258	0,4352	0,4473	0,4564	0,4106
	Breakeven	0,4205	0,4824	0,5467	0,5557	0,5673	0,5761	0,5248
Total: Precision		0,4384	0,4732	0,5087	0,5202	0,5240	0,5300	0,5000
Total: Recall		0,5324	0,6243	0,6640	0,6784	0,6915	0,7150	0,6530
Total: Breakeven		0,4854	0,5488	0,5864	0,5993	0,6078	0,6225	0,5765

Table 15: Performance of multi-label classification with Spanish document set $X_{multi\_es\_Desc}$, average performance measures over 30 test runs

If the same values were achieved with the same amount of 15 test examples, the respective totals for the threshold 0,0 would all be lower than the totals for 0,1. A more wondrous effect, which cannot be explained the same way is the extremely good precision achieved in case of the Spanish test set. The precision achieved here even outperforms the precision achieved for the single label test with the Spanish set as shown in the next section.

Recapitulating the results above, multi-label classification has shown overall worse performance than the single-label case. Figure 27 visualises the total results of the single-label cases vs. the best parameter configurations of the multi-label cases. The difference comparing the overall results between the two approaches is, however, reasonable. In respect of the higher complexity of the multi-label problem, the results are even surprisingly good. Regarding performance of different languages, we can already infer from the multi-label results, that languages different from English seem to perform equally well. Surprisingly, the Spanish set even outperforms the other two.

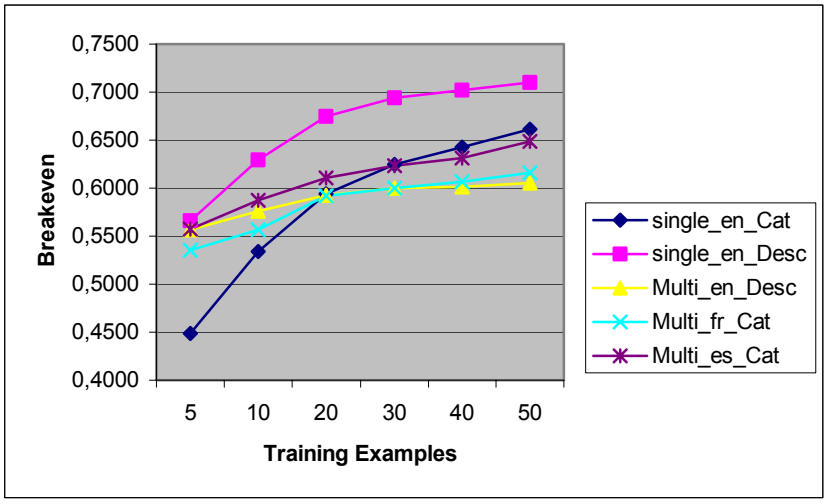


Figure 27: Single-label vs. multi-label classification: Comparison of overall performance

A score threshold of 0,1 consistently outperformed the value of 0,0. Even though, Student's T-Tests against the confidence interval 5% as conducted in the last section could not hold this statement, the continuity across all the tests seems to provide enough evidence. No clear statement can be made, however, on further raising this threshold. It depends on the intended goal of applying the classifier. If the classifier is used to help a human indexer suggesting a large set of possible index terms, from which the indexer can choose, then it is clearly of advantage to have a high recall, suggesting most of the 'good' terms amongst others. If, however, the automatic classifier is used without human support, one naturally wants to limit the risk of assigning wrong labels, hence thriving for a high precision. In the latter case, a higher score threshold should be chosen.

6.5.2 Multilingual classification

The second evaluation has been motivated by the idea that support vector machines are basically indifferent towards languages and document representations. The simplest possible scenario is a classifier that, given an arbitrary document, decides for one of the 3 classes English, French or Spanish. To accomplish this, a very simple document set has been compiled out of the single-label category test sets given in Table 9. $X_{single\_en\_Cat}$, $X_{single\_fr\_Cat}$ and $X_{single\_es\_Cat}$ each pre-classified to its corresponding language class (en, fr, es). Each class contains more than 500 documents, providing more than enough documents for this first test scenario. The classifier has been trained varying the number of training documents per class between 5 and 200, leaving the number of test documents at a constant rate of 100 test documents per class. The word pruning threshold has been set to 10 and 50 respectively.

Word stemming has not been applied and one big stop-word list has been compiled out of the respective ones existing in each language. Table 16 shows the precision values averaged over 15 evaluation runs for each parameter setting.

Avg(Precision)	Pruning Threshold		
Training Examples	10	50	**Total**
5	0,9958	0,9933	0,9946
10	0,9951	0,9924	0,9938
20	0,9960	0,9929	0,9944
30	0,9960	0,9947	0,9953
40	0,9964	0,9953	0,9959
50	0,9971	0,9969	0,9970
100	0,9962	0,9958	0,9960
200	0,9958	0,9944	0,9951
Total	0,9961	0,9945	0,9953

Table 16: Average precision results of simple language classifier

Obviously, support vector machines are able to almost perfectly distinguish between languages. A pruning value of 10 in this case leads to even better results. The difference is, however, negligible. This test run showed evidence that the support vector machine classifier used here can handle documents in different languages.

A more challenging scenario considers the single-label test sets $X_{single\_fr\_Cat}$, $X_{single\_es\_Cat}$ (the test sets compiled from the AGROVOC descriptors has not been considered here, due to the low amount of documents per class). A second evaluation setting tested, whether the classifier's overall performance is language independent. For this purpose the same configurations as in the evaluation with the English document set have been applied. However, due to the smaller number of documents per class, the training documents have been varied from 5 to 60 only, leaving the number of test documents constantly at a rate of 30. Words occurring less than 3 (10) times in all training documents have been filtered out.

	Training Examples							
Language	**5**	**10**	**20**	**30**	**40**	**50**	**60**	**Total**
English	0,4351	0,5240	0,5930	0,6244	0,6442	0,6636	0,6608	0,5893
French	0,4702	0,5375	0,5873*	0,6081	0,6260	0,6358	0,6383	0,5861
Spanish	0,4312	0,5000	0,5659	0,5802	0,5863	0,6043	0,5898	0,5561

Table 17: Average precision of single label test runs in all 3 languages

Table 17 shows the precision values averaged over the two word pruning parameters and 15 evaluation runs (with a different, disjoint set of training and test documents in each run). Obviously, above made hypothesis that support vector machines behave robust towards different language representations seems to hold well. Between the English and the French

set, the results show no significant difference in performance. Only the Spanish set is consistently outperformed by the two other sets in each parameter configuration. There could be several reasons for this. On the one hand, the document set retrieved in Spanish might contain more erroneous documents than the other sets. On the other hand, more than half of the Spanish document set has been categorised to belong to subcategories of E, whereas in the English and French sets only 3 subcategories of E have been used. The similarities of these classes and hence of the documents make it more difficult to build a good model. However, the difference is low and considering also above results with the multi-label classification, the classifier seems to behave robust towards different languages, hence applicable equally good for the English as for other languages.

6.5.3 Integration of domain specific background knowledge

The third and last evaluation tests the effect of integrating the domain specific background knowledge provided by the AGROVOC thesaurus. For this purpose, the converted AGROVOC has been pruned again using the algorithm discussed in chapter 5. This time, $X_{multi\_en\_Cat}$ and $X_{multi\_en\_Desc}$ have been used as domain specific document sets. According to the results of the pruner evaluation in Chapter 5, the parameter settings have been chosen to output the largest possible subset wrt the document set domains[33]. One ontology for each set has been retrieved being significantly smaller in size than the complete AGROVOC. The integration of the respective ontology has then been tested on the 4 document sets $X_{single\_en\_Cat}$, $X_{single\_en\_Desc}$, $X_{single\_fr\_Cat}$ and $X_{single\_es\_Cat}$. Two parameters have been varied: the concept integration depth (meaning the maximum depth up to which super concepts and related concepts of a term are included) and the concept integration mode (add concepts, replace with concepts, concepts only) as explained previously in section 6.3.3.

Avg (Precision)			Training Examples						
Ontology	**Concept Depth**	**Concept Integration Mode**	**5**	**10**	**20**	**30**	**40**	**50**	**Total**
False	**-**	**-**	0,5661	0,6293	0,6748	0,6939	0,7018	0,7102	0,6627
True	**1**	**Add**	0,5663	0,6267	0,6806	0,7006	0,7046	0,7170	0,6660
		Replace	0,5511	0,6087	0,6687	0,6811	0,7024	0,7124	0,6541
		Only	0,5517	0,5991	0,6487	0,6637	0,6646*	0,6873	0,6342
	2	**Add**	0,5478	0,6026	0,6704	0,6963	0,7096*	0,7131	0,6557
		Replace	0,5372	0,5961	0,6556	0,7002	0,7004	0,7010	0,6484
		Only	0,5331	0,5943	0,6369	0,6635	0,6640	0,6791	0,6285

Table 18: Performance of $X_{single\_en\_Desc}$ with ontology background knowledge, averaged precision over 30 runs

[33] Frequency weight: tf; Beat strategy: ONE; Ratio: 1.0; Generic Document Set: Gen.

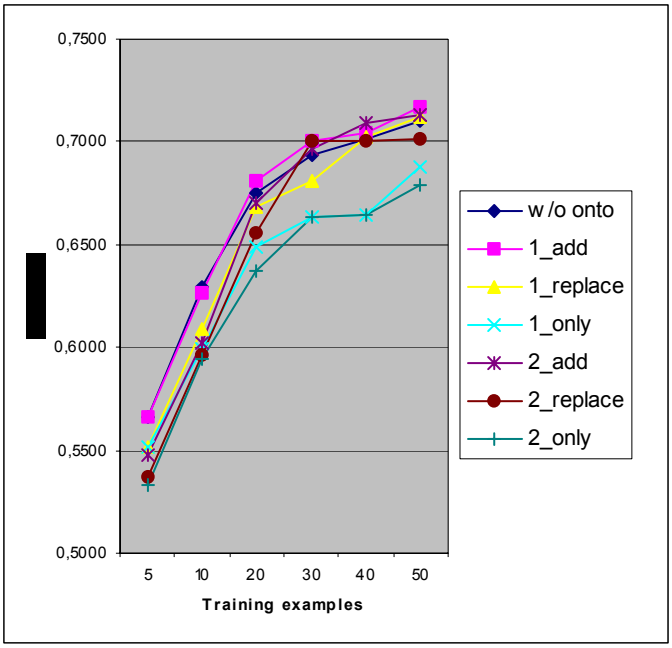


Figure 28: Ontology integration vs. no integration of background knowledge, $X_{single\_en\_Desc}$

Tables 18 – 21 show the resulting precision values averaged over 15 test runs, varying the word pruning threshold between 3 and 10 (except the Spanish test set, where due to limited time only a word prune threshold of 10 could been considered). The results are compared to the performance without ontology integration respectively. Only the table for the set $X_{single\_en\_Desc}$ is presented here for clarity. The other tables are attached in Appendix D for completeness. In all evaluations (except the one on the Spanish set), the extension of the word vector with the concepts (add) shows a slight improvement in performance. The improvements, however, are minimal, as Figure 28 shows exemplary for the test set $X_{single\_en\_Desc}$ and Student's T-Tests again could not show any significance.

The fact that the Spanish set could not support this performance gain could be due to the lower number of evaluation runs per parameter configuration setting. Another effect, which is slightly evident in each test setting, but very clear in the evaluation with the Spanish set is drawn in Figure 29. The deletion of all words, in favour of only building the word vector from the occurring concepts obviously results in significantly worse overall performance.

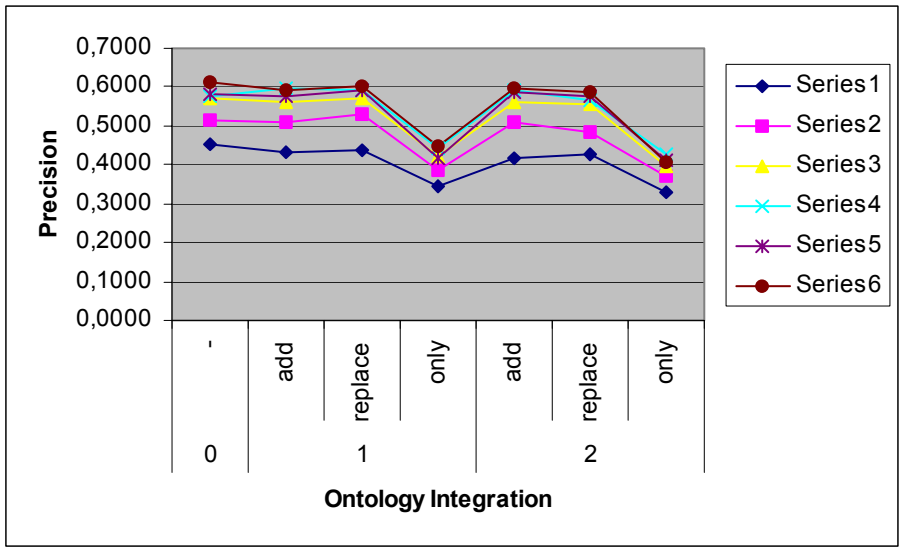


Figure 29: Influence of the different modes of ontology integration on the overall performance (each series corresponds to a specific number of training examples per class, starting at 5)

This effect seems reasonable. In a domain specific ontology like the pruned AGROVOC used here only very specific terms in the word vector of a document appear in the ontology. The information of all the other terms is lost in case of only including the concepts. Hence, less information can be used to build the support vectors.

Regarding the depth of integration, a level of 2 resulted in worse performance in all sets except the French one. Therefore, no clear conclusion can be drawn here from the results. This is equally the case for the integration mode to replace the terms with their respective concepts.

6.6 Related Work

A recently taken similar approach towards multi-label classification using binary classifiers is discussed in [CS02]. Here, the core motivations of the binary Perceptron algorithm are used to create the family of MMP algorithms. The difference to our approach is that these algorithms can be applied in online settings, where the examples are presented one at a time, as opposed to the batch setting used with support vector machines. The algorithms have been applied to the Reuters-21578 test collection, retrieving very good overall performance. The approach seems to be promising, however, should be tested against multilingual data sets like the ones used here in order to be able to make a valid statement in comparing the two results.

A different approach described in [McC99] uses a Bayesian classifier together with a document mixture model to predict multiple classes for each document. A document is represented by all its important words and all words of all documents of the training

document set build the vocabulary. A particular statistical word distribution over the words of the vocabulary is associated with each individual class. Each document is compiled of a weighed mixture of the word distribution of the classes it belongs to. Herewith, this approach takes into consideration all classes at the same time as opposed to splitting the whole problem into a number of binary classifiers.

The commercially available tool Thesaurus Master™ is a professional thesaurus management application, which can be extended by the Machine Aided Indexer™, a rule based automatic document indexer[34]. The approach is extremely interesting in the context of the FAO, since it offers automatic document indexing with terms directly taken from the thesaurus. However, the rule-based approach taken here requires extensive human intervention and programming in order to train the classifier. Indexing is only done based on pre-defined rules, which have to be manually created. This is an error-prone approach resulting particularly in large costs of maintenance. Regarding the generally limited human resources, automatic learning and training as performed in case of support vector machines is economically more reasonable.

6.7 Summary and Outlook

The automatic classifier formerly applied in [Pac02] has been adapted to process and predict multi-label documents. It could be shown that the performance of multi-label classification using support vector machines is comparable with the performance of the single-label case, taking into account that the multi-label case is far more complex. Regarding the important decision on the number of labels to be assigned in the multi-label case, an approach different from varying the score threshold would be interesting to explore. A classifier, predicting the number of labels could be built, with the respective numbers constituting the set of classes.

The classifier has moreover shown to behave robust towards document representation in different languages. This has been tested here for the languages English, French and Spanish. In future research, this result should, however, be supported by tests with other western languages. Additionally, it would be extremely interesting to evaluate the classifier with languages like Arabic and Chinese.

[34] Refer to the web site of the company Data Harmony at [http://www.dataharmony.com] for more detailed information.

Through the integration of background knowledge only insignificant performance gains in case of adding concepts to the word vector could be achieved. The fact that a higher level of integration resulted in worse performance in the majority of cases leads to the conclusion, that the integration of too many concepts enhances the word vector with too many features. In fact, if too many concepts from the ontology are added to the word vectors, there is a danger that these word vectors cannot be separated anymore. This is evident especially in case of adding arbitrarily related concepts as done in the current implementation. Integrating non-hierarchic relations might blur the distinctions between word vectors of different classes. Leaving out this option, restricting the concept inclusion only to super concepts should therefore be evaluated in future tests.

In this evaluation, the AGROVOC has been pruned to a smaller size due to performance reasons. A smaller ontology resulted in less time needed for each evaluation run. Even though, the largest relevant ontology subset according to the findings of Chapter 5 has been used here, relevant concepts might have been pruned. In future research, the integration of the complete AGROVOC should be reapplied with the respective document sets in order to validate the assumption made here, that the pruned version contains enough concepts for word vector extension. This additional test moreover serves as an indirect pruner evaluation.

A slightly different way of using the integration of multilingual ontologies has not been explored in this research work: The classifier can be trained with documents in one language, using the option to build the word vector only from the concepts. Testing with the equivalent document set in another language should then perform equally well than testing with documents of the language the classifier has been trained on. The re-implementation of the ontology integration module to allow for language independent representation of a concept in the word vector thus provides a promising opportunity.

Moreover, as already indicated, the application of different, adapted performance measures especially for measuring the multi-label case might lead to different interpretation of the results. In addition, subject expert assessment of automatic predictions, similarly to the empirical evaluation of the ontology pruner discussed in Chapter 5, would give a more complete base to measure against, than simply using the pre-classifications.

7 Conclusion

7.1 Summary

The data and information explosion on the World Wide Web has left us with the complex task to organise and structure this vast amount of resources to provide and facilitate access to it. Especially, the integration of similar domains in a multilingual environment and the aggregated retrieval of knowledge and information from these domains impose a challenging scenario. Uniquely identifying resources on the World Wide Web and exposing metadata about them in an unambiguous and machine-processable format is one of the solutions the Semantic Web initiative proposes in order to achieve this goal. Ontologies as part of the Semantic Web provide the necessary specifications and definitions needed to unambiguously define concepts of a domain and structure its data and knowledge. Ontologies thus need to be created in different domains and in large numbers. The engineering process is a time consuming task carried out mainly by human beings; hence needs extensive automatic support in the future.

In this thesis, I proposed a comprehensive, reusable framework for the semi-automatic creation of domain ontologies. The framework has been applied in a prototype project in the agricultural domain of the Food and Agriculture Organisation of the United Nations (FAO of the UN) to create an ontology on Food Safety, Animal and Plant Health. The framework consists of an iterative cycle of 5 phases mainly focussing on automatic tool support for collaborative, multilingual ontology editing, knowledge extraction from natural language texts and the reuse and automatic exploitation of already existing ontologies or other structured vocabularies. In this prototype, the agricultural thesaurus AGROVOC, which has been developed in the FAO and contains over 26,000 terms, has been reused. An automatic ontology pruning algorithm - extracting domain-specific concepts from an already existing larger ontology as part of the framework - has been adapted and evaluated in one central part of this work. The automatic pruner could provide significant support in reusing and incorporating the external AGROVOC into the ontology to be built. He pruned the initial AGROVOC to only 25% of its original size, therefore saving valuable time of human assessment. 75% of the extracted concepts have been identified to be important for the target domain in an empirical evaluation. I showed, that several parameter variations of the algorithm could increase this value of 75%, hence decreasing the time of human assessment. However, this could only be achieved losing other valuable concepts at the same time.

The framework has been applied iteratively in this project. In the first cycle, a prototype ontology consisting of 106 concepts has been created. Currently, the framework is in its second iteration in the phase of merging the pruned AGROVOC ontology into the created and extended core ontology from the first iteration.

The ontology is subject to be incorporated into the International Portal on Food Safety, Animal and Plant Health to help structure information and subject index large amounts of documents and other resources input into the system in 3 different languages from various locations all over the world. Subject indexing is part of the metadata creation process and again is an extremely time consuming process. Automatic support is needed to populate the portal in a controlled way. In a first step, an ontology web browser as part of the evaluation phase of the ontology engineering framework has been extended to support a human indexer in browsing and choosing concepts from an ontology to index a document. At the current stage, no further evaluation could be performed, since the portal as well as the ontology is still under construction and the application has not yet been incorporated.

In the second central part of this thesis, an automatic text classifier based on support vector machines has been evaluated with an extensive test set compiled from documents of the FAO in three different languages. The classifier has been extended to classify on multiple labels and integrate background knowledge in form of ontologies as being created with the here presented framework. On the test set compiled for this evaluation, I have shown, that the classifier performs almost equally well across different languages and is therefore a promising approach in a multilingual environment. Multi-label classification performed surprisingly well, compared with the single-label case, taking into consideration, that the assignment of a document to multiple categories is more complex than the deterministic assignment to only one. The integration of background knowledge did not bring the expected performance gains in the current application. However, I reasoned its potential using different settings especially in the multilingual environment to be explored in further research.

The lack of sufficient training examples for each possible index term makes it difficult to incorporate this classifier into a subject indexing application based on the whole AGROVOC or other ontologies of this size used for document indexing. However, given a smaller amount of categories to choose from (as in case of the 115 AGROVOC subject categories), the classifier evaluated here seems to be useful to semi-automatically support the task of a human indexer.

7.2 Outlook

The presented framework and the research work in this thesis has been carried out in an ongoing project, thus demonstrating its applicability. In future work, the application of the framework to other domains has to be evaluated and compared with the results of this first project. The ontology pruner needs to be re-evaluated in other domains in order to be able to hold or reject the here made statements and conclusions. At further development stages of the project, the framework shall be re-applied in a third iteration, this time exploiting the CAB[35] thesaurus for reuse. Further work needs to be done on specifying and formalising the processes within the phases of the framework. The usefulness of the extended ontology browser and especially the integration and use of the ontology in the portal needs to be evaluated in a next step. A possible use case scenario is to enhance search results, by providing the user with ontology guided search specifications or by retrieving similar items using the power given by the semantic relationships of the ontology. Extensive research still needs to be carried out in this field, especially regarding display and browsing functionality in order not to overload the user with the information given in an ontology.

Regarding document indexing, the automatic text classifier should be incorporated into a real application, exposing it for user testing in real environments. A usage scenario within this project is an automatic classifier operating on the main categories, which will be developed to structure the web site of the portal.

Overall, it can be concluded that the work of this thesis started off an ongoing project using an extendable approach, which can be reapplied to other domains. I gave hints and interesting starting points for a variety of promising future research work building on the here presented results.

Acknowledgements

I would like to express my gratitude to Raphael Volz and Andreas Hotho from the AIFB for their sound direction, technical guidance and supervision throughout the project. I particularly thank Boris Motik for his extensive technical support on running and configuring the KAON environment. I also gratefully recognise the help given by all the colleagues at the FAO, who substantially contributed in requirements analysis, tool testing and the compilation of the test document sets, which have been created within this project.

[35] The CABI thesaurus is another thesaurus in the agricultural domain differing from the AGROVOC in several parts.

References

[AE99] Aas, K.; Eikvil, L.: Text Categorization: a survey. Technical Report #941, Norwegian Computing Center. 1999.

[AF99] Amann, B.; Fundulaki, I.: Integrating Ontologies and Thesauri to Build RDF Schemas. In ECDL-99: Research and Advanced Technologies for Digital Libraries, Lecture Notes in Computer Science, pages 234-253. Springer-Verlag. Paris, France, September 1999.

[AOS01] Agricultural Ontology Service (AOS) "A tool for Facilitating Access to Knowledge" - Concept Note Version 5.5. Internal Document, UN FAO. [Online: http://www.fao.org/agris/aos/Documents/AOS_Draftproposal.htm]. August 2001.

[BA97] Breslow, L.; Aha, D.W.: Comparing Tree-Simplification Procedures. In Proceedings of the Sixth International Workshop on Artificial Intelligence and Statistics, pp. 67-74. Ft. Lauderdale, FL. 1997.

[BEH+02] Bozsak, E.; Ehrig, M.; Handschuh, S.; Hotho, A.; Mädche, A.; Motik, B.; Oberle, D.; Schmitz, C.; Staab, S.; Stojanovic, L.; Stojanovic, N.; Studer, R.; Stumme, G.; Sure, Y.; Tane, J.; Volz, R.; Zacharias, V.: KAON – Towards a large scale Semantic Web. In: Proceedings of EC-Web 2002. Aix-en-Provence, France, September 2-6, 2002. LNCS, Springer, 2002.

[Ber96] Berners-Lee, T.: The World Wide Web: Past, Present and Future. In IEEE Computer special issue, v.29 n.10, p.69-77. October 1996.

[BFIM98] Berners-Lee, T.; Fielding, R.; Irvine, U.C.; Masinter, L.: Uniform Resource Identifiers (URI): Generic Syntax. IETF Request for Comments: 2396. [Online: http://www.ietf.org/rfc/rfc2396.txt]. August 1998.

[BG02] Brickley, D.; Guha, R.V.: RDF Vocabulary Description Language 1.0: RDF Schema. W3C Working Draft. [Online: http://www.w3.org/TR/rdf-schema/]. 12 November 2002.

[BHL01] Berners-Lee, T.; Hendler, J.; Lassila, O.: The Semantic Web. In Scientific American, 284(5), 34-43. May 2001.

[BHL99] Bray, T; Hollander, D.; Layman, A.: Namespaces in XML. W3C Recommendation. [Online: http://www.w3.org/TR/1999/REC-xml-names-19990114/]. 14 January 1999.

[BP02] Beck, H.; Pinto, H.S.: Overview of Approach, Methodologies, Standards, and Tools for Ontologies. Internal Report, UN FAO. [Online: http://www.fao.org/agris/aos/Documents/BackgroundAOS.html]. December 2002.

[BPSM00] Bray, T.; Paoli, J.; Sperberg-McQueen, C.M.; Maler, E.: Extensible Markup Language (XML) 1.0 (Second Edition). W3C Recommendation. [Online: http://www.w3.org/TR/REC-xml]. 6 October 2000.

[Bri93] Brill, D.: Loom Reference Manual for Loom version 2.0. [Online: http://www.isi.edu/isd/LOOM/LOOM-HOME.html]. December 1993.

[CG00] Corcho, O.; Gómez-Pérez, A.: A Roadmap to Ontology Specification Languages. In Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management EKAW'00, October, 2000.

[CHH+01] Conolly, D.; Harmelen, F. van; Horrocks, I.; McGuinness, D.; Patel-Schneider, P.F.; Stein, L.A.: Annotated DAML+OIL Ontology Markup. W3C Note. [Online: http://www.w3.org/TR/daml+oil-walkthru/]. 18 December, 2001.

[Cle84] Cleverdon, C.: Optimizing convenient on-line access to bibliographic databases. In Information Sciences and Use, 4(1):37-47. 1984.

[CV95] Cortes, C.; Vapnik, V.: Support-Vector Networks. In Machine Learning, 20(3):273-297, September 1995

[CS02] Crammer, K.; Singer, Y.: A new family of online algorithms for category ranking. In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 151-158. Tampere, Finland. 2002.

[EBS+02 Everett, J. O.; Bobrow, D.G.; Stolle, R.; Crouch, R.; Paiva, V. de; Condoravdi, C.; Berg, M. van den; Polanyi, L.: Making Ontologies Work for resolving redundancies across documents, Communications of the ACM, 45(2):55-60. ACM Press. February 2002.

[Ehr02] Ehrig, M.: Ontology-Focused Crawling of Documents and Relational Metadata. Internal Document: Master's Thesis, AIFB/FZI, University of Karlsruhe. 2002.

[FBGG98] Fernandez, M.; Blazquez, M.; Garcia-Pinar, J.M.; Gomez-Perez, A.: Building Ontologies at the Knowledge Level using the Ontology Design Environment. In Proceedings of KAW'98, the 11th Knowledge Acquisition Workshop, 18-23. Banff, Alberta, Canada. April 1998.

[Fen01] Fensel, D.: Ontologies: a silver bullet for knowledge management and electronic commerce. Springer-Verlag New York, Inc., New York, NY. 2001.

[FFR97] Fikes, R.; Farquhar, A.; Rice, J.: Tools for Assembling Modular Ontologies in Ontolingua. Knowledge Systems Laboratory. 1997.

[FGPP99] Fernandez, M.; Gomez-Perez, A.; Pazos Sierra, A.; Pazos Sierra, J.: Building a Chemical Ontology Using METHONTOLOGY and the Ontology Design Environment. In IEEE Expert (Intelligent Systems and Their Applications), 14(1):37-46. 1999.

[FHK+00] Fensel, D.; Harmelen, F.; Klein, M.; Akkermans, H.; Broekstra, J.; Fluit C.; Meer, J. van der; Schnurr, H.; Studer, R.; Hughes, J.; Krohn, U.; Davies, J.; Engels, R.; Bremdal, B.; Ygge, F.; Lau, T.; Novotny, B.; Reimer, U.; Horrocks, I.: Ontoknowledge: Ontology-based Tools for Knowledge Management, In Proceedings of the eBusiness and eWork 2000 (EMMSEC 2000) Conference, Madrid. October 2000.

[Gan02] Gangemi, A.: Development of an Integrated Formal Ontology and an Ontology Service for Semantic Interoperability in the Fishery Domain. Internal Document, CNR – Institute of Cognitive Sciences and Technologies, Ontology and Conceptual Modelling Group. [Online: http://www.fao.org/agris/aos/Documents/FisheryConceptPaper.doc]. January 2002.

[GGM+02] Gangemi, A.; Guarino, N.; Masolo, C.; Oltramari, A.; Schneider, L.: Sweetening Ontologies with Dolce. To appear at EKAW2002. 2002.

[Gru93] Gruber, T.R.: Toward Principles for the Design of Ontologies Used for Knowledge Sharing. Formal Ontology in Conceptual Analysis and Knowledge Representation. Kluwer Academic Publishers. 1993.

[GSG96] Gangemi, A.; Steve, G.; Giacomelli, F.: ONIONS: An Ontological Methodology for Taxonomic Knowledge Integration. In ECAI-96 Workshop on Ontological Engineering, Budapest. August 1996.

[Gua98] Guarino N.: Formal Ontology and Information Systems. In Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, pages 3-15. Trento, Italy. IOS Press. June 1998.

[GW00] Guarino, N.; Welty, C.: A formal ontology of properties. In Proceedings of 12th Int. Conf. on Knowledge Engineering and Knowledge Management,

Lecture Notes on Computer Science. Springer Verlag. 2000.

[Joa01] Joachims, T.: A Statistical Learning Model of Text Classification for Support Vector Machines. In Proceedings of SIGIR-01, 24th ACM International Conference on Research and Development in Information Retrieval, pp.128-136. ACM Press, New York, US. 2001.

[Joa98] Joachims, T.: Text categorization with Support Vector Machines: Learning with many relevant features. In Proceedings of ECML-98, 10th European Conference on Machine Learning, 1398, 137-142. Springer Verlag, Heidelberg, DE. 1998.

[Jom72] Jomes, K. S.: A statistical interpretation of terms specificity and its application retrieval. In Journal of Documentation, 28, 11-20. 1972.

[Kim02] Kim, H.: Predicting how ontologies for the semantic web will evolve. In Communications of the ACM, 45(2):48-54. ACM Press New York, NY, USA. February 2002.

[KLW90] Kifer, M.; Lausen, G.; Wu, J: Logical Foundations of Object-Oriented and Frame-Based Languages. Technical Report TR-90-003. 1990

[KVM00] Kietz, J.-U.; Volz, R.; Maedche, A.: Extracting a Domain-Specific Ontology from a Corporate Intranet. In Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic Workshop, Lisbon, 2000, pp. 167-175. Association for Computational Linguistics. September 2000.

[Lew99] Lewis, D.D.: Reuters-21578 text categorization test collection distribution 1.0. [Online: http://www.research.att.com/#lewis]. 1999.

[LS99] Lassila, O.; Swick, R.R.: Resource Description Framework (RDF) Model and Syntax Specification. W3C Recommendation. [Online: http://www.w3.org/TR/REC-rdf-syntax/]. 22 February 1999.

[Luh58] Luhn, H.P.: The automatic creation a literature abstracts. In IBM Journal of Research and Development, 2(2):159-165. April 1958.

[McC99] McCallum, A.: Multi-label text classification with a mixture model trained by em. AAAI'99 Workshop on Text Learning. 1999.

[MI96] Mizoguchi, R.; Ikeda, M.: Towards Ontology Engineering. Technical Report AI-TR-96-1, I.S.I.R., The Institute of Scientific and Industrial Research, Osaka University, 567 Japan. 1996.

[MMV02] Motik, B; Maedche, A.; Volz, R.: A conceptual modelling approach for building semantics-driven enterprise applications. In Proceedings of the First International Conference on Ontologies, Databases and Application of Semantics (ODBASE-2002). Springer, LNAI, California, USA. 2002 .

[MNS02] Maedche, A.; Neumann, G.; Staab, S.: Bootstrapping an Ontology-Based Information Extraction System. In Studies in Fuzziness and Soft Computing, editor J. Kacprzyk. Intelligent exploitation of the web. Springer. 2002.

[MO97] Martin, J.; Odell, J.J.: Object-oriented methods (UML ed., 2nd ed.): a foundation. Prentice-Hall Inc., Upper Saddle River, NJ, USA. 1997.

[MRA95] Mehta, M.; Rissanen, J.; Agrawal, R.: MDL-based decision tree pruning. In Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD), pp. 216- 221. 1995.

[MS00] Maedche, A.; Staab, S.: Mining Ontologies from Text. In Proceedings of 12th International Workshop on Knowledge Engineering and Knowledge Management (EKAW'2000), Juan-les-Pins, French Riviera. Springer Lecture Notes in Artificial Intelligence (LNAI-1937). 2000.

[NN95] Nance, D.W.; Naps, T.L.: Introduction to Computer Science: Programming, Problem Solving and Data Structures, 3rd Edition. West Publishing Company, 1172-1175. 1995.

[Pac02] Pache, G.: Textklassifikation mit Support-Vektor-Maschinen unter Zuhilfenahme von Hintergrundwissen. Internal Document: Studienarbeit at University of Karlsruhe, Karlsruhe, Germany. April 2002.

[Pal01] Palmer, S.: The Semantic Web: An Introduction. [Online: http://infomesh.net/2001/swintro, 2001].

[PGM99] Pinto, H.S.; Gomez-Perez, A.; Martins, J.P.: Some issues on ontology integration. In Proceedings of the IJCAI'99 Workshop on Ontology and Problem-Solving Methods: Lesson learned and Future Trends, 18, pp. 7.1-7.11, Amsterdam. CEUR Publications. 1999.

[PM01] Pepper, S.; Moore, G.: XML Topic Maps (XTM) 1.0. TopicMaps.Org Specification. [Online: http://www.topicmaps.org/xtm/index.html]. August, 2001.

[RC00] Ribière, M.; Charlton, P.: Ontology Overview from Motorola Labs with a comparison of ontology languages. [Online: http://www.fipa.org/docs/input/f-

in-00045/f-in-00045.pdf]. December, 2000.

[Rol01] Rolf, D.: Seminar Wissensmanagment - Ontologien. [Online: http://www.informatik.hu-berlin.de/~rolf/kms/vortrag.html, 2001].

[RS01] Ruiz, M. E.; Srinivasan, P.: Combining Machine Learning and Hierarchical Structures for text categorization. In Proceedings of the 10th ASIS SIG/CR Classification Research Workshop, Advances in Classification Research–Vol. 10. November 1999.

[RSW02] Rose, T.G.; Stevenson, M.; Whitehead, M.: The Reuters Corpus Volume 1 - from Yesterday's News to Tomorrow's Language Resources. In Proceedings of the Third International Conference on Language Resources and Evaluation, Las Palmas de Gran Canaria. 29-31 May 2002.

[Sal88] Salton, G.: Automatic Text Processing: The transformation, Analysis and Retrieval of Information by Computer. Addison-Wesley, 1988.

[SBD+99] Staab, S; Braun, C. Düsterhöft; Heuer, A.; Klettke, M.; Melzig, S.; Neumann, G.; Prager, B. Pretzel, J.; Schnurr, H.-P.; Studer, R.; Uszkoreit, H.; Wrenger, B.: GETESS – searching the web exploiting German texts. In CIA'99 - Proceedings of the 3rd Workshop on Co-operative Information Agents. Upsala, Sweden, July 31-August 2, 1999, LNCS 1652, pages 113--124, Berlin, Springer. 1999.

[SBF98] Studer, R.; Benjamins, V.R.; Fensel, D.: Knowledge Engineering: Principles and Methods. In Data & Knowledge Engineering, 25(1-2), 161-197. 1998.

[Seb99] Sebastiani, F.: Machine learning in automated text categorization. Tech. Rep. IEI-B4-31-1999, Consiglio Nazionale delle Ricerche, Pisa, Italy. 1999.

[SEMD00] Staab, S.; Erdmann, M.; Mädche, A.; Decker, S.: An Extensible Approach for Modelling Ontologies in RDF(S). In Proceedings of ECDL 2000 Workshop on the Semantic Web, 11-22. 2000.

[SHH02] Schneider, P.; Horrocks, I.; Harmelen, F.: OWL Web Ontology Language 1.0 Abstract Syntax. W3C Working Draft. [Online: http://www.w3.org/TR/2002/WD-owl-absyn-20020729/].July, 2002.

[SSA02] Sure, Y.; Staab, S.; Angele, J.: OntoEdit: Guiding Ontology Development by Methodology and Inferencing. In Proceedings of the International Conference on Ontologies, Databases and Applications of Semantics ODBASE 2002. University of California, Irvine, USA, Springer, LNCS. October 28 -

November 1, 2002.

[TI94] Tokunaga, T.; Iwayama, M.: Text Categorization based on Weighted Inverse Document Frequency. Technical Report 94 TR0001, Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan. 1994.

[Tro02] Trochim, W.M.K.: The T-Test. [Online: http://trochim.human.cornell.edu/kb/stat_t.htm, Dec 2002].

[UG96] Uschold, M.; Gruninger, M.: Ontologies: Principles, methods and applications. In Knowledge Engineering Review, 11(2), 93-155. 1996.

[UK95] Uschold, M.; King, M.: Towards a methodology for building ontologies. In Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95. Montreal, Canada. 1995.

[Usc96] Uschold, M.: Building Ontologies: Towards Unified Methodology, In Proceedings of experts systems 96, the 16 th Annual Conference of the British Computer Society Specialist Group on Expert Systems, held in Cambridge, UK. December 1996.

[Volz00] Volz, R.: Akquisition von Ontologien mit Text-Mining-Verfahren. Technical Report 27, Rentenanstalt/Swiss Life, CC/ITRD, CH-8022 Zürich, Switzerland, ISSN 1424-4691. 2000.

[WF94] Welty, C. A.; Ferrucci, D. A.: What's in an instance?. RPI Computer Science Technical report 94-18. 1994.

[WF99] Witten, I.; Frank, E.: Data Mining, Practical Machine Learning Tools and techniques with Java implementations. Morgan Kaufmann. 1999.

[WG01] Welty, C.; Guarino, N.: Supporting Ontological Analysis of Taxonomic Relationships. In Data and Knowledge Engineering 39(1), pp. 51-74. 2001.

[WSWS01] Wielinga, B.J.; Schreiber, A.Th.; Wielemaker, J.; Sandberg, J. A. C.: From thesaurus to ontology. Report ICES-MIA project, University of Amsterdam, Social Science Informatics. [Online: http://www.swi.psy.uva.nl/usr/Schreiber/papers/Wielinga01a.pdf]. 2001.

[WW99] Weston, J.; Watkins, C.: Support-Vector-Machines for Multi-Class Pattern Recognition. In Proceedings of the Seventh European Symposium On Artificial Neural Networks. April 1999.

[Yan99] Yang, Y.: An Evaluation of Statistical Approaches to Text Categorization. Journal of Information Retrieval, 1(1/2) pp. 67-88. 1999.

A KAON RDFS representation of the Ontology on Food Safety, Animal and Plant Health (extract)

```
<!DOCTYPE rdf:RDF [
    <!ENTITY kaon 'http://kaon.semanticweb.org/2001/11/kaon-lexical#'>
    <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
    <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
    <!ENTITY a 'http://www.fao.org/OFsAPH#'>
]>

<?include-rdf     logicalURI="http://kaon.semanticweb.org/2001/11/
                  kaon-root"
                  physicalURI="jar:file:/C:/CVS/build/kaon_build_root/
                  kaon/release/lib/kaonapi.jar!/edu/unika/aifb/kaon/api/
                  res/kaon-root.xml"?>
<?include-rdf     logicalURI="http://kaon.semanticweb.org/2001/11/
                  kaon-lexical"
                  physicalURI="jar:file:/C:/CVS/build/kaon_build_root/
                  kaon/release/lib/kaonapi.jar!/edu/unika/aifb/kaon/api/
                  res/kaon-lexical.xml"?>

<rdf:RDF xml:base="http://www.fao.org/OFsAPH"
    xmlns:kaon="&kaon;"
    xmlns:rdf="&rdf;"
    xmlns:rdfs="&rdfs;"
    xmlns:a="&a;">

<kaon:Label rdf:ID="1034783441478-1832388080"
    kaon:value="international trade">
    <kaon:references rdf:resource="#international-trade"/>
    <kaon:inLanguage rdf:resource="&kaon;en"/>
</kaon:Label>
<kaon:Label rdf:ID="urn:rdf:6705a3de868b3ccb79e41ef3c4d5d255-815"
    kaon:value="international food trade">
    <kaon:references rdf:resource="#international_food_trade"/>
    <kaon:inLanguage rdf:resource="&kaon;en"/>
</kaon:Label>
<kaon:Label rdf:ID="urn:rdf:6705a3de868b3ccb79e41ef3c4d5d255-763"
    kaon:value="commodities">
    <kaon:references rdf:resource="#commodities"/>
    <kaon:inLanguage rdf:resource="&kaon;en"/>
</kaon:Label>
<kaon:Label rdf:ID="1034673406391-1754206929"
    kaon:value="involve">
    <kaon:references rdf:resource="#involve"/>
    <kaon:inLanguage rdf:resource="&kaon;en"/>
</kaon:Label>

<rdfs:Class rdf:ID="commodities">
    <rdfs:subClassOf rdf:resource="&kaon;Root"/>
</rdfs:Class>
<rdfs:Class rdf:ID="international-trade">
    <rdfs:subClassOf rdf:resource="&kaon;Root"/>
</rdfs:Class>
<rdfs:Class rdf:ID="international_food_trade">
    <rdfs:subClassOf rdf:resource="#international-trade"/>
    <a:involve rdf:resource="#commodities"/>
</rdfs:Class>

<rdf:Property rdf:ID="involve"
    kaon:symmetric="true">
    <rdfs:range rdf:resource="&kaon;Root"/>
    <rdfs:domain rdf:resource="&kaon;Root"/>
</rdf:Property>
</rdf:RDF>
```

B Complete list of web sites output by the Focused Crawler

	URL	score
1	http://www.foodsafety.gov/~fsg/cstrpl-4.html	3122
2	http://www.uark.edu/depts/fsc/othersites.html	2838
3	http://www.dfst.csiro.au/fshlist.htm	2675
4	http://www.foodsafetynetwork.ca/food.htm	2592
5	http://www.inspection.gc.ca/english/ppc/biotech/safsal/fooalie.shtml	2199
6	http://www.foodsafety.gov/~fsg/fssyst4.html	2150
7	http://www.inspection.gc.ca/english/index/fssae.shtml	2046
8	http://vm.cfsan.fda.gov/~lrd/foodteam.html	2046
9	http://food4kids.missouri.edu/QA/searchform.html	1999
10	http://www.foodsafety.gov/~fsg/fssyst2.html	1921
11	http://www.uark.edu/depts/fsc/index.html	1884
12	http://www.cfsan.fda.gov/~dms/nutrlist.html	1789
13	http://ific.org/food/	1676
14	http://foodnet.fic.ca/page4.html	1666
15	http://www.inspection.gc.ca/english/corpaffr/foodfacts/fftoce.shtml	1638
16	http://www.fao.org/es/ESN/biotech/safety.htm	1586
17	http://www.ers.usda.gov/briefing/ConsumerFoodSafety/index.htm	1564
18	http://www.ers.usda.gov/briefing/ConsumerFoodSafety/	1564
19	http://www.ers.usda.gov/Briefing/consumerfoodsafety/index.htm	1564
20	http://www.ers.usda.gov/Briefing/ConsumerFoodSafety/index.htm	1564
21	http://www.foodsafety.gov/~fsg/presidentscouncil.html	1551
22	http://www.foodsafety.gov/~fsg/PresidentsCouncil.html	1551
23	http://www.foodsafety.gov/~fsg/fsggov.html	1539
24	http://www.cfsan.fda.gov/~dms/fs-toc.html	1536
25	http://www.foodsafety.gov/%7Edms/fs-toc.html	1536
26	http://www.extension.iastate.edu/foodsafety/fspubs.html	1407
27	http://www.extension.iastate.edu/foodsafety/haccp.html	1392
28	http://www.inspection.gc.ca/english/ppc/biotech/conse.shtml	1387
29	http://www.inspection.gc.ca/english/bureau/bureaue.shtml	1382
30	http://www.inspection.gc.ca/english/corpaffr/foodfacts/powere.shtml	1356
31	http://www.inspection.gc.ca/english/bureau/retdet/retdete.shtml	1349
32	http://www.inspection.gc.ca/english/bureau/invenq/comme.shtml	1335
33	http://www.inspection.gc.ca/english/corpaffr/foodfacts/picnice.shtml	1291
34	http://www.nal.usda.gov/fnic/foodborne/fbindex/index.htm	1278
35	http://www.nal.usda.gov/foodborne/fbindex/index.htm	1278
36	http://www.extension.iastate.edu/foodsafety/fdlaw.html	1274
37	http://www.extension.iastate.edu/foodsafety/resources.html	1243
38	http://www.ers.usda.gov/Emphases/SafeFood/	1233
39	http://www.fao.org/es/ESN/biotech/tabconts.htm	1205
40	http://www.inspection.gc.ca/english/corpaffr/foodfacts/campinge.shtml	1199
41	http://www.cfsan.fda.gov/~dms/opa-guid.html	1183
42	http://www.inspection.gc.ca/english/corpaffr/foodfacts/storagee.shtml	1179
43	http://www.fao.org/es/ESN/riskcomm/HTTOC.htm	1158
44	http://www.foodsafetynetwork.ca/onfarm.htm	1155
45	http://vm.cfsan.fda.gov/~lrd/cfsan2.html	1152
46	http://www.inspection.gc.ca/english/corpaffr/foodfacts/floodse.shtml	1139
47	http://ccr.ucdavis.edu/irr/what3.shtml	1104
48	http://www.health.state.ut.us/els/envsvc/foodsafety/links.html	1082
49	http://www.extension.iastate.edu/foodsafety/rad/irwhat.html	1081
50	http://www.fmi.org/foodsafety/supermarket_news/	1052
51	http://www.foodsafety.gov/%7Edms/fs-impor.html	1050
52	http://www.inspection.gc.ca/english/corpaffr/foodfacts/barbece.shtml	1045
53	http://www.extension.iastate.edu/foodsafety/tailgate.html	1036

54	http://www.extension.iastate.edu/foodsafety/resources1.html	1036
55	http://www.foodsafetynetwork.ca/trade/trade.htm	1017
56	http://www.extension.iastate.edu/foodsafety/entertain.html	1005
57	http://www.foodsafety.gov/~fsg/sr2.html	1004
58	http://www.ers.usda.gov/briefing/IndustryFoodSafety/index.htm	988
59	http://www.ers.usda.gov/briefing/IndustryFoodSafety/	988
60	http://www.ers.usda.gov/Briefing/IndustryFoodSafety/index.htm	988
61	http://www.inspection.gc.ca/english/corpaffr/foodfacts/kitchene.shtml	977
62	http://www.ers.usda.gov/briefing/FoodSafetyPolicy/	973
63	http://www.ers.usda.gov/briefing/FoodSafetyPolicy/index.htm	973
64	http://www.ers.usda.gov/Briefing/FoodSafetyPolicy/index.htm	973
65	http://www.ers.usda.gov/briefing/foodsafetypolicy/	973
66	http://www.foodsafety.gov/~fsg/fsggrant.html	960
67	http://www.acsh.org/publications/priorities/0903/foodsafety.html	955
68	http://www.extension.iastate.edu/foodsafety/rad/other_irrad.html	924
69	http://www.worldfoodscience.org	924
70	http://www.inspection.gc.ca/english/corpaffr/foodfacts/leftove.shtml	917
71	http://www.inspection.gc.ca/english/corpaffr/foodfacts/microe.shtml	903
72	http://www.agr.state.nc.us/cyber/kidswrld/foodsafe/	899
73	http://www.agr.state.nc.us/cyber/kidswrld/foodsafe/index.htm	899
74	http://www.cfsan.fda.gov/~dms/opa-notf.html	893
75	http://www.cfsan.fda.gov/~dms/fsirp006.html	878
76	http://www.foodsafety.gov/~fsg/fsgoth.html	866
77	http://www.cfsan.fda.gov	862
78	http://www.cfsan.fda.gov/list.html	862
79	http://www9.myflorida.com/environment/facility/food/default.html	862
80	http://www.inspection.gc.ca/english/ppc/psps/haccp/haccpe.shtml	856
81	http://www.mda.state.mi.us/food/mufl/FoodDigest/FoodDigest.htm	833
82	http://www.inspection.gc.ca/english/corpaffr/foodfacts/holidaye.shtml	831
83	http://www.cfsan.fda.gov/~lrd/foodadd.html	818
84	http://www.extension.iastate.edu/foodsafety/seasons.html	818
85	http://www.fmi.org/consumer/foodkeeper/search.htm	816
86	http://www.cfsan.fda.gov/~dms/wh-alrgy.html	811
87	http://www.dfst.csiro.au/whatson/Whats.htm	796
88	http://www.inspection.gc.ca/english/lab/fooalie.shtml	787
89	http://www.ifst.org/hottop19.htm	775
90	http://nafs.tamu.edu	772
91	http://www.nal.usda.gov/fnic/foodborne/fbindex/038.htm	766
92	http://www.nal.usda.gov/foodborne/fbindex/038.htm	766
93	http://www.foodsafety.iastate.edu	762
94	http://www.foodsafety.iastate.edu/homepage.html	762
95	http://www.law.cornell.edu/topics/food_drugs.html	758
96	http://www.state.oh.us/agr/FoodSafetyDiv.html	756
97	http://www.uark.edu/depts/fsc/newslinks.html	740
98	http://www.extension.iastate.edu/foodsafety/rad/irhow.html	738
99	http://www.foodsafety.gov/~fsg/additive.html	735
100	http://www.extension.iastate.edu/foodsafety/take_out.html	731
101	http://www.foodsafety.gov/~fsg/cwelcome.html	714
102	http://ccr.ucdavis.edu/irr/what1.shtml	710
103	http://www.mda.state.mi.us/food/mufl/Training.htm	702
104	http://www.cfsan.fda.gov/~dms/opa-cg4.html	697
105	http://www.nal.usda.gov/fnic/foodborne/fbindex/035.htm	696
106	http://www.nal.usda.gov/foodborne/fbindex/035.htm	696
107	http://www.extension.iastate.edu/foodsafety/rad/irradhome.html	686
108	http://www.fda.gov/fdac/features/2001/401_food.html	679
109	http://www.inspection.gc.ca/english/corpaffr/foodfacts/turkeye.shtml	675
110	http://www9.myflorida.com/environment/facility/food/index.html	673
111	http://www.inspection.gc.ca/english/corpaffr/foodfacts/eggtipse.shtml	661

112	http://health.utah.gov/els/envsvc/foodsafety/	658
113	http://www.inspection.gc.ca/english/corpaffr/foodfacts/hallowe.shtml	655
114	http://www.extension.iastate.edu/foodsafety/ccp/point5.html	653
115	http://www.dfst.csiro.au	651
116	http://www.foodsafety.gov/~fsg/september.html	639
117	http://www.fmi.org/foodsafety/	632
118	http://www.hhs.gov/news/speech/2001/011127.html	625
119	http://www.cfsan.fda.gov/~dms/opa-help.html	624
120	http://www.fsis.usda.gov/Orlando2002/index.htm	622
121	http://www.cdc.gov/ncidod/EID/vol3no4/osterhol.htm	621
122	http://www.foodsafety.iastate.edu/abstract.html	620
123	http://www.cfsan.fda.gov/~dms/fsiupd11.html	612
124	http://www.state.ak.us/dec/deh/sanitat/homesan.htm	599
125	http://www.nal.usda.gov/fnic/foodborne/fbindex/030.htm	599
126	http://www.nal.usda.gov/foodborne/fbindex/030.htm	599
127	http://www.acsh.org/publications/booklets/biotechnology2000.html	598
128	http://www.extension.iastate.edu/foodsafety/index.html	585
129	http://www.cfsan.fda.gov/~dms/foodcode.html	567
130	http://www.mda.state.mi.us/food/basics/index.html	564
131	http://www.foodsafety.gov/~fsg/fsgadvic.html	553
132	http://www.foodsafety.gov/%7Efsg/fsgadvic.html	553
133	http://www.nraef.org/ifsc/	551
134	http://www.health.state.mn.us/divs/dpc/food/news/archv.htm	539
135	http://www.fmi.org/consumer/foodkeeper/refriger.htm	536
136	http://www.foodsafety.gov/~fsg/fsgnews.html	535
137	http://www.foodsafety.gov/%7Efsg/fsgnews.html	535
138	http://www.cfsan.fda.gov/~dms/fsterr.html	534
139	http://www.extension.iastate.edu/foodsafety/contacts.html	531
140	http://food4kids.missouri.edu/intro.html	524
141	http://www.fsis.usda.gov/OA/pubs/recallfocus.htm	517
142	http://www.foodsafety.gov/~fsg/irradiat.html	516
143	http://www.fsis.usda.gov/OA/pubs/jerky.htm	515
144	http://www.agr.state.nc.us/cyber/kidswrld/foodsafe/Foodlink.htm	514
145	http://www.agr.state.nc.us/cyber/kidswrld/foodsafe/facts/Sffacts.htm	513
146	http://www.state.oh.us/agr/FoodSafetyRlses.html	513
147	http://www.nal.usda.gov/fnic/etext/000056.html	509
148	http://www.ift.org/publications/jfs/index.shtml	509
149	http://www.ifis.org	502
150	http://www.ers.usda.gov/briefing/ConsumerFoodSafety/gallery/risks.htm	499
151	http://www.ers.usda.gov/Briefing/consumerfoodsafety/gallery/risks.htm	499
152	http://www.ers.usda.gov/Briefing/ConsumerFoodSafety/gallery/risks.htm	499
153	http://www.ifrn.bbsrc.ac.uk	497
154	http://www.state.oh.us/agr/FoodSafetyDivLinks.html	479
155	http://www.cfsan.fda.gov/~dms/opa2pmnc.html	475
156	http://www.fsai.ie	470
157	http://www.dfst.csiro.au/foodfacts.htm	464
158	http://www.extension.iastate.edu/foodsafety/	460
159	http://hna.ffh.vic.gov.au/phb/hprot/food/safefood/contents.html	457
160	http://www.nal.usda.gov/fnic/etext/000020.html	456
161	http://www.mda.state.mi.us/food/index.html	450
162	http://www.cfsan.fda.gov/~dms/seniors.html	449
163	http://www.extension.iastate.edu/foodsafety/special.html	449
164	http://www.ift.org/publications/ft/index.shtml	448
165	http://www.fao.org	444
166	http://www.fsis.usda.gov/OA/pubs/aids.htm	435
167	http://pigtrail.uark.edu/news/2000/may00/tyson.html	431
168	http://www.extension.iastate.edu/foodsafety/Lesson/	427
169	http://www.extension.iastate.edu/foodsafety/databases.html	421

170	http://www.agr.state.nc.us/cyber/kidswrld/foodsafe/FoodQuiz.html	420
171	http://www.foodsafety.iastate.edu/reprint.html	416
172	http://www.who.it/Ht/Food_safety.htm	412
173	http://www.fsis.usda.gov	409
174	http://www.fsis.usda.gov	409
175	http://www.ift.org	408
176	http://www.foodsafety.gov/~fsg/foodlaw.html	406
177	http://www.mda.state.mi.us/food/alliance.html	405
178	http://www.cdc.gov/foodsafety/	404
179	http://www.extension.iastate.edu/foodsafety/news.html	400
180	http://www.health.state.mn.us/divs/dpc/food/irrfood/irrd.htm	398
181	http://ag.utah.gov/regsvcs/foodcomp.htm	391
182	http://www.fmi.org/consumer/foodkeeper/freezing.htm	381
183	http://www.ers.usda.gov/briefing/ConsumerFoodSafety/purchasing/index.htm	369
184	http://www.ers.usda.gov/Briefing/consumerfoodsafety/purchasing/index.htm	369
185	http://www.ers.usda.gov/Briefing/ConsumerFoodSafety/purchasing/index.htm	369
186	http://www.cfsan.fda.gov/~dms/fsiupd09.html	361
187	http://www.fda.gov/bbs/topics/ANSWERS/2001/ANS01105.html	360
188	http://www.acsh.org/publications/booklets/irradiated.html	359
189	http://www.ers.usda.gov/briefing/IndustryFoodSafety/pdfs/liability.htm	355
190	http://ccr.ucdavis.edu/irr/index.shtml	350
191	http://www.acsh.org/food/index.html	336
192	http://www.bcveg.com/prod03.htm	333
193	http://www.nal.usda.gov/fnic/foodborne/fbindex/025.htm	332
194	http://www.nal.usda.gov/foodborne/fbindex/025.htm	332
195	http://www.fsis.usda.gov/OA/pubs/washing.htm	329
196	http://www.nal.usda.gov/fnic/foodborne/fbindex/036.htm	329
197	http://www.nal.usda.gov/foodborne/fbindex/036.htm	329
198	http://ctr.uvm.edu/ext/nfsh/	323
199	http://www.cfsan.fda.gov/~dms/opa-toc.html	320
200	http://www.health.state.mn.us/divs/dpc/food/foodsafe.htm	313
201	http://www.inspection.gc.ca/english/corpaffr/recarapp/recaltoce.shtml	312
202	http://nafs.tamu.edu/structure.htm	299
203	http://www.nal.usda.gov/fnic/foodborne/fbindex/006.htm	299
204	http://www.nal.usda.gov/foodborne/fbindex/006.htm	299
205	http://www.cdc.gov/foodsafety/partners.htm	295
206	http://www.fsis.usda.gov/OA/pubs/cfg/cfg7.htm	293
207	http://www.cdc.gov/foodsafety/image.htm	291
208	http://www.fsis.usda.gov/OA/programs/whatdoes.htm	290
209	http://www.nal.usda.gov/fnic/foodborne/fbindex/016.htm	289
210	http://www.nal.usda.gov/foodborne/fbindex/016.htm	289
211	http://www.foodsafetynetwork.ca/food/fd-ills-HC-data-FandC-may99.htm	282
212	http://www.extension.iastate.edu/foodsafety/ccp/point2.html	280
213	http://www.health.state.mn.us/divs/dpc/food/fscinfo/fscinfo.htm	280
214	http://www.mda.state.mi.us/food/basics/serve.html	280
215	http://www.uark.edu/depts/fsc/news.html	273
216	http://www.mda.state.mi.us/kids/countyfair/food/camping/index.html	271
217	http://www.dfst.csiro.au/fia.htm	268
218	http://www.fsis.usda.gov/OA/pubs/mailorder.htm	266
219	http://www.fsis.usda.gov/OA/pubs/focus_ref.htm	264
220	http://www.foodsafety.gov/%7Efsg/fsglang.html	262
221	http://www.foodaust.com.au	250
222	http://www.nal.usda.gov/fnic/Fpyr/pyramid.html	245
223	http://www.msue.msu.edu/msue/imp/modfs/masterfs.html	245
224	http://health.utah.gov/els/envsvc/	245
225	http://www.fsis.usda.gov/OA/pubs/consumerpubs.htm	233
226	http://www.health.state.mn.us/divs/dpc/food/fscinfo/contacts.htm	225
227	http://www.fsis.usda.gov/OA/pubs/facts_barbecue.htm	214

228	http://www.acsh.org/publications/priorities/0902/foodlyin.html	205
229	http://www.state.ak.us/dec/deh/sanitat/sanalert.htm	205
230	http://www.fsis.usda.gov/OA/news/xrecalls.htm	204
231	http://www.reeusda.gov/pas/programs/foodsafety/foodstates/usamap1.htm	198
232	http://www.ifis.org/forum/forum.html	190
233	http://www.ift.org/publications/jfs/engineering.shtml	181
234	http://www.ift.org/publications/jfs/microbiology.shtml	175
235	http://www.fsis.usda.gov/OA/pubs/focusrabbit.htm	172
236	http://www.mda.state.mi.us/food/basics/prepare.html	145
237	http://www.texasfoodsafety.org	144
238	http://www.nlm.nih.gov/medlineplus/foodallergy.html	133
239	http://vric.ucdavis.edu/selectnewtopic.foodsafety.htm	125
240	http://www.mda.state.mi.us/food/basics/store.html	122
241	http://www.ift.org/divisions/food_law/jumpmain.htm	101
242	http://www.mda.state.mi.us/food/survey.html	96
243	http://www.nasda-hq.org/nasda/nasda/Foundation/foodsafety/index.html	89
244	http://www.extension.iastate.edu/foodsafety/turkey.html	82
245	http://www.inspection.gc.ca/english/ppc/science/fsra/fsra_e.shtml	72
246	http://www.fda.gov/OHRMS/DOCKETS/98fr/081800a.txt	51
247	http://www.inspection.gc.ca/english/corpaffr/publications/fsydas/tabdese.shtml	37
248	http://www.inspection.gc.ca/english/ops/ofsr/ofsre.shtml	32
249	http://www.fsis.usda.gov/OA/topics/y2k.htm	28
250	http://www.iit.edu/~ncfs/	15
251	http://www.fda.gov/cvm/fsi/fsior/FSIOR.htm	12
252	http://www.inspection.gc.ca/english/corpaffr/newcom/2002/20020514e.shtml	12
253	http://www.oznet.ksu.edu/ext_f&n/newslet.htm	11
254	http://www.foodmicro.nl/Titelpagina.htm	10
255	http://www.nal.usda.gov/fnic/foodborne/fbindex/023.htm	9
256	http://www.nal.usda.gov/foodborne/fbindex/023.htm	9
257	http://www.foodonline.com/content/homepage/default.asp?VNETCOOKIE=NO	6
258	http://www.cdfa.ca.gov/ahfss/ah/food_safety.htm	3
259	http://www.fsis.usda.gov/OA/speeches/1998/cw_purdue.htm	2
260	http://www.fda.gov/cvm/fsi/fsi.html	2
261	http://www.cfsan.fda.gov/~dms/opa-antg.html	1
262	http://www.fao.org/worldfoodsummit/english/newsroom/focus/focus6.htm	1
263	http://www.acsh.org/publications/booklets/biotechnology.html	1
264	http://www.fao.org/es/ESN/riskcomm/riskcom4.htm	1

C AGROVOC categories

1. AGROVOC main categories

Category	Main Category Name
A	Agriculture
B	Geography and history
C	Education, extension, and advisory work
D	Administration and legislation
E	Economics, development, and rural sociology
F	Plant production
H	Protection of plants and stored products
J	Handling, transport, storage and protection of agricultural products
K	Forestry
L	Animal production
M	Aquatic sciences and fisheries
N	Machinery and buildings
P	Natural resources
Q	Food science
S	Human nutrition
T	Pollution
U	Auxiliary disciplines

2. AGROVOC subject categories

Each of the 115 categories below is a sub-category of one of the above main categories. The mapping is according to the matching capital letters.

Category	Descriptor
A01	Agriculture - General aspect
A50	Agricultural research
B10	Geography
B50	History
C10	Education
C20	Extension
C30	Documentation and information
D10	Public administration
D50	Legislation
E10	Agricultural economics and policies
E11	Land economics and policies
E12	Labour and employment
E13	Investment, finance and credit
E14	Development economics and policies
E16	Production economics
E20	Organization, administration and management of agricultural enterprises or farms
E21	Agro-industry
E40	Cooperatives
E50	Rural sociology

E51	Rural population
E70	Trade, marketing and distribution
E71	International trade
E72	Domestic trade
E73	Consumer economics
E80	Home economics, industries and crafts
E90	Agrarian structure
F01	Crop husbandry
F02	Plant propagation
F03	Seed production
F04	Fertilizing
F06	Irrigation
F07	Soil cultivation
F08	Cropping patterns and systems
F30	Plant genetics and breeding
F40	Plant ecology
F50	Plant structure
F60	Plant physiology and biochemistry
F61	Plant physiology - Nutrition
F62	Plant physiology - Growth and development
F63	Plant physiology - Reproduction
F70	Plant taxonomy and geography
H01	Protection of plants - General aspects
H10	Pests of plants
H20	Plant diseases
H50	Miscellaneous plant disorders
H60	Weeds
J10	Handling, transport, storage and protection of agricultural products
J11	Handling, transport, storage and protection of plant products
J12	Handling, transport, storage and protection of forest products
J13	Handling, transport, storage and protection of animal products
J14	Handling, transport, storage and protection of fisheries and aquacultural products
J15	Handling, transport, storage and protection of non-food or non-feed agricultural products
K01	Forestry - General aspects
K10	Forestry production
K11	Forest engineering
K50	Processing of forest products
K70	Forest injuries and protection
L01	Animal husbandry
L02	Animal feeding
L10	Animal genetics and breeding
L20	Animal ecology
L40	Animal structure
L50	Animal physiology and biochemistry
L51	Animal physiology - Nutrition
L52	Animal physiology - Growth and development
L53	Animal physiology - Reproduction
L60	Animal taxonomy and geography
L70	Veterinary science and hygiene
L72	Pests of animals
L73	Animal diseases
L74	Miscellaneous animal disorders
M01	Fisheries and aquaculture - General aspects
M11	Fisheries production
M12	Aquaculture production and management
M40	Aquatic ecology

N01	Agricultural engineering
N02	Farm layout
N10	Agricultural structures
N20	Agricultural machinery and equipment
P01	Nature conservation and land resources
P05	Energy resources and management
P06	Renewable energy resources
P07	Non-renewable energy resources
P10	Water resources and management
P11	Drainage
P30	Soil science and management
P31	Soil surveys and mapping
P32	Soil classification and genesis
P33	Soil chemistry and physics
P34	Soil biology
P35	Soil fertility
P36	Soil erosion, conservation and reclamation
P40	Meteorology and climatology
Q01	Food science and technology
Q02	Food processing and preservation
Q03	Food contamination and toxicology
Q04	Food composition
Q05	Food additives
Q51	Feed technology
Q52	Feed processing and preservation
Q53	Feed contamination and toxicology
Q54	Feed composition
Q55	Feed additives
Q60	Processing of non-food or non-feed agricultural products
Q70	Processing of agricultural wastes
Q80	Packaging
S01	Human nutrition - General aspects
S20	Physiology of human nutrition
S30	Diet and diet-related diseases
S40	Nutrition programmes
T01	Pollution
T10	Occupational diseases and hazards
U10	Mathematical and statistical methods
U30	Research methods
U40	Surveying methods

3. KAON RDFS representation of AGROVOC categories (extract)

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
    <!ENTITY a 'http://www.fao.org/agris/aos/agrovoc-categories.kaon#'>
    <!ENTITY kaon 'http://kaon.semanticweb.org/2001/11/kaon-lexical#'>
    <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
    <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
]>

<?include-rdf     logicalURI="http://kaon.semanticweb.org/2001/11/kaon-
root"
                  physicalURI="file:/C:/CVS/build/kaon/classes/edu/unika/
                  aifb/kaon/api/res/kaon-root.xml"?>
<?include-rdf     logicalURI="http://kaon.semanticweb.org/2001/11/
                  kaon-lexical"
                  physicalURI="file:/C:/CVS/build/kaon/classes/edu/unika/
                  aifb/kaon/api/res/kaon-lexical.xml"?>

<rdf:RDF xml:base="http://www.fao.org/agris/aos/agrovoc-categories.kaon#"
    xmlns:a="&a;"
    xmlns:kaon="&kaon;"
    xmlns:rdf="&rdf;"
    xmlns:rdfs="&rdfs;">

<kaon:Label rdf:ID="1031926074615-166128520"
    kaon:value="Agriculture">
    <kaon:inLanguage rdf:resource="&kaon;en"/>
    <kaon:references rdf:resource="#A"/>
</kaon:Label>
<kaon:Label rdf:ID="1031926074746-1058167586"
    kaon:value="Agriculture - General aspect">
    <kaon:inLanguage rdf:resource="&kaon;en"/>
    <kaon:references rdf:resource="#A01"/>
</kaon:Label>

<a:Category rdf:ID="A"/>
<a:Category rdf:ID="A01">
    <a:subCategoryOf rdf:resource="#A"/>
</a:Category>

<rdfs:Class rdf:ID="Category">
    <rdfs:subClassOf rdf:resource="&kaon;Root"/>
</rdfs:Class>

<rdf:Property rdf:ID="subCategoryOf">
    <rdfs:range rdf:resource="#Category"/>
    <rdfs:domain rdf:resource="#Category"/>
</rdf:Property>

</rdf:RDF>
```

D Results of Ontology Integration into automatic text classification

Avg(Precision)			Training Examples						
Ontology	**Concept Depth**	**Concept Integration Mode**	**5**	**10**	**20**	**30**	**40**	**50**	**Total**
False	**-**	**-**	0,4577	0,5433	0,5984	0,6235	0,6436	0,6570*	0,5840
True	**1**	**Add**	0,4821	0,5411	0,5995	0,6238	0,6350*	0,6551	0,5868
		Replace	0,4651	0,5336	0,5748	0,6134	0,6284*	0,6471	0,5724
		Only	0,4664	0,5292	0,5699	0,6092	0,6246	0,6227	0,5703
	2	**Add**	0,4610	0,5385	0,5998	0,6195*	0,6432	0,6583	0,5852
		Replace	0,4470	0,5256	0,5790	0,6149*	0,6239	0,6382	0,5686
		Only	0,4550	0,4990	0,5592	0,5830*	0,5976	0,6128*	0,5479

Table 19: Performance of $X_{single\_en\_Cat}$ with ontology background knowledge, averaged precision over 30 runs

Avg(Precision)			Training Examples							
Ontology	**ConceptDepth**	**Concept Integration Mode**	**5**	**10**	**20**	**30**	**40**	**50**	**60**	**Total**
False	**-**	**-**	0,4695	0,5392	0,5889*	0,6312*	0,6281	0,6399	0,6387	0,5907
True	**1**	**Add**	0,4954	0,5590	0,6119	0,6243	0,6474	0,6474	0,6424	0,6040
		Replace	0,4883	0,5563	0,6060	0,6240	0,6288	0,6448	0,6471	0,5993
		Only	0,4717	0,5179	0,5921	0,6028	0,6067	0,6060	0,6243	0,5745
	2	**Add**	0,4900	0,5427	0,6206	0,6289	0,6418	0,6536	0,6473	0,6036
		Replace	0,4722	0,5522	0,6076	0,6269	0,6256	0,6543	0,6629	0,6003
		Only	0,4433	0,5371	0,6059	0,6167	0,6091	0,6215	0,6201	0,5791

Table 20: Performance of $X_{single\_fr\_Cat}$ with ontology background knowledge, averaged precision over 30 runs

Avg(Precision)			Training Examples						
Ontology	**Concept Depth**	**Concept Integration Mode**	**5**	**10**	**20**	**30**	**40**	**50**	**Total**
False	**-**	**-**	0,4517	0,5140	0,5708	0,5765	0,5824	0,6114	0,5512
True	**1**	**Add**	0,4311	0,5073	0,5594	0,5953	0,5784	0,5897	0,5435
		Replace	0,4387	0,5286	0,5737	0,5933	0,5930	0,6012	0,5547
		Only	0,3444	0,3870	0,4238	0,4415	0,4179	0,4501	0,4108
	2	**Add**	0,4162	0,5086	0,5629	0,5903	0,5883	0,5964	0,5438
		Replace	0,4276	0,4848	0,5533	0,5675	0,5755	0,5890	0,5329
		Only	0,3292	0,3702	0,3962	0,4288	0,4128	0,4091	0,3910

Table 21: Performance of $X_{single\_es\_Cat}$ with ontology background knowledge, averaged precision over 15 runs