Ontology Based Machine Learning for Semantic Multiclass Classification

François-Élie Calvier, Michel Plantié, Gérard Dray, Sylvie Ranwez

Ontology Based Machine Learning for Semantic Multiclass Classification

François-Élie Calvier[1], Michel Plantié[1],Gérard Dray[1], Sylvie Ranwez[1]

LGI2P Ecole Nationale Supérieure des Mines d'Ales Site EERIE
Parc Scientifique Georges Besse 69, rue Georges Besse 30035 NIMES Cedex 1
firstname.lastname@mines-ales.fr

Abstract. Following the development of semantic web technologies, many ontologies and thesauri have been proposed to index resources during the last decade. However, despite their expressiveness, those knowledge models do not always cover all the points of interest within dedicated applications. Therefore, alternative *ad hoc* taxonomies have been developed to support resources classifying processes.

This paper proposes a method that bridges existing knowledge models with *ad hoc* taxonomies to address the problem of textual documents classification. Usually, documents are indexed according to different knowledge models: keywords, thesauri, ontologies. Nevertheless, for a project leader, additional information are needed to organize documents. In response to a particular need of one of our partners, we have developed a learning method based on the use of ontologies for modelling a semantic classification process. This method allows the expert user to match their needs by optimising text document classification.

1 Introduction

Scientists, project leaders or managers are overwhelmed by numerous digital data. Most of the time, these resources are generated from different communities and may have been characterized using different and *ad hoc* indexing systems (keywords, thesauri, ontologies, ...) according to dedicated applications. However, within a project, coordinators may have their own way of organizing and classifying resources, particularly textual ones, usually with respect to several classes. This classification process may be very personal and does not always take into account previous indexing. One of our institutional partners faces this situation. The ITMO Cancer[1] is a French consortium that coordinates different research institutes under the AvieSan[2]. It is in charge of coordinating biomedical research and projects in the field of oncology domain in France. Within this consortium, projects are classified following CSO classification (Common Scientific Outline), a *ad hoc* classification system organized into seven broad areas of

[1] https://itcancer.aviesan.fr/
[2] Alliance nationale pour les sciences de la vie et de la santé - French National Alliance for Life Sciences and Health

scientific interest in cancer research. This classification is highly used by ITMO Cancer governance even if it does not take into account other indexing resources as MeSH[3] (Rogers (1963)) used by PubMed[4] to index biomedical publications. Yet it is of particular interest to building bridges between these resources (here CSO and MeSH), to infer some complementary knowledge. Indeed, by having the description of one area in CSO using MeSH knowledge, it will be possible to identify relevant bibliography of the domain or experts of this domain.

From textual documents classified into several categories, the system proposed in this paper uses domain ontology to learn to learn semantic descriptions of those categories. Then using lexical analysis associated with domain ontology it automatically classifies any new documents. Since a document may be classified into several categories the method is called multiclass classification. This work aims at reducing user intervention in the construction of the classification model, but still fits its own logic and customized process. Our learning method starts with the analysis of some document classification made by an end-user. Therefore it analyses indexes that have been associated with each document using domain ontology. The efficiency of the system depends strongly on the index quality.

Indexing process has been tackled in two ways in literature. On one hand, the machine learning community proposes a text based classification approach, in which categories can be easily handled by the users. Nevertheless, being exclusively based on text, i.e. on sets of words, the semantic aspect is poorly exploited and it is hard to deal with multiple indexing. This problem occurs especially when there are few indexed documents with respect to the number of categories. On the other hand, the knowledge engineering community proposes a semantic approach, i.e. based on domain ontology. Semantic aspects of the texts are then managed through entities defined within these ontologies (concepts, relations and instances). However, since ontologies are not easy to grasp and to exploit by human operators, they should be hidden through a dedicated interface.

In this paper we propose a novel approach which combines semantic strength from the knowledge engineering community and the user adaptation permitted by machine learning classification. As for machine learning, the user can define his own categories. Categories are sets of documents and do not refer to any ontology nor semantic explanation. A semantic description of each document is produced using MeSH concepts. The semantic descriptions of all the documents that belong to a same category are aggregated to get a semantic description of this category. When a new document is provided, its semantic description is compared to all category semantic descriptions in order to assess the categories which best suit the document. Our approach particularly focuses on multiple indexing with poor learning sets.

The next section presents the background of our approach and insists on ontological v.s. lexicological solutions that have been proposed in literature for

[3] Medical Subject Headings http://www.ncbi.nlm.nih.gov/mesh
[4] http://www.ncbi.nlm.nih.gov/pubmed

classification. Then the semantic multi-class classification is presented through its two main phases: the semantic learning (section 3) and the classification process (section 4). Section 5 discusses the results through evaluation and results analysis, while the last section gives conclusions and perspective on this work.

2 Background

The research work presented here derives from two main fields: ontological and lexical analysis and classification theory. This section starts with a survey of semantic based methodologies particularly the ones that relies on hybrid approaches. Then classification principles are described.

2.1 Related work in ontological and lexical analysis

During last decade, research work has underlined pros and cons of the lexical v.s. ontological approaches. If lexical approaches seem to be tailored to open contexts (like the Web) dealing with heterogeneous and unstructured data, ontology based approaches are more dedicated to specialized domains. They are efficient in many applications, such as information retrieval, but require high formalization and conceptual indexing. In this case the document index is often implemented at the document level where a lexical index may be attached to a fine-grained description at the sentence level. Hence there is a continuum of solutions from terminology to ontology based systems (OntoLex approaches). The association of lexical data with domain ontology is also called onto-terminology (Roche et al. (2009)). Indeed, even if they have been considered as competitors end exclusive, both lexical and ontological approaches are nowadays recognized as complementary (Ranwez et al. (2013)). After experiencing this with information retrieval, we present in this paper a hybrid solution for document classification.

Many lexical approaches aim at extracting concepts and relations from texts in order to build ontologies or to populate them. That is called ontology learning Maedche and Staab (2000)). In Dinh and Tamine (2011), a similar method for concept extraction is proposed. A weight is computed for each word (term) of a concept label and for each word of a document. Then, for each document, concepts are ranked using a cosine similarity based on those weights and a correlation measure based on the order of the words appearing in the document and in the labels. Finally, the N top-concepts are selected as the index kernel of the document. In our approach, we do not index documents with concepts. Although extracted concepts are used to describe the document, this description is not the final result. Thus, later steps of our process can deal with a slightly faulty extraction while the index kernel can not suffer any mis-extracted concept.

The recent multi-label classification problem is usually reduced to the simple label classification problem which has been widely studied. In de Carvalho and

Freitas (2009), an overview of the most common approaches is proposed. Although most of these approaches aim at providing a single best set of categories, some works like Brinker et al. (2006) propose a ranking type result.

In Rubin et al. (2011), the authors point out the weakness of machine learning approaches to multi-label classification on real data sets. The main causes of this weakness are the large number of labels per document and the skewed distribution of the labels. The author proposes to use dependency-LDA model in order to deal with little used labels.

The number of labels is also the focus of Charte et al. (2012). The reduction to a simple label problem is done either through a data transformation consisting in studying separately if a document belongs to each category, or through a method transformation consisting in considering new categories referring to any combination of initial categories. This implies the loss of category interdependence in the former case, and a large number of categories in the latter. The authors propose to complete a data transformation approach with dependency rules between categories. A dependency rule is a probabilistic representation of category interdependence.

In our approach, the result is a ranking of the initial categories. The interdependence between categories is captured through integration of the knowledge of virtual categories into initial categories. We deal with the category proliferation by considering only the virtual categories indexing at least a minimal number of documents.

2.2 Overview of classification process

Classification is a two step process. During the first step, the system exploits documents indexed by an expert in order to identify the differences between documents of each category. Both documents and their indexes are given to the system during this step. The second step consists in providing a (set of) category (ies) to index a new document.

A category is a set of documents with a label (e.g. *Cancer Initiation: Alterations in Chromosomes*). The label consists in one or several words. In our setting, documents are texts. No semantic information is available about categories and documents. The first step of our approach consists in learning the semantic aspect of the classification using a domain ontology the MeSH in our case) i.e. to categorize a semantic model for each category and thus to *learn* the expert's understanding of the categories. The second step of the approach is the classification of new documents. For each new document, the system suggests categories that would be the predicting expert's choices.

Learning the semantic aspect of a category is a three steps process described in Fig. 1.

First comes a selection within the document set of each category. The categories are the ones from the *ad hoc* classification system, which is CSO in our case. Then, the description of each selected document is extracted. These descriptions use only concepts from the domain ontology, i.e. MeSH for our application.

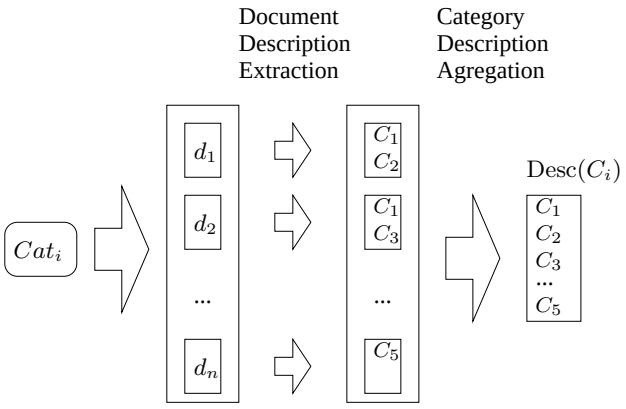

Fig. 1. Simplified vue of Learning process

Finally, the description of each category is obtained by aggregation of the description of its selected documents. The result is then a conceptual description built using concepts from the domain ontology (MeSH) for each of the categories from the *ad hoc* classification system (CSO).

3 Semantic learning

Since no semantic description of the categories is available, we have to build it. The category itself is very poorly described but contains documents. In our case, categories are those of the CSO and contain documents (projects) according to experts from ITMO Cancer. A semantic description of the category can be built from its documents. Thus, a preliminary task consists in building a semantic descriptor of documents. Then, we propose an aggregation method of those descriptors into a semantic description of the category.

3.1 Semantic document description

A document is a simple text written in English which can be seen as a list of words. Machine learning approaches apply statistical measures on word occurrence and co-occurrence to state a descriptor of each document as a vector of words. Empty-sense word problem is addressed using TF.IDF measures (Jones (1972); Salton and McGill (1986)).

Instead of considering vectors of words, we propose to consider vectors of concepts. Using concepts is interesting because the empty word problem become an empty concept problem. We thus split it into a meaningfulness-identification problem and a meaning-relevance-identification problem. These are the two steps of our semantic document description extraction process: first, concept extraction from the document to describe and, second, concept weighting from the complete corpus.

Moreover, using concepts is also interesting because, if the domain of the document is known, and if an ontology of this domain exists, we can assume that the only relevant concepts for the document description are those from the domain ontology. Thus, concept extraction is turned into checking the presence in the document of every concept from the ontology. This presence may be detected, thanks to the presence of one of concept labels within documents or through identification of associated lexicons (see "Concept identification through lexical analysis" in Ranwez et al. (2013))

Definition 1 ($Desc(Doc)$).
Given Doc a document, and C_1, C_2,...C_n the concepts from a domain ontology, then $Desc(Doc) = \{w_1, w_2, \ldots, w_n\}$ is the semantic description of Doc where w_i is the weight of C_i.

We assume that reference domain ontologies are available. In a reference ontology, concepts have several normalized labels. When labels are simple, concept extraction may consist in seeking each label in the given document. A lemmatisation of the document would lead to better results. In our study, we exploit MeSH in which labels are named *terms*. MeSH terms can be complex nominal groups. Thus, they barely ever appear in a document under their complete form. Nevertheless, they can appear in a partial form. It is not relevant to seek exact occurrences of a MeSH term in a document. In order to seek partial occurrences of a MeSH term in a document, we define the smallest partial form of a term which is semantically representative of the MeSH term. We call this smallest partial form a *semantic unit*.

A MeSH term is a nominal group. Thus, the grammatical elements carrying the meaning are nouns (Kleiber (1996)). This is why we build semantic units upon nouns. We propose to compare semantic units extracted from the document with the semantic units extracted from each of the MeSH terms. If a semantic unit from a MeSH term is also a semantic unit from the document, then the MeSH term is considered as found in the document.

The complexity of the semantic units is relevant. For example, considering single noun semantic units and two concepts C_1 and C_2 such as C_2 is more specific than C_1 (i.e. C_2 is a kind of C_1). In this case, C_2 can possess a label which is built upon one of C_1's by addition of one or several words. For instance, the concepts C_1=*Neoplasms*, C_2=*Breast Neoplasms*, and C_3=*Inflammatory Breast Neoplasms* are listed from the most general to the most specific. Then, C_1 has one semantic unit (i.e. *neoplasms*), C_2 and C_3 have 2 semantic units (i.e. *neoplasms* and *breast*). When the semantic unit *neoplasms* is found in a document, then the occurrence number is incremented for the 3 concepts. In the same way, when the semantic unit *breast* is found in a document, then the occurrence number is incremented for C_2 and C_3. Using too simple semantic units leads be unable to distinguish the occurence of specific and general concepts in a document.

Moreover, it also leads to mistakes since the semantic units of a general concept C can be semantic units of all the concepts more specific than C. Thus, in our example, *neoplasms* is also a semantic unit of the concept *Bone Neoplasms* which then is considered as found in any document about breast cancer.

A *too-simple-built* semantic unit leads to noisy matching. On the other hand, a *too-complex-built* semantic unit would heighten silence, like seeking MeSH terms does. We propose two kinds of semantic units based on couple of words (w_1, w_2):

- a bigram shape semantic unit where w_1 and w_2 are two nouns from the same sentence s in the document. w_1 is before w_2 in s and if w_3 is a word of s such as w_3 is between w_1 and w_2 then w_3 is not a noun.
- a nominal group shape semantic unit where w_1 is a noun and w_2 is a noun or an adjective from the same sentence s in the document.

Due to the complexity of MeSH terms, an exact term extraction is not possible. Thus we split MeSH terms into semantic units to solve the silence problem. Unlike MeSH terms, semantic units are not unique and thus, introduce some additional noise. Building *not too simple semantic* units addresses this additional noise problem. To complete this task, we propose dropping semantic units which are relative to a high number of concepts. These semantic units will be called empty semantic units. The concept extraction process is depicted in Fig. 2.

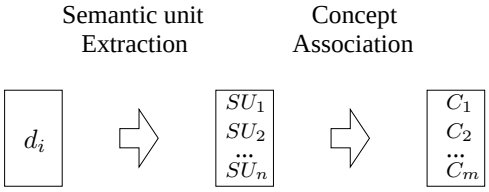

Fig. 2. Concept extraction

Semantic units are identified with the help of a morpho-syntactical analyser such as TreeTagger (Schmid (1994)).

At the end of concept extraction, we assume that all extracted concepts are meaningful. We then want to assess how relevant they are for the description of the given document. The relevance is represented by the weight of the concept in the concept vector of the document. The number of occurrences of a MeSH term is the sum of the number of distinct occurrences of its semantic units. In the same way, the number of occurrences of a concept is the sum of the number of distinct occurrences of its relative MeSH terms. Given this, we can apply the usual TF or TF.IDF measures of relevance to the occurrence number of a concept in a document.

At this point, we have built a semantic description of a document. The description of indexed documents will be exploited to construct the semantic description of the categories. We describe this construction in section 3.2. The description of unindexed documents will be exploited to propose a classification of the described documents. This process is explained in section 4.

3.2 Semantic category description

A category is a collection of documents. Thus it can be described as an aggregation of the semantic descriptions of this document collection. As it has been done for documents, we define the semantic description of a category as a vector of concepts from the domain ontology.

Definition 2 ($Desc(Cat)$).
Given Cat a category, and C_1, C_2,... C_n the concepts from a domain ontology, then $Desc(Cat) = \{w_1, w_2, \ldots, w_n\}$ is the semantic description of Cat where w_i is the weight of C_i.

The occurrence number of a concept C for a given category Cat is the number of documents Doc indexed by Cat with a non null weight for C in $Desc(Doc)$. The relevance of a concept for Cat and thus its weight in $Desc(Cat)$ can be obtained by application of TF or TF.IDF measures to this occurrence number of a concept in a category.

When many documents indexed with a single category Cat share a same common concept C, it may be assessed that C is important for Cat and then should have a high weight in $Desc(Cat)$. But when many documents indexed with at least two categories Cat_1 and Cat_2 share a same common concept C, we cannot asses whether C is important for Cat_1, Cat_2 or both. This problem can be dealt with by considering multiple indexes as virtual categories and then apply to them the same category description learning process than for any real category. This statistical approach would be suitable as long as there are enough documents in each category. In our context, real categories having few documents, introducing virtual categories would cause single document categories.

Besides, we have built a semantic description of a category in order to index new documents. This description should take into account the importance of concepts for a category with regards to their importance for other categories. Using TF measure, the cross importance is not taken into account. Using TF.IDF measure, the overall importance is taken into account for each category. But none of these measures allow to take into account the importance of a concept in a subset of the categories. For example, given Cat_1 and Cat_2 two categories, such as $Desc(Cat_1) = \{w_1, w_2, \ldots, w_{i-1}, w_i, w_{i+1}, \ldots, w_n\}$ and $Desc(Cat_2) = \{w_1, w_2, \ldots, w_{i-1}, w'_i, w_{i+1}, \ldots, w_n\}$ using a TF measure. If C_i is considered as a common concept, when applying TF.IDF, $|Desc(Cat_2) - Desc(Cat_1)|$ would tends to 0. The system would then be unable to distinguish Cat_1 from Cat_2.

Furthermore, a category can contain documents having in their respective description different concepts which have a lowest common ancestor in the ontology. Even if this ancestor never appears in any document, its belonging to the category description is relevant. The importance of such an ancestor concept for a given category is proportional to the sum of the importance of each of the more specific concepts found in the descriptions of the documents into this category.

At the end of the semantic learning process, we have built a semantic description of each category.

4 Classification

The objective of a classification process is to propose a (set of) category(ies) to index a given document. We have detailed how to build a semantic description of the document (section 3.1) and then a semantic description of each category (section 3.2). In both cases, the semantic description is a vector of concepts from the domain ontology. Our classification process is based on a comparison of the description of a document referring to the description of each categories. This comparison provides a (dis)similarity value of the given document with the categories. Categories having the highest similarity value w.r.t. the document is then proposed to index the document. We first describe the similarity measures we propose to compare document description to category description. Then we propose a solution to deal with the multi category problem specificities. As it is often proposed in literature, the similarity between two semantic entities depends on the features they share (their *commonality*) and their *differences* (Tversky (1977)). Following these approaches we choose to discuss both aspects separately in the following subsections.

4.1 Similarity measures (commonality)

The comparison of description vectors has to take into account their specificity. The similarity is based on the set of concepts in common within both the document description and the category description. But it shall take into account proper-concept sets. Since all the descriptions are made of concepts from the same ontology, the structure of this ontology can also be exploited to reinforce the similarity measure. In order to distinguish the weight of a concept in a category description vector from its one from a document description vector, we note them respectively w_{Cat} and w_{Doc}.

4.1.1 Shared concepts

Whatever the choices made for their construction, there is a set of concepts common to a document description and a category description. Without further information about the importance of each concept, we can assume that the larger this set, the closer both descriptions. Thus, the similarity measure can been expressed as a function of the size of the common concept set.

Definition 3 (CSS Similarity).
$sim(Doc, Cat) = |C \in Doc \cap Cat|$

Nevertheless, since our descriptions are vectors, we also know the importance of each concept to the document and to the category perspectives. We define a Common Set Weight (CSW) Similarity measure as follows:

Definition 4 (CSW Similarity).
$sim(Doc, Cat) = \alpha_{\in [0\ 1]} \sum w_{Doc}(C \in Doc \cap Cat) \times (1 - \alpha) \sum w_{Cat}(C \in Doc \cap Cat)$

This definition of the CSW similarity is based on a macro weighting approach. Indeed, the weight of the common concept set is computed on one hand according to the category description and on the other hand according to the document description. The similarity measure is then a function of those two weights. A micro approach would have computed only one weight using both descriptions: a hybrid weight would then been computed for each concept of the set. A macro approach was preferred to a micro approach because the α parameter can be chosen after the weight computing. Thus, this choice can depend on the value distribution in the weight sets of each document.

In our study, the α parameter was only tested at 0, 0.5 and 1 values. When α is set to 1 (resp. 0), we define the particular CSW similarity based only on the document (resp. category) description weight.

Definition 5 (Document Balanced CSW Similarity).
$sim(Doc, Cat) = \sum w_{Doc}(C \in Doc \cap Cat)$

Definition 6 (Category Balanced CSW Similarity).
$sim(Doc, Cat) = \sum w_{Cat}(C \in Doc \cap Cat)$

Those CSW similarity variants are used as a tool-kit for a preliminary study. The aim is to evaluate the CSW similarity without considering the α parameter setting problem.

4.1.2 Proper concepts (difference)

Similarity can be completed by a dissimilarity evaluation. For instance, if the similarity measures between a document and two categories are identical then, the document is closer to the category with which the dissimilarity is the lowest. The dissimilarity measure is based on the sets of concepts proper to the category or to the document. Just as the similarity, the dissimilarity may be defined w.r.t. the sizes or the weights of each of these sets.

In our study, we have chosen to introduce dissimilarity within the Common Set and Proper Sets (CSPS) similarity measure as follows :

Definition 7 (CSPS Similarity).
$sim(Doc, Cat) = \alpha_{\in [0\ 1]} \frac{\sum w_{Doc}(C \in Doc \cap Cat)}{\beta \sum w_{Doc}(C \in Doc \setminus Cat)} \times (1 - \alpha) \frac{\sum w_{Cat}(C \in Doc \cap Cat)}{\gamma \sum w_{Cat}(C \in Cat \setminus Doc)}$

We only present one definition of the CSPS similarity but, as the CS similarity in section 4.1.1, we have studied this similarity measure w.r.t. to the sizes and weights of the involved sets. We also applied the same restriction on the α parameter values.

The CSPS similarity also allows two asymmetric definitions when only one proper set is considered.

Definition 8 (Category Asymmetric CSPS Similarity).
$sim(Doc, Cat) = \alpha_{\in[0\ 1]} \sum w_{Doc}(C \in Doc \cap Cat) \times (1-\alpha) \frac{\sum w_{Cat}(C \in Doc \cap Cat)}{\gamma \sum w_{Cat}(C \in Cat \setminus Doc)}$

Definition 9 (Document Asymmetric CSPS Similarity).
$sim(Doc, Cat) = \alpha_{\in[0\ 1]} \frac{\sum w_{Doc}(C \in Doc \cap Cat)}{\beta \sum w_{Doc}(C \in Doc \setminus Cat)} \times (1-\alpha) \sum w_{Cat}(C \in Doc \cap Cat)$

Those asymmetric definitions correspond to specific values of the β and γ parameters. As for the balanced CSW similarities, they are used as an adjuster for this preliminary study.

4.1.3 Ontology structure exploitation

Concepts in the description vectors are components of an ontology. Thus, they are organised along a hierarchical structure. During the concept aggregation described in section 3.2, this structure was exploited in order to reveal under expressed concepts. A similar problem is encountered during the construction of the shared concept set. For instance, considering two concepts C_1 and C_2 such as C_1 is an ancestor of C_2. When a category description contains C_1 and a document description contains C_2, C_2 is actually considered as a proper concept of the document.

Our proposal is to add for each concept C_2 from the document description, its lowest ancestor C_1 from the category description. The value of $w_{Doc}(C_1)$ is then risen by $w_{Doc}(C_2)$.

4.2 Multi category problem

Usually, the result provided by classifiers is the category that best matches the document. This category can be either a real or virtual category. In case of a virtual category, this implies that the document should be multi indexed.

We noticed in section 3.2, that in our context, the introduction of virtual categories causes a lack of documents for learning the semantic of categories. Thus providing the category which description is the most similar to the description of a document. Instead of providing a single result which can be a multiple index, we propose to provide an ordered set of the most similar categories for each document. The size of the set is the highest number of categories for one document in the learning set. Then, a document is called "well classified" if all the categories given by the expert belong to the most similar categories set provided by the system. In our experiment, we used a relaxed version of this definition. The

size of the result set is raised by the number of categories indexing the current document.

Recall and precision should also be carefully measured in a multi category setting. A naive macro-averaging would involve a multiple consideration of documents belonging to several categories. Thus, the more categories a document belongs to, the higher its weight would be. Moreover, the confusion matrix would make more documents appear than given in the evaluation set. In order to avoid this phenomenon, each document is weighted with the inverse of the number of categories it belongs to.

Finally, precision rate is measured with respect to our definition of "well classified" as described in table 1.

	Cat∈Exp	Cat∉Exp
Cat∈Res	true positive	if $rank_{Res \setminus Exp}(Cat) > \|Exp \setminus Res\|$ then true negative otherwise false positive
Cat∉Res	false negative	true negative

Table 1. Truth table of the **well classified** function

5 Experiments

We have conducted experiments from articles available on Pubmed and we have chosen MeSH as a reference domain ontology. The construction of the document set is described first. Then we describe the experimental protocol and finally, the results and their analysis.

5.1 Document set

Pubmed is a wide source of biomedical articles manually indexed using MeSH concepts. In order to avoid font problems and character identification, we have focused on the abstract of the articles which are available in a simple text version. The initial set was consisted of 36288 articles in the oncology domain indexed with 13997 concepts. Some of the abstracts were empty and some were semi-structured. We have chosen not to consider them because the keywords of the structure are meaningful but since they are keywords, their weight would be artificially raised. There were 27673 articles indexed with 5821 concepts remaining

Besides, we define a set of categories as a flat and anonymous version of the subset of the MeSH concepts indexing the Pubmed articles. This set of categories replace the MeSH indexing from this point. Most of the categories contain few

documents. For instance, only 1096 categories contain at least 20 documents. We have chosen not to consider categories with less than 100 documents which means 283 categories and 27126 documents.

Afterwards, we have focused on virtual categories. From the selected subsets of categories and documents, a set of 19368 virtual categories containing documents can be found. Within those virtual categories, only 1311 contain more than one document and none more than five. Thus, we have selected a subset of the categories according to the size of the involved virtual categories. Finally, we got a set of 11863 articles indexed by 24 categories and 32 virtual categories.

5.2 Protocol, results and analysis

Our experiment aims to compare the results obtained using all the semantic units, all the weighting methods and all the similarities described in this article. For each track, a 10-fold-cross-validation Geisser (1975) was used.

The best results are obtained using the nominal group shape semantic units for concept extraction, any weighted category descriptions and the Category Balanced CSW similarity measure . Tab. 2 describes the best results obtained for each kind of semantic unit. Each document description was used for learning only one category description.

semantic unit	well classified rate	precision	recall	f-measure
nominal group	**0.6**	0.51	**0.68**	**0.58**
pseudo-bigram	0.51	0.4	0.58	0.48
noun	0.47	0.38	0.59	0.46

Table 2. Best results. Obtained with the relaxed definition of **well classified documents**

Despite our expectations, pseudo-bigram results are similar to noun results. The low number of concepts in each document description obtained with a pseudo-bigram extraction may be the cause of this weakness. An extraction melting pseudo-bigrams and nouns in a balanced way could be interesting.

Tab. 3 describes the results obtained when using a document description for learning every category description it belongs to.

The results of these tracks are very low. The main cause is that more than half of the documents belong to several categories. Then learning the category descriptions becomes very noisy. In some cases, the classifier is unable to propose a result for half of the documents.

Tab. 4 presents the results obtained with the strict version of the *well classified* definition.

semantic unit	well classified rate	precision	recall	f-measure
nominal group	0.02	0.12	0.05	0.07
pseudo bigram	0.02	0.24	0.17	0.2
noun	0.02	0.2	0.26	0.22

Table 3. Best results of the experimental track based on multiple uses of document descriptions

semantic unit	well classified rate	precision	recall	f-measure
nominal group	0.55	0.34	0.39	0.36
pseudo bigram	0.45	0.28	0.37	0.32
noun	0.42	0.24	0.33	0.27

Table 4. Best results obtained with the strict definition of **well classified documents**

In this case, best results are obtained here using the CSPS similarity measure. As we expected, the relaxed version allows to have less mistakes with multi-category documents. This track also allows us to observe that relaxation does not modify the "well classified" document rate with respect to single category documents.

Besides, we also experiment our exploitation of the MeSH structure in the similarity measure. But the results of those tracks are identical to the results of the equivalent tracks not exploiting the MeSH structure.

6 Conclusion and perspectives

In this paper, we present a novel approach to assist and optimize the organization of digital text documents. The main features of our approach are: i. the initial indexing of the documents does not correspond to the desired classification, ii. a document can belong to multiple categories, iii. we seek to minimize user intervention in the design of the support system of classification. To do this, we start from a set of documents indexed by an ontology and classified into one or more categories by the user. This first categorization performed by the user constitutes the seed of our training system. By methods of machine learning, we create a support system for the classification process that models the semantic classification of the expert. The developed process is presented in detail in the first parts of the article. We also present an experiment conducted on a corpus of publications in oncology. All publications are indexed by the MeSH ontology. The results are promising and encourage us to continue our work in this area. The short-term outlook could be using the topology of the ontology to improve our classification results and generate sub-ontologies associated with each class defined by the expert.

Acknowledgements

The authors wish to thank Ansata Dada Balde Sy and Pierre Fontana for their contributions and help primarily on the constitution of the corpus of documents and the database. Our thanks also goes to our partner "ITMO Cancer" who has given us data and authorisation to use them in our experiments.

Résumé

Du fait du développement des technologies du Web sémantique, de nombreuses ontologies et thésaurus ont été proposés au cours de la dernière décennie afin d'indexer des ressources. Cependant, en dépit de leur expressivité, ces modèles de connaissances ne parviennent pas toujours à couvrir tous les aspects abordés dans des applications dédiées. Ainsi, des taxonomies *ad hoc* alternatives ont été développées pour guider les processus de classification de ressources.

Cet article propose une méthode qui construit un lien entre des modèles de connaissances existants et des taxonomies *ad hoc* afin de résoudre le problème de la classification de documents textuels. Habituellement, les documents sont indexés à l'aide de différents modèles de connaissances : mots-clés, thésaurus, ontologies. Toutefois, un chef de projet peut avoir besoin d'informations supplémentaires pour organiser les documents.

En réponse à un besoin d'un de nos partenaires, nous avons développé une méthode d'apprentissage basée sur l'utilisation d'ontologies pour modéliser un processus de classification sémantique. Nous présentons cette méthode ici et nous l'illustrons au travers d'une application réelle.

References

Klaus Brinker, Johannes Frnkranz, and Eyke Hllermeier. A unified model for multilabel classification and ranking. In *in proceedings of the 17th European Conference on Artificial Intelligence*, pages 489–493, 2006.

Francisco Charte, Antonio Jesús Rivera Rivas, María José del Jesús, and Francisco Herrera. Improving multi-label classifiers via label reduction with association rules. In *HAIS (2)*, pages 188–199, 2012.

Andre de Carvalho and Alex A. Freitas. A tutorial on multi-label classification techniques, 2009.

Duy Dinh and Lynda Tamine. Biomedical concept extraction based on combining the content-based and word order similarities. In *Proceedings of the 2011 ACM Symposium on Applied Computing*, SAC '11, pages 1159–1163, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0113-8.

S. Geisser. The predictive sample reuse method with applications. *Journal of The American Statistical Association*, 70(350), 1975.

Karen Sprck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972.

Georges Kleiber. Noms propres et noms communs : un problème de dénomination. In A. Oltramari, P. Vossen, L. Qin, and E. Eds. Hovy, editors, *Meta: translators' journal*, pages 567–589. Les Presses de l'Université de Montréal, 1996.

Alexander Maedche and Steffen Staab. Ontology learning from text. In Mokrane Bouzeghoub, Zoubida Kedad, and Elisabeth Mtais, editors, *Natural Language Processing and Information Systems, 5th International Conference on Applications of Natural Language to Information Systems, NLDB 2000, Versailles, France, June 28-30, 2000, Revised Papers*, volume 1959 of *Lecture Notes in Computer Science*, page 364. Springer, 2000. ISBN 3-540-41943-8.

Sylvie Ranwez, Benjamin Duthil, Moameth Francois Sy, Jacky Montmain, Patrick Augereau, and Vincent Ranwez. How ontology based information retrieval systems may benefit from lexical text analysis. In A. Oltramari, P. Vossen, L. Qin, and E. Eds. Hovy, editors, *New Trends of Research in Ontologies and Lexical Resources*, Theory and Applications of Natural Language Processing, pages 209–230. Springer, 2013. ISBN 978-3-642-31781-1.

Christophe Roche, Marie Calberg-Challot, Luc Damas, and Philippe Rouard. Ontoterminology - a new paradigm for terminology. In Jan L. G. Dietz, editor, *KEOD 2009 - Proceedings of the International Conference on Knowledge Engineering and Ontology Development, Funchal - Madeira, Portugal, October 6-8, 2009*, pages 321–326. INSTICC Press, 2009. ISBN 978-989-674-012-2.

Frank Rogers. Medical subject headings. *Bulletin of the Medical Library Association*, 51:114–116, January 1963. ISSN 0025-7338. URL http://www.ncbi.nlm.nih.gov/pmc/articles/PMC197951/.

Timothy N. Rubin, America Chambers, Padhraic Smyth, and Mark Steyvers. Statistical topic models for multi-label document classification. *CoRR*, abs/1107.2462, 2011.

Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986. ISBN 0070544840.

Helmut Schmid. Probabilistic part-of-speech tagging using decision trees, 1994.

Amos Tversky. Hfeatures of similarity. In *Psychological Reviews*, volume 84, page 327352, 1977.