Ontology-based Text Classification into Dynamically Defined Topics

Mehdi Allahyari[1], Krys J. Kochut[1], and Maciej Janik[2]
[1]Department of Computer Science
University of Georgia
Athens, GA, USA
[2]Sabre Holdings, Krakow, Poland
mehdi@uga.edu, kochut@cs.uga.edu, maciej.janik@gmail.com

Abstract— **We present a method for the automatic classification of text documents into a dynamically defined set of topics of interest. The proposed approach requires only domain ontology and a set of user-defined classification topics, specified as contexts in the ontology. Our method is based on measuring the semantic similarity of the thematic graph created from a text document and the ontology sub-graphs resulting from the projection of the defined contexts. The domain ontology effectively becomes the classifier, where classification topics are expressed using defined ontological contexts. In contrast to the traditional supervised categorization methods, the proposed method does not require a training set of documents. More importantly, our approach allows dynamically changing the classification topics without retraining of the classifier. In our experiments, we used the English language Wikipedia converted to an RDF ontology to categorize a corpus of current Web news documents into selection of topics of interest. The high accuracy achieved in our tests demonstrates the effectiveness of the proposed method, as well as the applicability of Wikipedia for semantic text categorization purposes.**

Keywords: Text Categorization, Background Knowledge, Topic filtering, Information Retrieval, Semantic Relatedness

I. Introduction

Text categorization is a task of assigning one or more predefined categories to the analyzed document, based on its content. People categorize text documents based on their general knowledge and their interest that determines which facts are treated as more important. While reading a news document we can capture most important actors, facts and places, connecting them into a one coherent event. Computers equipped with proper knowledge represented by an ontology that is comprehensive enough, can spot the same actors and facts in the document. Furthermore, using predefined semantic relationships between recognized entities and knowledge from the ontology, they can construct a model of a presented event, augmenting it with important background facts that are not directly present in the document. Interpretation of which facts and entities are more important is determined by dynamically defining the ontological context (topic) of interest. Using this information, the system can distinguish which facts are more important and focus the categorization on more precise information. Leveraging the knowledge from ontology directly in the categorization process not only allows us to skip the training step in building a categorizer but also to dynamically change the topics of the categorization without any retraining when the user's interests change.

Traditional supervised text categorization methods use machine learning or statistical approaches to perform the task. Most of them learn category definitions and create the categorizer from a set of training documents pre-classified into a number of fixed categories. Such methods, including Support Vector Machines [1], Naïve Bayes [1], decision trees [1], and Latent Semantic Analysis [2] are effective, but all of them require a set of pre-classified documents to train the categorizer.

In contrast to the traditional text classification methods, which rely on a set of training documents pre-classified into a number of fixed categories, we propose to use ontology and dynamically defined ontological contexts as classification categories. The novelty of our categorization method is that it does not require a training set of documents distributed into a fixed set of categories and relies exclusively on the knowledge represented in the ontology: (1) named entities, relationships between them, entity classification and the class hierarchy and (2) dynamically definable ontology contexts, representing the topics of interest (classification categories). In computing, ontology is usually defined as set of concepts within a particular domain, along with the relationships connecting those concepts. Ontology can be used to define knowledge about the domain, and consequently to reason about properties of that domain. Based on this definition, we can use a provided ontology as background knowledge that allows us to perform a task of text categorization in a given domain.

Since the proposed text categorization method relies exclusively on a supplied ontology, in a way, the ontology itself can be regarded as a classifier. Using a general, encyclopedic knowledge-based ontology, such as one based on Wikipedia, allows us to recognize and classify entities from numerous domains. Furthermore, having the ability to dynamically define classification categories turns such a classifier into a universal text classifier, as we can define our topics of interest as combinations of any existing domains in the ontology.

II. Ontology-based Text Categorization

We argue that automatic text classification can be accomplished by relying on the semantic similarity between the information included in a text document and a suitable fragment of the ontology. Our argument is based on the assumption that entities occurring in the document text along with relationships among them can determine the document's categorization, and that the entities classified into the same or similar domains in the ontology are semantically closely related to each other.

In the proposed approach, the ontology effectively becomes the classifier. In order to be able to achieve meaningful results, we require that the ontology (i) cover the categorization domain(s), (ii) include a rich instance base of named entities and meaningful relationships among them, (iii) have proper labels for named entities that enable their recognition in categorized documents, and (iv) have the entities classified according to a class taxonomy included in the ontology.

In our opinion, Wikipedia fulfills most of the requirements for text categorization purposes. Its major advantages are in the richness of represented domains, high number of entities, and in the included categorization scheme. Wikipedia includes a large number of entries (named entities), together with multiple alternative names for each entity, which makes it an excellent resource for recognizing a wide variety of entities. Extensive links connecting entity descriptions using infoboxes, templates, and simple hypertext references (hrefs) define a simple, yet strong backbone of semantic associations interconnecting entities. Furthermore, a broad category graph provides a usable categorization scheme.

Wikipedia was already successfully used for the supervised text categorization [3] and predicting document topics [4]. We successfully used an RDF ontology created from Wikipedia in our previous ontology-based text categorization experiments described in [5] and [6]. A related task of predicting concepts that characterize sets of documents using ontology created based on Wikipedia is presented in [7].

Wikipedia is the world largest encyclopedia to date containing a large number of well-defined and interconnected entities. They cover a multitude of domains of basic, encyclopedic knowledge, in many cases even describing highly specialized knowledge. Continuous updates and additions of entries (entities) make Wikipedia a very good source of up-to-date knowledge in different fields. For example, Wikipedia contains a large amount of health information and a well-developed biomedical domain.

Furthermore, the structure of Wikipedia facilitates the discovery of related entities (entries). This feature helps our categorization algorithm to find entities from within the same or similar domains, and as a result, focus on a specific thematic interpretation of the analyzed graph. Approaches to computing semantically related entities have been presented in [8] and [9].

However, there are some problems that must be addressed before using Wikipedia as ontology for text categorization. Wikipedia is "a free encyclopedia" and not an ontology. More specifically, it does not have explicitly defined semantic relationships, and it does not have a schema with a proper taxonomy.

Fortunately, most of these problems can be successfully solved or accounted for in the proposed categorization algorithm. A conversion of Wikipedia into a Wikipedia-based ontology has been done by the DBpedia project [10]. Categories in Wikipedia form a thesaurus [11] and not taxonomy, but there are already ongoing efforts focusing on extracting taxonomy from the category graph [12]. Wikipedia contains only implicit semantic relationships encoded by hypertext references, infoboxes, and templates. Explicit semantics could be expressed using SemanticWiki [13], yet there is no simple and automatic way to convert Wikipedia into proper SemanticWiki.

The ontology-based categorization method proposed in this paper can be properly adjusted to use a Wikipedia-based ontology and successfully categorize documents using it. We decided to rely on the already existing implicit semantics and adjust the category selection for a thesaurus-like structure instead of relying on a properly defined taxonomy. Note, that in the remainder of this paper, we will refer to our Wikipedia-based ontology simply as Wikipedia.

A. Motivating Example

Let us present a fragment of a recent news article to illustrate the process of ontology-based categorization:

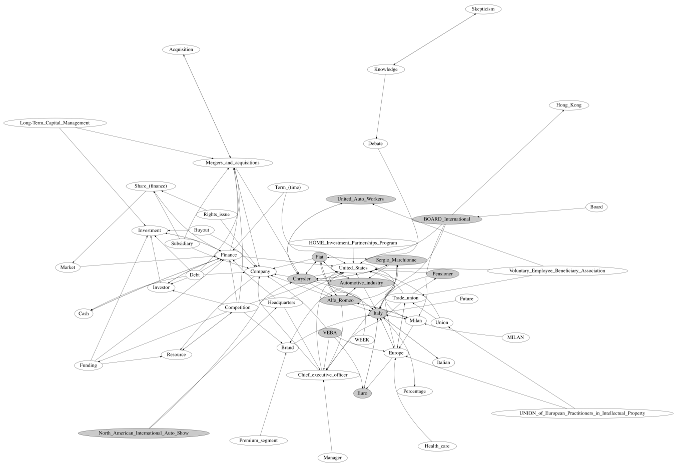
> **Fiat** has **completed** its **buyout** of **Chrysler**, making the **U.S.** business a wholly-owned **subsidiary** of the **Italian carmaker** as it **gears** up to **use** their **combined resources** to **turn around** its loss-making **operations** in **Europe**. The **company** **announced** on January 1 that it had **struck** a $4.35 **billion** **deal** - cheaper than analysts had **expected** - to **gain** full **control** of **Chrysler**, **ending** more than a year of **tense** talks that had obstructed **Chief Executive** **Sergio Marchionne**'s efforts to create the world's seventh-largest **auto maker** […]
>
> **Marchionne** said at the **Detroit car show** last **week** that a **listing** of the **combined** **entity** was on the **agenda** for this year. While **New York** is the most **liquid** **market**, **Hong Kong** is also an option, the CEO said, pledging to **stay** at the **helm** of the **merged** **group** for at least three years.
>
> The first big **test** for the **merged** **Fiat**-**Chrysler** will be a three-year **industrial** **plan** **Marchionne** is expected to unveil in May, in which he will **outline** **planned** **investments** and **models**. […]
>
> **Fiat** has said its **new** **strategy** will **focus** on revamping its **Alfa Romeo** brand and **keeping** **production** of the **sporty** **marque** in **Italy** as it **seeks** to utilize **plants** operating below **capacity**, protect **jobs** and **compete** in the higher-margin **premium segment** of the **market**.

> **Shares** in **Fiat** were up 1.77 **percent** at 7.46 **euros** by 1630 **GMT**, outperforming a 0.11 **percent** **rise** for **Milan**'s **blue-chip** **index**.

The categorization process of the above article could be performed as follows. First, we could identify the entities (underlined) and using the information from the Wikipedia ontology, induce relationships among them. This would lead to the creation an initial semantic graph of connected entities that were recognized in the document. Note that several matched entities do not belong to the main subject of the document and even some can be matched ambiguously (initial entity recognition is based on matching their labels occurring in the text). Disambiguation issues are addressed in the subsequent analysis of the graph.

Distinct connected components in the semantic graph are associated with different domains of interest, as we assume that entities from the same domain are closely connected. The selection of the most important component (a sub-graph of the semantic graph), based on its size and weights of the included entities, not only effectively eliminates entities assigned to different domain(s), but also helps to disambiguate multiple entities that were matched by the same phrase in the text by choosing a specific interpretation context. The entities included in the most important component, called the thematic graph, form the basis for the categorization of the article. The central fragment of the dominant thematic graph created for the above example is presented in Figure 1.

Fig. 1. Thematic graph from the example text



Even after the elimination of less important connected components, the thematic graph may contain multiple sub-domains and/or different interpretations of entities and relationships among them. In order to establish the focus of the categorization process, we choose the core entities in the thematic graph (shaded entities in Figure 1). These are the entities discovered as (1) the best hubs and authorities by the HITS algorithm [14], (2) the best entities describing the graph taking into account global information recursively computed from the entire graph by using TextRank algorithm [15] as well as (3) the most central entities in the analyzed thematic graph. They are the starting points of the categorization.

The categorization of the thematic graph to classification topics, defined as compositions of ontology contexts, requires measuring the semantic similarity between the supplied context definition and the intersection of the context projection and the thematic graph. This similarity measure shows how close the thematic graph is to the selected context (topic).

Continuing with the example article, our categorization could assign a number of most likely Wikipedia categories to the analyzed document. The top 5 assigned categories are:

- Stock Market
- Automotive Industry
- Debating
- Corporate Finance
- Legal Entities

III. Classification Categories

Our text classification method allows dynamic specification of classification categories. In this section, we present our notion of an ontology context, defined as a projection over ontology classes and instances (entities), and how ontology contexts and their compositions can be used as classification topics (categories).

A. Context as an ontology sub-graph

A significant amount of work has been done on the subject of context and its formal specification [16] [17]. Our definition of a categorization context is to some extent based on the previous research on views in semi-structured databases [18] and in ontologies [19]. While context and its relationship to the ontology can be presented in many ways, we will define it in terms of an RDF/RDFS ontology. In the following definitions, R represents an RDF description base, while S an associated RDFS schema.

Def. 1. The *hierarchical distance* between an instance entity *e* from a description base R and a class *c* from an RDFS schema S, denoted as $dist_H(e,c)$, is defined as the length of the shortest path formed by one rdf:type and zero or more rdfs:subClassOf properties connecting *e* and *c*. In case the entity *e* is not an instance of class *c* (directly or via the rdfs:subClassOf properties), $dist_H(e,c)$ is set to 0.

By extension of Def. 1, the hierarchical distance between an instance entity *e* and a set of classes *C*, denoted as $dist_H(e,C)$, is defined as the minimum, positive value among all $dist_H(e,c)$, where $c \in C$. If *e* is not an instance of any of the classes in *C* (directly or via the rdfs:subClassOf properties), $dist_H(e,C)$ is set to 0.

Def. 2. Let C be a set of schema classes included in an RDFS schema S. A *projection* of classes C onto an RDF description base R is a set of instance entities in R paired with their corresponding hierarchical distances to C, defined as:

$$\Pi(C,R) = \{\ e(k):\ e \in \mathrm{R} \wedge \mathrm{k} = \mathrm{dist_H}(e,C) \wedge \mathrm{k} > 0\ \}.$$

Def. 3. A *categorization context* defined by a set of schema classes C is a projection of C onto an RDF description base. A categorization context will be also called a *classification topic (category)*.

Def. 4. Given two categorization contexts m_1 and m_2, the following *context expressions* are also categorization contexts:

- $m_1 \cap m_2$ – (intersection of contexts $\{ e(k)$: $e(k_1) \in m_1 \wedge e(k_2) \in m_2 \wedge k = \min(k_1,k_2) \}$)
- $m_1 \cup m_2$ – (union of contexts $\{ e(k)$: $(e(k_1) \in m_1 \vee e(k_2) \in m_2) \wedge k = \min(k_1,k_2) \}$)
- $m_1 \setminus m_2$ – (difference of contexts $\{ e(k)$: $e(k) \in m_1 \wedge \forall k_2 > 0$: $e(k_2) \notin m_2 \}$).

We say that an instance entity *e,* which is a member of a categorization context, is *covered* by the context. A simple illustration of the coverage of instance entities is shown in Fig. 2. In the remainder of this paper, we will regard the terms *classification topic* and *categorization context* as synonymous and refer to a classification topic (or a categorization context) defined by a set of schema classes C as a classification topic (or categorization context) C.

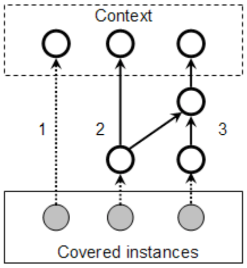


Fig. 2. Categorization context, hierarchical distance and covered entities

B. *Composition of contexts*

Topic definitions based on ontology context projections may not offer sufficient flexibility in defining classification topics. More specifically, a classification topic should capture user's interest in a specific area or even a *combination* of topics. In order to further enhance the specification of classification topics, we extend the definition of a classification topic to include a linear combination of a number of selected categorization contexts.

A combination of categorization contexts enables us to the use vector space model for calculating the categorization results and gives the user much greater flexibility and precision in defining a category of interest. Expressing intersection, union or difference of entity sets obtained by context projections can be done using the provided context expressions (Def. 4). The use of a linear combination of contexts enables us to define categories involving multiple contexts, but which cannot be expressed as an intersection, union or difference of these contexts.

As an example, consider contexts defining "business" and "sports". Different topics represented as a combination of the two contexts are presented in Figure 3. Topic (A), defined as a union of the two contexts, will match documents that belong to "business", "sports" or both. Topic (B), defined as an intersection of the two contexts, will match documents with entities that belong at the same time both to "business" and "sports". Using only context expressions introduced in Def. 4, we are not able to specify a topic of documents that fall into *both* contexts, meaning that the document belongs to "business" and to the "sports" category (for example, business activities of football teams, or a football league). Such documents must include entities both from "business" context (area 1) and "sports" context (area 2), but not necessarily entities from their intersection. Intuitively, we name such documents as belonging to the intersection of "sports" and "business", but at the instance level such topic (C) should be defined as a linear combination of both contexts. It must contain entities from each of the included contexts, whereas the union of contexts is too wide and the intersection too narrow. Even using symmetric difference of contexts still does not guarantee that entities from both contexts will be represented in a document graph. The following is an extension of Def. 4.

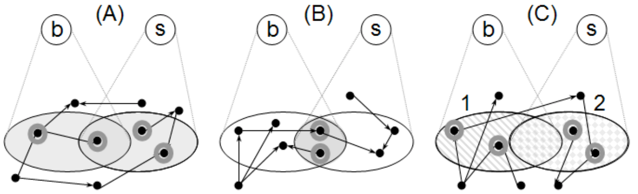


Fig.3. Instanced graph with selected matched entities for topics defined as union, intersection, and a combination of contexts.

Def. 5. Let C_i, $1 \le i \le n$, be categorization contexts (classification topics). A *composition of categorization contexts* is defined as vector of pairs (C_i, a_i), $1 \le i \le n$, where the coefficients a_i are normalized and indicate relative importance of the contexts C_i in the composition.

IV. CATEGORIZATION ALGORITHM

As discussed earlier, we have converted Wikipedia to an RDF ontology using a modified DBpedia [20] tool. DBpedia offers a tool for converting Wikipedia into RDF/S [21] format. We used it with some minor adjustments to facilitate a better and more precise discovery of named entities in the document. Literals (name, alias, redirection or disambiguation names) associated with an entity are the key information for the matching process. Each of the literal types has a different confidence in unambiguous identification of the entity. We associated such literals with each entity using specific relations to distinguish their confidence level during the matching process.

Our categorization algorithm consists of three main steps: (1) construction of the semantic graph, (2) selection and analysis of the thematic graph, and (3) categorization of the selected thematic graph. We have modified and extended our previously presented categorization algorithm [5, 6] with dynamically defined classification topics. The categorization topics are defined as ontology contexts, introduced in the previous section. They can be perceived as *ontology views* that specify user's contexts of interest. Classification topics are defined *dynamically* and *independently* from the document corpora.

A. Semantic graph construction

Document's semantic graph is constructed from the named entities identified in the document. We assume that each entity in the ontology has one or more literal properties associated with it (such as name and synonym) that can be used for matching purposes. For each literal associated with the entity we assign a confidence level that reflects how uniquely it can identify the entity. Note, that one literal can be associated with multiple entities and produce ambiguous entity matches. Disambiguation in such cases takes place in the later phase of the categorization. In what follows, we will treat documents as sequences of words and sub-sequences of words as phrases (a single word is regarded as a phrase, as well).

Def. 6. Given a document d, the *entity matching function*, $E(d)$, returns a set of ontology entities e, such that for each e in $E(d)$, there exists a phrase in d matching one of e's identifying labels. Each entity in $E(d)$ is assigned a weight $w(e)$, given by the formula:

$$w(e) = 1 - \frac{1}{1 + \sum_{i=1..n} p_i * s(l_i, sp_i)} \qquad (1)$$

where n is the number of occurrences in d of a phrase matching e, p_i is the confidence of the relationship (property) used for entity identification and $s(l_i,sp_i)$ is the similarity of the spotted phrase sp_i and the entity's identifying literal l_i.

It should be noted that the function s measures the similarity between the spotted phrase sp in document d and the entity e's label (literal) l in the ontology, taking into account the removed stop words and/or stemming.

Def. 7. Two entities e_i, e_j in $E(d)$ are *ambiguous* if their corresponding matched phrases p_i, p_j overlap in document d, i.e. the two phrases have words in common, or are identical.

Def. 8. A *semantic graph* of a document d, denoted $SG(d)$, is a labeled graph with a set of vertices $E(d)$ and a set of labeled edges $\{(e_i, e_j)$ with label *r, such that* $e_i, e_j \in E(d)$ and e_i, and e_j are connected by a relationship (property) r in the ontology$\}$.

Even though the ontology relationships induced in $SG(d)$ are directed, from now on, we will consider $SG(d)$ as an undirected graph.

Since the semantic graph of a document is created by forming associations among the identified entities based on the properties existing in the ontology, it can be seen as adding the background knowledge to the document in order to explain the associations between the entities. Each property type (edge label) included in the semantic graph has some notion of information, which we interpret as the importance of the relationship between the entities.

B. Thematic graph selection

The selection of the thematic graph combines the steps of feature selection and entity match disambiguation. It is based on the assumption that entities related to a single topic are closely associated in the ontology, while entities from different topics are placed far apart, or even not connected at all. As a result, the analyzed semantic graph may be composed of multiple connected components, as each set of connected entities represents a different topic recognized in the document.

Def. 9. A sub-graph of $SG(d)$ is called an *interpretation of a document d*, denoted $I(d)$, if the sub-graph does not contain any ambiguous entities.

Def. 10. A connected component of $I(d)$ is called a *thematic sub-graph*. In particular, if the whole $I(d)$ is a connected graph, it is also a thematic sub-graph.

In general, an interpretation of a document may have many thematic sub-graphs, one for each of its connected components. The importance of entities in a thematic graph of a document is determined not only by their initial weights but also by their placement in the graph. We utilize the HITS algorithm with the assigned initial weights for both entities and relationships to find the authoritative entities in the semantic graph.

Def. 11. A thematic sub-graph with the largest number of nodes and the highest total of entity weights is selected as the *dominant thematic graph* for the document.

Selecting a dominant thematic graph sets a specific interpretation context and effectively *disambiguates* any incorrectly matched entities. Furthermore, we locate central entities in the graph (based on the geographical centrality measure), since they can be identified as the thematic landmarks of the graph. We also use TextRank algorithm to find the best entities describing the graph [15]. The most authoritative, descriptive and central entities are later used as the basis for document classification.

Def. 12. The *core of the dominant thematic graph* is composed of k most authoritative, descriptive and most central entities.

From now on, we will simply write thematic graph when referring to the dominant thematic graph of a document.

C. Classification into defined ontological contexts

Classification of a document into the defined ontological contexts (topics) is based on calculating a similarity of the document's thematic graph to each of the defined contexts. The similarity of the thematic graph to a given context represents how well the semantic interpretation of the document fits within the selected ontological context. In general, the similarity is calculated based on the following objectives:

- The intersection of the context projection with the thematic graph should be maximized (coverage).
- The hierarchical distance of the entities in the thematic graph to the classes included in the context should be minimized (closeness).
- The highest number of the core entities should be covered and close (in hierarchical distance) to the context.

To establish the similarity of the document's d thematic graph to each of the defined contexts c_1, …, c_n (topics), we perform the following steps:

1. Find the *expanded core entities* in d, which include the core entities in the thematic graph of d and all of their immediate neighbors.

2. Construct a *taxonomy* graph out of the Wikipedia categories network for the expanded core entities. That is, identify the categories of the expanded core entities and compute the hierarchy of their ancestors. It should be noted that we empirically restrict the hierarchy's height to 3, due to the fact that increasing the height further quickly leads to excessively general categories.
3. Compute the *semantic associativity score* of the categories located in step 2 to the expanded core entities.
4. Find the top-*k* categories based on their score as the best categories describing the document.
5. For every defined context (user defined topic)[1], execute Algorithm 1 and return the semantic relatedness score of document *d* to the defined context.

Steps 3 and 5 above are described in greater detail in sections D and E, respectively.

D. Computing category semantic associativity score

To compute the *semantic associativity* of a document to a categorization context, we first calculate the *membership score* and the *coverage score.* We have adopted a modified Vector-based Vector Generation method (VVG) described in [22] to calculate the category *membership score.* Given Wikipedia as a directed graph G = {W, V, E} and a Wikipedia concept w_i and category v_j, the *membership score mScore*(w_i,v_j) of concept w_i to category v_j is defined as follows:

$$mScore(w_i,v_j) = \prod_{e_k \in E_l} m(e_k) \tag{2}$$

$$m(e_k) = \frac{1}{n} \tag{3}$$

where $m(e_k)$ is the weight of membership links (category links), e_k, from node v_i (or w_i) to category $v \in$ V, n is the number of membership links, and $E_l = \{e_1, e_2, ..., e_m\}$ represents a set of all membership links forming the shortest path p from the concept w_i to category v_j.

The coverage score *cScore*(*c*,*e*) of an entity *e* by a Wikipedia category *c* is computed by the following formula:

$$cScore(c,e) = \begin{cases} 1 & \textit{if there is a path between c and e} \\ 0 & \textit{otherwise} \end{cases}$$

The *semantic associativity* score between a category and a set of entities is defined as follows:

$$cSAssScore(c,E_{ee}) = \beta * \sum_{e \in E_{ee}} mScore(c,e) + (1-\beta) * \sum_{e \in E_{ee}} cScore(c,e) \tag{4}$$

where E_{ee}={*expanded core entities*}, *c* is the Wikipedia category and β is the smoothing factor to control the influence weight of two scores. We used β =0.8 in our experiments.

E. Document categorization score to ontological context

To find the *categorization score* of a document to an ontological context (topic), we start by measuring the semantic relatedness among Wikipedia entities (concepts). In order to do that, we adopted the Wikipedia Link-based Measure (WLM) introduced in [23]. Given two Wikipedia entities *a* and *b*, we define the *semantic relatedness* between them as follows:

$$sr(a,b) = \frac{log(max(|A|,|B|)) - log(|A \cap B|)}{log(|W|) - log(min(|A|,|B|))} \tag{5}$$

where *A* and *B* are the sets of Wikipedia entities that link to *a* and *b* respectively and *W* is the set of all entities in Wikipedia. By extension, the *semantic relatedness* between a Wikipedia entity *a* and a categorization context *C* (a categorization context is a projection of C onto the background ontology, including entities $\{e_1, e_2, ..., e_t\}$) is defined as follows:

$$sr(a,C) = \frac{1}{t}\sum_{i=1}^{i=t} sr(a,e_i) \tag{6}$$

Algorithm 1- *computeSemRelatedness(d,C)*

```
Input: $d$ is a document, $c_1, \ldots, c_n$ is a set of categorization
       contexts (topics), and $t$ is a threshold
For each $c_i$, $1 \le i \le n$:
    Find the intersection $S$ of the top-$k$ categories of $d$ and $c_i$
    For each category $c_p$ in $S$
        Find all the entities belonging to $c_p$, denoted as $E_{c_p}$
        Find the maximum of $sr(e_j, E_{c_p})$ , $e_j \in E_{ee}$
    Sort the maximums in descending order and compute, the
    average of the top-$k$ of them, denoted as $\zeta_i(c_i)$
    If $\zeta_i(c_i) < t$ , set $\zeta_i(c_i)$ to 0.
Return $\zeta_1(c_1) + \zeta_2(c_2) + \cdots + \zeta_n(c_n)$
```

If $\zeta_i(c_i) \ge t$, we conclude that document *d* belongs to topic c_i. The threshold ***t*** is established empirically.

Note, that in case a topic is defined as a composition of contexts, we determine the final score of a document based on the set operators used in the composition as follows:

- $c_i \cap c_j$: ζ = *min*(ζ$_i$(c$_i$), ζ$_j$(c$_j$))
- $c_i \cup c_j$: ζ = *max* (ζ$_i$(c$_i$), ζ$_j$(c$_j$))
- !c_i: ζ = 1- ζ$_i$(c$_i$)

V. Experiments

In our experiments, we used an RDF ontology created from the full version of English Wikipedia XML dump from 2013-06-04. The created ontology contained 5,047,075 entities connected by 287,016,171 statements and 13,062,411 literals describing the entities. They were classified using 930,472 categories defined in Wikipedia. We used Virtuoso[2] for ontology storage (triple store) and querying.

[1] "Ontology context", "user defined category" and "topic" are interchangeable.

[2] http://virtuoso.openlinksw.com/dataspace/doc/dav/wiki/Main/

We evaluated our system on a text corpus obtained from the Reuters[3] RSS feed (2013-10-24 – 2014-01-30). We divided some of the main topics into fine-grained sub-topics in order to evaluate our classification method. The details of the text corpus, as well as the fine-grained categorization of the main topics are presented in Table 1 and Table 2, respectively.

Reuters Category	Number of Documents
Sports	254
Technology	927
Business	786
Arts	94
Science	140
Health	864
Politics	807
Total	**3,872**

Table 1. Category details of used text corpus

Main Topics	Sub-topics
Sports	• Baseball • Basketball • National_Hockey-League • Tennis • Golf • National_Football_League • Football (Soccer)
Technology	• Digital_Technology • Space_Technology • Mobile_Technology • Telecommunications
Business	• Economics • Industry • Financial_Markets

Table 2. Fine-grained categorization of main categories

Due to the nature of the analyzed news documents, we decided to exclude time related entities since they provided highly misleading connections among other entities from the categorization process.

A. Experiment results

We conducted three experiments on the Reuters corpus. In the first experiment, we wanted to assess the basic topic categorization of our system. Here, we created categorization contexts consisting of high-level Wikipedia categories to represent the topics best corresponding to those in the Reuters corpus. The defined contexts included Wikipedia categories with names directly corresponding to the Reuters' category names. Table 3 shows the micro averaged precision (MAP) of the first performed experiment.

TOPICS	Micro Averaged Precision (MAP)
Arts	96.8%
Science	92.1%
Health	90.6%
Politics	95.7%
Total	**93.8%**

Table 3. Categorization result on Reuters with high–level ontological contexts

In the second experiment, we wanted to evaluate the effectiveness of categorizing into topics composed of *unions of contexts*. Therefore we created topic contexts for Sports, Technology and Business as the unions of their sub-topics, according to Table 2, (e.g. Business = Economics ∪ Industry ∪ Financial_markets). Hence, we not only identified the high-level topics of documents, but also recognized the specific sub-topics within them. The results are presented in Table 4.

TOPICS	Micro Averaged Precision (MAP)
Sports	97.8%
Technology	85.6%
Business	79.4%
Total	**87.6%**

Table 4. Categorization result on Reuters with high–level ontological contexts

In the third experiment, we wanted to assess our system's ability to categorize documents into topics expressed as more complex context compositions. Consequently, we created *compositions of contexts* from the "technology", "business" and "politics" topics. The topics were defined as follows:

- (*Digital* ∩ *Telecom*) ∩ (!*Mobile*),
  i.e. documents that belong to "digital_technology" and "telecommunications" topics, but not to "mobile_technology" topic.
- *Economics* ∩ (!*Financial_Markets*)
- !(*Economics* ∪ *Industry* ∪ *Financial_Markets*)
- *Politics* ∩ (!*Immigration*)

We chose random samples of 94, 68 and 236 documents from "technology", "business" and "politics" topics respectively and ran our classification system on them. Table 5 represents the details of the third experiment.

TOPICS	Micro Averaged Precision (MAP)
(*Digital* ∩*Telecom*)∩ (!*Mobile*)	86.7%
Economics ∩ (!*Financial_Markets*)	80%
!(*Economics*∪*Industry*∪*Financial_Markets*)	100%
Politics ∩ (!*Immigration*)	90.4
Total	**89.3%**

Table 5. Categorization result for composition of topics

B. Results analysis

Our ontology-based categorization method achieved very good results. These results are especially promising in view of the fact that our method did not rely on classifier training and that it can be readily applied to any other set of topics defined as classification contexts or their compositions.

The analysis of the incorrectly classified documents and the created semantic graphs revealed the following clues for possible causes of misclassifications:

- The prepared categorization contexts used the Wikipedia category hierarchy, which not always reflects the topics covered in news.
- In some cases, highly connected domains in Wikipedia favored a context different than the major one described in the document. This was caused by the imbalance between densely and sparsely populated domains in Wikipedia.
- There is a disagreement between the category best describing the majority of document content, and the

[3] http://www.reuters.com/

originally assigned category based on user's perceived interest. For example, a document about a famous politician undergoing a complex medical procedure could be classified as politics, even though the majority of the document talks about the politician's health issues.

- In some cases, although the categories are entirely different, one is a sub-category of the other. For example, American football and football (Soccer) are two different sports, but the former is a subcategory of the latter one in Wikipedia, which results in categorizing the American football documents into football (Soccer) category, as well.

Despite the imprecise coverage of the news topics by the Wikipedia-based created categorization contexts and the imbalance in the coverage of some domains, our ontology-based training-less categorization method was able to achieve comparable results to the traditional categorization methods. We intend to conduct a thorough evaluation of our categorization method and compare it with traditional methods, e.g. Naïve Bayes, SVM, etc. However, such evaluation will be difficult, since a direct comparison to traditional categorization methods will require a document corpora containing both the training sets and the ontological definitions of the topics used for categorization.

Categorization using the prepared contexts provides a view of documents through specific *interest lenses*. An important aspect of the proposed method is that with the change of user's topics of interest, the classification contexts can be quickly redefined. Consequently, the same documents can be categorized into newly defined contexts without the need for a new set of training documents and classifier re-training.

Conclusion and Future Work

In this paper, we presented a novel approach to text categorization, where the categorization relies only on the ontological knowledge and classifier training is not required. Categories of interest can be easily defined as context projections or their combinations. We have defined categorization contexts in the ontology together with operators allowing their manipulation. The performed experiments proved the applicability of ontologies for automatic text categorization. They also demonstrated a significant value of knowledge represented in Wikipedia as applied to text categorization purposes. In the near future, we intend to conduct additional testing of our method, especially involving the combination of classification contexts.

Multiple language versions of Wikipedia open new possibilities for ontology-based categorization. The presented method relies on entities, relationships, and the assigned categories. The structure of Wikipedia is similar across languages in that it has defined entities, their names (labels), and assigned categories. Once the entities in a document are recognized as belonging to a specific language, a proper language version of Wikipedia can directly be used for categorization. Moreover, using the Wikipedia encoded mapping of entities and categories across different languages, we can transform a thematic graph to a different language in a straightforward way. This enables the presentation of the main domain of the document, as well as its final categorization in a language different than the original language of the document.

References

[1] F. Sebastiani, "Machine learning in automated text categorization," ACM computing surveys (CSUR), vol. 34, pp. 1-47, 2002.

[2] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," Discourse processes, vol. 25, pp. 259-284, 1998.

[3] E. Gabrilovich and S. Markovitch, "Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge," in AAAI, 2006, pp. 1301-1306.

[4] P. Schönhofen, "Identifying document topics using the Wikipedia category network," Web Intelligence and Agent Systems, vol. 7, pp. 195-207, 2009.

[5] M. G. Janik, "Training-less ontology-based text categorization," 2008.

[6] M. Janik and K. J. Kochut, "Wikipedia in action: Ontological knowledge in text categorization," in Semantic Computing, 2008 IEEE International Conference on, 2008, pp. 268-275.

[7] Z. S. Syed, T. Finin, and A. Joshi, "Wikipedia as an Ontology for Describing Documents," in ICWSM, 2008.

[8] M. Strube and S. P. Ponzetto, "WikiRelate! Computing semantic relatedness using Wikipedia," in AAAI, 2006, pp. 1419-1424.

[9] E. Gabrilovich and S. Markovitch, "Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis," in IJCAI, 2007, pp. 1606-1611.

[10] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, et al., "DBpedia-A crystallization point for the Web of Data," Web Semantics: Science, Services and Agents on the World Wide Web, vol. 7, pp. 154-165, 2009.

[11] J. Voss, "Collaborative thesaurus tagging the Wikipedia way," arXiv preprint cs/0604036, 2006.

[12] S. P. Ponzetto and M. Strube, "Deriving a large scale taxonomy from Wikipedia," in AAAI, 2007, pp. 1440-1445.

[13] M. Völkel, M. Krötzsch, D. Vrandecic, H. Haller, and R. Studer, "Semantic wikipedia," in Proceedings of the 15th international conference on World Wide Web, 2006, pp. 585-594.

[14] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," Journal of the ACM (JACM), vol. 46, pp. 604-632, 1999.

[15] R. Mihalcea and P. Tarau, "TextRank: Bringing order into texts," in Proceedings of EMNLP, 2004, p. 275.

[16] J. McCarthy, "Notes on formalizing context," 1993.

[17] R. Guha, R. McCool, and R. Fikes, "Contexts for the semantic web," in The Semantic Web–ISWC 2004, ed: Springer, 2004, pp. 32-46.

[18] S. Abiteboul, R. Goldman, J. McHugh, V. Vassalos, and Y. Zhuge, "Views for semistructured data," 1997.

[19] S. Decker, M. Sintek, and W. Nejdl, "The modeltheoretic semantics of TRIPLE."

[20] S. Auer and J. Lehmann, "What have innsbruck and leipzig in common? extracting semantics from wiki content," in The Semantic Web: Research and Applications, ed: Springer, 2007, pp. 503-517.

[21] D. Brickley and R. V. Guha, "{RDF vocabulary description language 1.0: RDF schema}," 2004.

[22] M. Shirakawa, K. Nakayama, T. Hara, and S. Nishio, "Concept vector extraction from Wikipedia category network," in Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication, 2009, pp. 71-79.

[23] I. Witten and D. Milne, "An effective, low-cost measure of semantic relatedness obtained from Wikipedia links," in Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy, AAAI Press, Chicago, USA, 2008, pp. 25-30.