

Training-less Ontology-based Text Categorization

Maciej Janik and Krys Kochut

Large Scale Distributed Information Systems Lab (LSDIS)
Department of Computer Science, University of Georgia
410 Boyd Graduate Studies Research Center, Athens, GA 30602-7404
{janik, kochut}@cs.uga.edu

Abstract

We present a new, ontology-based approach to the automatic text categorization. An important and novel aspect of this approach is that our categorization method does not require a training set, which is in contrast to the traditional statistical and probabilistic methods that require a set of pre-classified documents in order to train the classifier.

In our approach, the ontology, which holds the schema, including the domain entities organized into categories and interconnected by relationships, as well as instances and linkages among them, effectively becomes the classifier for the categories of the domain concepts. After a document is converted into a thematic graph of entities, the ontological classification of the entities in the graph is then analyzed in order to determine the overall categorization of the thematic graph, and as a result, of the document.

In presented experiments, we used an RDF ontology constructed from the full English version of Wikipedia, a Web-based encyclopedia. The experiments, conducted on a collection of news articles, show that our training-less categorization method has achieved a satisfactory overall accuracy, in one experiment nearly identical to a selected traditional categorization method.

1. Introduction

Automatic text categorization is a task of assigning one or more pre-specified categories to an electronic document, based on its content. Nowadays, text classification is extensively used in many contexts. One of the examples is the automatic classification of incoming electronic news into categories, such as entertainment, politics, business, sports, etc. Standard categorization approaches utilize statistical or machine learning methods to perform the task. Such methods include Naïve Bayes [14], Support Vector Machines [27], Latent Semantic Analysis [7] and many others. A good overview of the traditional text categorization methods is presented in [22]. All of these methods

require a training set of pre-classified documents that is used for classifier training; later, the classifier can correctly assign categories to other, previously unseen documents.

However, it is often the case that a suitable set of well categorized (typically by humans) training documents is not available. Even if one is available, the set may be too small, or a significant portion of the documents in the training set may not have been classified properly. This creates a serious limitation for the usefulness of the traditional text categorization methods.

As described by the World Wide Web Consortium (W3C), ontology defines the terms used to describe and represent an area of knowledge. Ontologies are used by people, databases, and applications that need to share domain information (a domain is just a specific subject area of knowledge, such as medicine, real estate, automobile repair, or financial management). More specifically, ontology is a data model that represents a set of concepts (entities) within a given domain and the relationships between those concepts. It is used to reason about the concepts within that domain.

In this paper, we introduce a novel text categorization method based on leveraging the existing knowledge represented in a domain ontology. The novelty of this approach is that it is not dependent on the existence of a training set, as it relies solely on the entities, their relationships, and the taxonomy of categories represented in the ontology.

In the proposed approach, the ontology effectively becomes the classifier. Consequently, classifier training with a set of pre-classified documents is not needed, as the ontology already includes all important facts.

The proposed approach requires a transformation of the document text into a graph structure, which employs entity matching and relationship identification. The categorization is based on measuring the semantic similarity between the created graph and categories defined in the ontology.

2. Motivation

An *ontology* is defined as “an explicit specification of a conceptualization” [10]. An ontology created for a given domain includes a set of concepts as well as relationships connecting them within the domain. Collectively, the concepts and the relationships form a foundation for *reasoning* about the domain.

Within the area of computing, the ontological concepts are frequently regarded as *classes* which are organized into hierarchies. The classes define the types of *attributes*, or properties common to individual objects within the class. Moreover, classes are interconnected by *relationships*, indicating their semantic interdependence (relationships are also regarded as attributes) [24]. Class hierarchies and class relationships form the *schema level* of the ontology, while the individuals (object instances or just instances) and links among them (relationship instances) form the so called *ground level* of the ontology. RDF/S [5] and OWL [18] are two examples of popular ontology specification languages.

A comprehensive, well populated ontology with classes and relationships closely modeling a specific domain represents a vast compendium of knowledge in the domain. It is only natural to expect that having such a comprehensive knowledge about the domain, one should be well-equipped to create software systems implementing a variety of tasks concerning the domain of the ontology. Recently, ontologies have been used in various semantic applications, ranging from business analytics [23] to semantic data integration [6].

We believe that the knowledge represented in such a comprehensive ontology can be used to identify topics (concepts) in a text document, provided the document thematically belongs to the domain represented in the ontology. Furthermore, if the concepts in the ontology are organized into hierarchies of higher-level categories, it should be possible to identify the category (or a few categories) that best classify the content of the document.

As an example, let us assume that we have a well-defined and comprehensive ontology containing knowledge about a variety of disciplines of sports in the United States, including baseball, (American) football, basketball, golf, and others. We will assume that the ontology includes a wide variety of concepts of each sport, such as a *home run*, *pitch*, *inning*, *hitter*, *quarterback*, *touchdown*, and so on, relationships, specifying that a baseball game is *composed of* innings, a punt is an *element of* the game of football and so is the *free throw* of the game of basketball. Furthermore, let us assume that the ontology contains all the relevant instances, such as sports teams (Boston Red Sox,

Cleveland Indians), players, as well as the coaching and managing staffs of each of the teams, and links among them, for example specifying that one of the infielders for the Red Sox is Dustin Pedroia, and that Rafael Betancourt plays as pitcher with the Cleveland Indians. We will also assume that our ontology classes are organized into a hierarchy of higher level classes that group our concepts and instances into a number of broad categories, such as Major League Baseball, Baseball leagues, Baseball, and so on.

Now, consider a news article describing a typical baseball game. We believe that such an article most likely contains a number of occurrences of concepts and/or individuals represented in our ontology (such occurrences are known as *named entity occurrences*). There may be many clues that the document is, in fact, about baseball. First, we may be able to identify several named entity occurrences in the text of the document. For example, in the following document fragment:

In the seventh inning, Red Sox rookie second baseman Dustin Pedroia hit a two-run home run off of Rafael Betancourt that drove Boston's Fenway Park wild. Boston scored a total of 6 runs in a crazy eighth inning, on a single by J.D. Drew, a three-run double by Pedroia, and a two-run Kevin Youkilis home run which bounced off a large Coke bottle advertisement. Betancourt gave up the first four runs, with the home run allowed by Jenson Lewis. Youkilis was the first batter he faced in relief of Betancourt.

we could identify several named entities (represented here by underlined words and phrases). In addition, on the basis of our ontology, we could establish a number of relationships among the identified entities, such as the facts that Dustin Pedroia is a second baseman for Red Sox, and Kevin Youkilis is a batter for the same team. Also, we could discover that the entities (concepts) home run and inning are associated with baseball and Red Sox in an American baseball team.

Based on the above information, we could construct a *semantic graph* represented by the entities in the document and the relationships identified in the ontology. The semantic graph, which represents the thematic content of the document, could then be used to determine the document's category, or perhaps identify its topics.

Note, that the same phrase (or a single word) may identify a number of distinct entities in the ontology. For example, Betancourt (last name), recognized in the above text, may refer to the Rafael Betancourt playing for Boston Red Sox, Cuban baseball player Danny Betancourt, or perhaps Agustín de Betancourt, a structural engineer and educator from nineteenth century. It is even possible that some phrases used in the document would match many completely different

entities, not belonging to the same domain (or sub-domain). As a consequence, we could create more than one semantic graph for the entities identified in the document. Each semantic graph would offer a plausible, different *interpretation* of the thematic content of the document. If so, a semantic graph which would present the best fit with the ontology would be selected as the *dominant semantic graph*.

Finally, the dominant semantic graph could be used to establish the overall category of the document, in that we might be able to identify one, or perhaps a small number of categories that classify all, or most of the entities and relationships in the dominant semantic graph.

Another important observation is that in one of the sentences in the above text: “In the seventh inning, Red Sox rookie second baseman Dustin Pedroia hit a two-run home run off of Rafael Betancourt that drove Boston’s Fenway Park wild,” we could identify not only the named entities, but we might even be able to recognize the direct relationship “is second baseman of” between Red Sox and Dustin Pedroia. Such relationships recognized directly in the document, perhaps with the use of Natural Language Processing (NLP), could be used to strengthen the degree to which the semantic graph fits within our ontology.

We are approaching a time when comprehensive ontologies will be available for numerous domains. As of today, several interesting ontologies have been created in the area of biology [3], medicine [8] and culture [17]. Work is in progress on creating an ontology based on Wikipedia¹, Web encyclopedia. An RDF version of Wikipedia described in [2] is an interesting intermediate step towards this goal. The information found in such a Wikipedia-based ontology can be regarded as a source of comprehensive encyclopedic knowledge on just about any domain, ready for supporting semantics-based applications. We believe that automatic, training-less text categorization is an important example of such applications.

3. Related work

External or background knowledge can significantly improve text categorization, especially for short or ambiguous documents. It helps to unify the vocabulary, match important phrases, strengthen co-occurrences, or use related information not included in the original document in order to perform document categorization.

One of the best known sources of external knowledge is WordNet[1] – a network of related words, that can be used to match similar words and treat them as the

same in classification process. One possible approach of utilizing WordNet in text classification is described in [20].

Ontologies offer knowledge that is organized in a more structural and semantic way. Their use in text categorization and topic identification has lately become an intensive research topic. As ontologies provide named entities and relationship between them, an intermediate categorization step requires matching terms to ontological entities. Afterwards, an ontology can be successfully used for term disambiguating and vocabulary unification, as presented in [4]. Another approach, presented in [16], reinforces co-occurrence of certain pairs of words or entities in the term vector that are related in the ontology. The use of descriptions of neighboring entities to enrich the information about a classified document is described in [9]. Interesting approach, although very different, is presented in [29], where authors automatically build partial ontology from the training set to improve keyword-based categorization method. Other categorization approaches based on using recognized named entities are described in [25] and [11].

Initial work has been done lately in using Wikipedia for categorization purposes. These approaches utilized the fact that Wikipedia contains a vast amount of knowledge that is interconnected and categorized. Pages in Wikipedia can be treated as named entities and categories form a kind of thesaurus that is a mixture of taxonomy and collaborative tagging [28]. Although the category graph cannot be directly transformed into a taxonomy, the work presented in [19] shows some solutions to overcome this issue. Authors also describe a method for creating additional taxonomic relations between instance entities, directly from the entity descriptions.

The analysis presented in [30] shows that Wikipedia resources can be successfully used for various NLP and categorization tasks. Semantic relatedness presented in [26] can replace WordNet in classification and even outperform it. Finally, Wikipedia’s category network can be used to identify document topics, as described in [21]. This approach utilizes statistical methods based on the similarity of phrases in the document to entity names, and later, their category assignment.

4. Training-less text categorization

The proposed categorization method relies on converting the analyzed text into a semantic graph based on the ontological knowledge, and later finding categories that closely describe the constructed graph in terms of coverage of the entities in the graph, especially focusing on the core entities in the graph as

¹ <http://en.wikipedia.org>

well as the height of the covering categories in the category hierarchy.

We assume that the domain ontology used for the purpose of text categorization has a rich instance base of interconnected entities (with proper labels) that can be used for spotting them in the analyzed text. The entities are classified according to a taxonomy that will be used for categorization purposes. The target classification categories are defined as a taxonomy sub-hierarchy, list of related classes or mix of both the above. We also assume that the analyzed text is related to the knowledge domain represented in the ontology.

The outline of the categorization algorithm is presented below. The algorithm has two distinct phases: the construction of the semantic graph and its classification. The details of each step of the algorithm are explained in detail later in this section.

Semantic graph construction

1. Identify all named entities in the text of the document using different name labels in ontology associated with them and assign initial weights to the entities, based on the strength of each match; the entities are the nodes of the initial semantic graph.
2. Add the edges connecting the spotted entities, based on the relationships present in the ontology and establish the connectivity weights based on the importance of the relationship in the ontology schema.
3. Propagate and recalculate the weights of entities in the created graph; locate the entities with the highest weights, which are called authoritative entities.

Thematic graph identification and classification

4. Identify the dominant thematic graph, the largest and most important connected component of the semantic graph for further analysis.
5. Identify the central and authoritative entities in the dominant thematic graph.
6. Assign ontology categories to entities in the dominant thematic graph, based on the taxonomy categories included in the ontology schema.
7. Identify the target classification categories that (i) include the authoritative and the central entities, (ii) cover the largest part of the component, (iii) are closest to the graph entities in terms of their height in the category hierarchy.
8. The identified categories represent the classification of the document.

We now present our algorithm in detail. Example document, created relevant thematic graph and categorization result is presented in the appendix.

4.1. Semantic graph construction

The first step in preparing the text for ontology-based classification is the construction of the semantic graph, based on the text of the document. The purpose of having a semantic graph is to shift the analysis focus from the words, strings, and phrases occurring in the document to the entities and semantic relationships among them.

We will assume that word stemming and stop words removal may be applied to the document text before entity identification step. The ontology entities occurring in the analyzed document are identified by matching document phrases with entity literals (used as entity names) stored in the ontology. Such literals are usually represented as the values of certain properties associated with the entity and used as its identification. We assume that these properties define the entity name (usually known as its label), and may also specify the entity name's synonyms (aliases). We assign the *weight of an entity match* based on which of the identification properties was used in the match. We give preference to an exact match to the entity's label.

An entity name can be matched in several places in the document. It is important information, which is analogous to the term frequency used in the traditional text categorization methods. Such a multi-occurrence entity match is reflected by an increased weight of the entity. However, in order to limit a drastic increase in the weight of a frequently occurring entity, we use the following formula to establishing the initial weight of each entity:

$$w = 1 - \frac{1}{1 + \sum_{i=1..n} p_i * s_i}$$

In the formula, w is the initial entity weight and n represents the number of matches for the entity. The term p_i represents the weight of the identification property connecting the matched literal (name or alias) to the entity in the i -th match, and s_i is the measure of the similarity between the matched literal and text phrase, taking into account any differences introduced by word stemming and/or stop word removal. In case the entity identification process does not involve stemming and stop words removal, s_i set to 1.

Note, that a matched literal may point to multiple entities in the ontology, since different entities may have the same names or aliases. Therefore, the number of identified entities may be higher than the number of matched phrases in text. Many of them may be incorrectly identified (false positives), and will be eliminated later. However, at this stage, all of the

identified entities are used as nodes in the semantic graph being constructed.

Since all of the identified entities are represented as concepts or individuals in the ontology, the ontology may contain relationships connecting many pairs of them. Such existing relationships are added as edges to the identified entities (nodes) in order to form the *semantic graph* for the document.

The addition of the relationships into the semantic graph is a very important step in determining the categorization of the document. In fact, we view this step as the addition of the domain knowledge, represented in the ontology, in order to connect the discrete concepts (entities) in the document to form semantically related graph regions. The added ontological knowledge, even though it may not have been directly represented in the document, offers plausible semantic interpretations for co-occurrence of these entities. These semantic interpretations form the key information in determining the document classification.

Our categorization algorithm concentrates on most important and most central entities in the analyzed document. To recognize most important entities we utilize the hubs and authorities algorithm [13]. It helps to reinforce entities that are important according to ontology, even if they were underrepresented in the original text. In this approach we can also weigh different named relationships differently, in order to increase or decrease their importance reflecting the importance of their semantics. Such weights can be assigned according to the relationship rarity or other schema chosen by user.

4.2. Thematic graph and core entities

It is possible that the analyzed document covers more than one topic. In addition, during the entity matching phase, many entities may have been added to the semantic graph even though they are unrelated or, perhaps, weakly related to the main topic of the document. Furthermore, some phrases in the document might have led to the identification of multiple entities, but, perhaps, only one of them represents the proper match within the context of the document.

This step of the algorithm involves the selection of a sub-graph of the previously constructed semantic graph which represents the best interpretation of the recognized entities and relationships. We call such a sub-graph the *thematic graph*. The selection of the thematic graph is based on the assumption that the entities within one topic are related to each other, forming a connected component in the semantic graph. The semantic graph is created using the entities and relationships from the ontology, therefore the entities

and relationships in that component should fall within one topic (category). Entities in the semantic graph that are not connected to other entities, or that belong to other, perhaps smaller connected components most probably belong to other topics.

If a given document is focused on specific topics (which is the assumption of automatic text categorization), there should be a single or just very few *dominant thematic graphs* in the document's semantic graph that correspond to main topics of the document. For further analysis and categorization, we select a thematic graph that has the largest number of instances and has the largest total of entity weights. In case a few thematic graphs have very similar scores, all of them are included for further analysis. If more than one thematic graph has been selected, it can mean that the document is focused on more than one topic.

The selection of the dominant thematic graphs effectively eliminates the entities unrelated to the main topics of the document, such as incorrectly selected entities, or ambiguous entities that share the same name. Furthermore, the graph reduction entails the removal of *satellite* (or *fringe*) entities that are weakly related to dominant thematic graph. This step reduces the number of low-value information, decreases the level of noisy information, and enables to shift the analysis to the core topics of the document.

Furthermore, we compute the centrality score of the entities in the thematic graph in order to find the most central entities as *topic landmarks*. In our experiments, we used geodesic closeness measure to find most central entities. The geodesic closeness measure is defined as the reciprocal of the sum of the shortest paths between the selected vertex and all other vertices in the component:

$$Centrality(v_i) = \frac{1}{\sum_j d(v_i, v_j)}$$

where $d(v_i, v_j)$ is the shortest path distance in between vertices v_i and v_j (here, we treat the thematic graph as an undirected graph).

The calculation of the authorities and the centrality measure results in locating the core entities in the graph. The best authorities and the most central entities are selected as the core of the thematic graph. They are determined to be the most relevant to the document topic. Note, that the best authorities do not have to be the most central entities, and vice versa.

Selection of core entities should include both the best authorities and the most central entities. This ensures that the topic landmarks and important entities will be

included in the categorization step. The selection can include a certain percent of all entities from the thematic graph, or finding a good cut-off point. In our experiments, we decided to include up to 10% of all entities in the thematic graph the core entities from both groups. We also set minimum of 3 entities from each group to assure presence of most central and most important entities in graph core.

4.3. Thematic graph categorization

The categorization process shifts the attention from the dominant thematic graph, an instance level graph composed of the matched entities and relationships, to the taxonomy represented in the ontology schema. Each entity in the selected dominant thematic graph has its importance weight and almost each one of them has assigned at least one class in the taxonomy.

Finding a category that offers the best fit for whole thematic graph (or its part) is an optimization of a number of different, possibly conflicting objectives. The best category should:

- cover (be a class or super-class of) the highest number of entities in the thematic graph,
- be the lowest level category (in terms of the hierarchy of categories), and
- include the highest number of the core entities.

The category coverage of the thematic graph is illustrated in Figure 1.

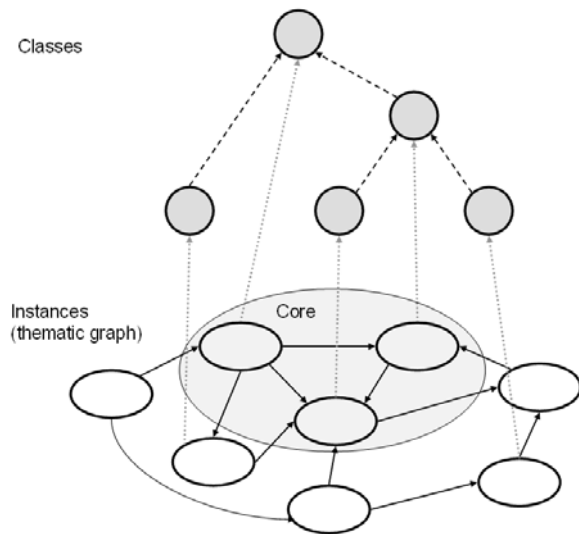


Figure 1. Thematic graph and categories

The selected class should offer the best fit for whole thematic graph (or its large part). Taking into consideration the properties of the best coverage class, we use the following formula for calculating the class score:

$$s_{C_i}(h_{max}) = 1 - \left(1 - \frac{1}{1 + \sum_j \frac{w_j}{h(C_i, e_j)^2} + \sum_k \frac{w_k}{h(C_i, e_{C_k})}} \right)$$

where s_{C_i} is the categorization score for class C_i , that includes reachable entities e up to depth h_{max} ; e_j and e_{C_k} represent respectively entity and core entity reachable up to depth h_{max} from class C_i ; w_j and w_{C_k} are weights of entity e_j and core entity e_{C_k} ; $h(C_i, e)$ is the hierarchical distance between category C_i and the covered entity e . The first summation is over all of the entities reachable from category C_i up to depth h_{max} , while second include only core entities.

Any class which does not cover at least one of the core entities is rejected, as it is not associated with the main topic of the document. The remaining classes, ranked according to their scores, represent the categorization of the document, relatively to the taxonomy in the ontology schema. At this step document has assigned multiple ranked categories that describe its content.

The final step of the document categorization requires matching between the internal categorization and user-defined topic categories. There are many possible approaches to match the assigned taxonomy classes with selected topics. We suggest following different approaches that depend on the method used in defining the topic categories.

The proposed score computation favors the classes that are closest to the thematic graph, not taking into account their depth in the taxonomy. In case the final topics are defined as parts of the taxonomy, we should favor matching to the lowest category in the hierarchy as the most specific one. Final categorization score s_{C_i} for class C_i should be increased as the depth of the matched class increases. To capture user's interest in matching hierarchy, score s_{C_i} can increase linearly, exponentially or in other preferred method with the depth of class C_i .

In case the class hierarchy is similar to that found in Wikipedia, where the defined categories instead of a clear taxonomy form a thesaurus, such an approach will likely give unsatisfactory results. In a thesaurus, the further we move away from the original category, the less relevant the matched category becomes. The categorization score should be modified to incorporate the relational (thesaurus-like) distance between classes.

Assigning external category is based on matching modified highest ranked classes to category definition. Starting from the class with the highest score, assign it to all appropriate categorization topics and increase their weight accordingly. Process can continue for all found categories, or only include selection of top k classes, or until one topic has dominant score.

5. Experiment design

In our experiments, we used the RDF ontology created from the English version of Wikipedia, using a slightly modified DBpedia approach [2] and text corpora of news articles gathered from the CNN Web site (www.cnn.com). Our implementation used Brahms as the backend RDF storage for the ontology [12]. We compared the accuracy of our training-less categorization method with one of the traditional, text categorization methods implemented by the BOW toolkit [15]. The BOW toolkit, similarly to other traditional methods, relies on the existence of a document training set, required for training of the classifier.

5.1. Wikipedia ontology

The RDF/S ontology was derived from Wikipedia dump from 2007-09-08. It contains 2,062,198 instance entities that contribute to a highly connected graph of 67,279,865 statements, and 4,409,200 literals that can be used for entity matching. On average, each entity has assigned 2.85 literals using different relationships. The schema part has 311,908 classes (Wikipedia categories) organized in 532,191 statements, mostly describing semi-hierarchical dependencies among the classes. Each entity, on average, has been assigned to 2.64 classes.

We utilized a modified DBpedia approach to create an RDF/S ontology from Wikipedia. Our modifications related to handling of the extraction of the templates included in a typical Wikipedia entry and assignment of literal values.

In DBpedia, the included templates (except from Infoboxes) become separate entities, connected to the source page. We shortcut these links and entities mentioned in the template content are set as directly related to the source entity. As these additional entities come from named templates, they are not linked by *the href* relationship, but by a named relationship derived from the template name. These named relationships are more important in our categorization method, as they carry more specific information about the existing connections between the entities, than simple *href* links.

The literal values, such as entity names, redirections, and disambiguation are very important for creating phrases to spot entities in the document. We have created separate named relationships to distinguish among the direct names of entities (page names in Wikipedia), redirections (redirection pages), and entities names included in the disambiguation pages. Wikipedia also utilizes a convention of disambiguating entity names by adding contextual information enclosed within parentheses and listed after the entity

name, e.g., “Jaguar”, “Jaguar (car)” and “Jaguar (band)”. Such full phrases do not exist in documents, as the context of the document provides enough information for human to properly disambiguate the entity. For entities with such names, we create a shorter literal by omitting the context information and add it as an alternate, shorter name, using specific property to distinguish it from the full name.

5.2. CNN text corpora

We tested the proposed categorization method on the recent CNN news articles (www.cnn.com) obtained from CNN RSS feeds between 2007-07-03 and 2007-09-04. The choice of recent news articles is related to choosing Wikipedia as our categorization ontology. Wikipedia is an encyclopedia of general knowledge that contains very recent entries. CNN articles describe facts from general knowledge and broadly defined CNN categories can relatively easy mapped to Wikipedia categories.

Our CNN text corpora is composed of 2,590 news articles assigned to 12 different categories. Each category was associated with a single RSS feed. For comparison with a traditional, probabilistic categorization method, we divided it into a 50/50 split, where the training and testing sets had 1,295 documents each. The selected categories with the split details are presented in Table 1.

Table 1 CNN text corpora details

	CNN Category	Train set	Test set
1	Education	4	7
2	Health	91	87
3	Money – autos	37	26
4	Money – companies	271	275
5	Money – taxes	15	12
6	Politics	171	167
7	Science and space	35	27
8	Sport – MLB	143	171
9	Sport – NBA	139	122
10	Sport – NFL	203	222
11	Sport – NHL	93	100
12	Travel	93	79

5.3. Category mapping: CNN and Wikipedia

The direct categorization to the ontology classes, as proposed in the description of our method, does not require supplying definition of the categories. For the purpose of evaluation of document categorization and comparison to one of the traditional methods, a

mapping between the selected CNN categories and suitable Wikipedia categories had to be created.

We decided to prepare the mapping between CNN and Wikipedia categories using a simple approach. For each CNN category, we have manually selected the main concepts from among the Wikipedia categories (roots) and added their subcategories up to the depth of 3. Depth limit was set due to fact, that the Wikipedia categories form not a taxonomy, but a thesaurus. Subcategories do not follow the strict semantics of the *rdf:subclass* property, but only are (closely) related to each other. The mapping of the roots of the selected Wikipedia categories to CNN categories and the numbers of the included subcategories in presented in Table 2.

Table 2 Wikipedia root categories for CNN classes.

CNN category	Wikipedia root classes	Number of subcategories
education	Category:Education	1467
health	Category:Health	1505
money_autos	Category:Automobiles	1350
money_companies	Category:Business Category:Economics Category:Stock_market	2509
money_taxes	Category:Accountancy Category:Taxation	146
politics	Category:Politics Category:Politicians	7746
science_and_space	Category:Science Category:Space	833
sport_mlb	Category:Baseball Category:Major_League_Baseball	1710
sport_nba	Category:Basketball Category:National_Basketball_Association	2438
sport_nfl	Category:Football Category:National_Football_League	8251
sport_nhl	Category:Hockey Category:National_Hockey_League	1858
travel	Category:Travel	714

5.4. Reference categorization method

We selected the Naïve Bayes classification method available in the BOW toolkit as a baseline for comparing categorization accuracy. We performed document categorization using two types of training sets. In the first experiment, we used as the training set a random subset of the Wikipedia entries (full pages) assigned to the categories identified in the created CNN category mapping. In this experiment BOW source of training documents differs from the categorized ones. In the second experiment, a BOW classifier was trained on the articles from CNN text corpora. This followed a traditional approach for

classifier training, where the training documents come from the same source as the documents to be classified.

The selected Wikipedia categories for our CNN category mapping cover over 400,000 entries. For each CNN category, we randomly chose up to 2,000 representative pages from Wikipedia and trained BOW on them. To check the consistency of the categorization results, 10 different training sets were created and tested. We believe that this experiment offered a better comparison with the direct ontology-based classification, as both the traditional (probabilistic) and ontology-based classifiers used the same source of information for the categorization task.

6. Experiment results

We performed three types of experiments:

- Our proposed training-less ontology-based categorization with the use of Wikipedia and CNN category mapping,
- BOW categorization trained on a subset of Wikipedia articles relevant to the CNN-mapped categories, and
- BOW categorization trained on our test split of CNN articles.

Categorization of CNN articles was performed using 1,295 documents from the testing set.

Our ontology-based method that used prepared CNN category mappings reached an accuracy of 80%. No training was necessary in this case. Different runs of categorization of the CNN corpora by Naïve Bayes categorization using prepared subsets of Wikipedia documents as the training set achieved only 73% accuracy. In this test both categorization methods used the same knowledge (ontology-based method) and documents (BOW) to perform categorization. Difference in original CNN categories and Wikipedia categories required to use prepared mapping.

When BOW was trained on the training set of the CNN articles, it was able to achieve accuracy slightly over 94%. In this case training and testing document came from the same source and no intermediate mapping was used.

The detailed categorization results from all three tests are respectively presented in Tables 3, 4 and 5.

Table 3 Ontology-based categorization of CNN document split using Wikipedia ontology with prepared category mapping.

CATEGORY	0	1	2	3	4	5	6	7	8	9	10	11	12	total	correct
0Education	6	1	7	85.71%
1Health	2	70	.	3	.	4	4	4	.	87	80.46%
2money_autos	.	.	17	8	.	.	1	26	65.38%
3money_companies	.	20	10	213	19	2	5	1	.	.	.	5	.	275	77.45%
4money_taxes	.	.	.	3	8	1	12	66.67%
5Politics	.	6	.	8	.	148	3	2	.	167	88.62%
6science_and_space	1	1	.	1	.	1	21	2	.	27	77.78%
7sport_mlb	.	2	.	6	.	3	1	153	.	.	.	6	.	171	89.47%
8sport_nba	.	3	.	7	.	12	3	.	91	1	.	4	1	122	74.59%
9sport_nfl	.	3	1	7	.	16	4	.	.	185	.	6	.	222	83.33%
10sport_nh	.	.	1	7	.	3	2	.	1	3	77	6	.	100	77.00%
11Trave	.	5	.	7	.	9	10	48	.	79	60.76%
12Unknown	0	0.00%
Classified documents : 1295															
Correctly classified : 1037															
Achieved accuracy : 80.077															

Table 4 Naïve Bayes categorization (BOW implementation) results of CNN document split with Wikipedia documents training set.

Correct: 1949 out of 1295 (73.28 percent accuracy); Confusion details, row is actual, column is predicted															
Classname	0	1	2	3	4	5	6	7	8	9	10	11	total	Correct	
0Education	7	7	100.00%	
1Health	23	55	.	1	1	.	3	4	87	63.22%	
2money_autos	.	.	23	3	26	88.46%	
3money_companies	1	11	16	169	74	4	275	61.45%	
4money_taxes	12	12	100.00%	
5Politics	11	4	.	2	40	100	10	167	59.88%	
6science_and_space	1	22	4	27	81.48%	
7sport_mlb	1	1	.	3	5	.	.	155	.	.	.	6	171	90.64%	
8sport_nba	2	.	.	3	4	.	.	.	99	1	.	13	122	81.15%	
9sport_nfl	13	1	.	9	7	2	.	.	8	160	.	22	222	72.07%	
10sport_nhl	5	.	1	7	5	.	.	.	2	.	75	5	100	75.00%	
11Travel	1	1	.	2	3	72	79	91.14%	
Percent_Accuracy average 73.28 stderr 0.00															

Table 5 Naïve Bayes categorization (BOW implementation) results of CNN document split with CNN training set.

Correct: 1220 out of 1295 (94.21 percent accuracy); Confusion details, row is actual, column is predicted															
Classname	0	1	2	3	4	5	6	7	8	9	10	11	total	correct	
0Education	2	2	4	50.00%	
1Health	.	80	.	4	2	2	.	3	91	87.91%	
2money_autos	.	.	16	20	1	37	43.24%	
3money_companies	.	1	4	263	.	2	1	271	97.05%	
4money_taxes	.	.	.	9	4	2	15	26.67%	
5Politics	.	2	.	.	.	169	171	98.83%	
6science_and_space	33	.	1	.	.	1	35	94.29%	
7sport_mlb	.	.	.	1	.	.	.	140	.	2	.	.	143	97.90%	
8sport_nba	1	135	3	.	.	139	97.12%	
9sport_nfl	1	.	.	.	202	.	.	203	99.51%	
10sport_nhl	.	.	.	1	2	.	90	.	93	96.77%	
11Travel	.	1	.	2	.	4	86	93	92.47%	
Percent_Accuracy average 94.21 stderr 0.00															

6.1. Analysis of the results

Our training-less ontology-based method achieved good results, compared to statistical method trained on Wikipedia knowledge, although when BOW was trained on source CNN articles, its accuracy was considerably higher.

We investigated the potential sources of misclassification problems in the CNN corpora. Analysis of several articles and created thematic graphs, together with the ranked categories revealed following causes of misclassifications:

- the created mapping between the CNN and Wikipedia categories were too broad and imprecise,
- the difference between the article's actual thematic content and the assigned category by CNN,
- an unevenly developed structure of Wikipedia link and category for different domains.

The imprecise mapping of CNN categories is both a result of ambiguity in defining CNN categories and the used category hierarchy in Wikipedia. In some cases, Wikipedia categories obtained by descending the Wikipedia category hierarchy were poorly related to the source category. In other cases, due to a thesaurus-like structure of Wikipedia categories, some categories were included in incorrect CNN mappings.

The second type of misclassifications is tightly related to the difference of article's actual content and a reader's *perceived* interest. It has been responsible for a larger portion of the misclassified documents. The ontology-based categorization analyzes the document content in order to create a thematic graph, and then finds its best fit into the ontological knowledge. On the other hand, the categories assigned by CNN mainly reflect the reader's perceived interest. The created Wikipedia-based ontology contains encyclopedic knowledge, which describes basic facts and their relationships. It does not favor any specific types of entities such as people, companies, or places. The (human assessed) perceived interest in the article may decide the article's category solely on a single type of high-interest entity, and not on the thematic content of the document.

As an example, consider an article about cardiovascular health problems of a certain politician. From a reader's perspective, the article belongs to *politics*, as the politician is the main point of interest. On the other hand, the majority of the document content is about the disease, treatment, or perhaps recovery. In the analysis of the created semantic graph, the politician most probably will not become one of the core entities, and

the graph core will concentrate on medical issues. This will result in the final categorization into the *health* domain.

Finally, some misclassifications were related to the ontology and Wikipedia itself. Some parts of Wikipedia are much better covered and interconnected than others. Consequently, entities from the better covered regions have a higher chance to be recognized and, due to their high connectivity, create a better thematic graph.

Focusing only on the document content, represented by entities and relationships, can be perceived both as the strength and the weakness of our categorization method. The strength comes from utilizing the background knowledge from the ontology that may not be present in the document. The weakness lies in the very difference between facts and perceived interests, which may require a much more sophisticated mapping or a modification of our algorithm to favor certain types of entities, relationships, or structures. We believe that it may be overcome by using certain NLP methods in building the thematic graph, and providing a more specific and defined context of interest in the classification step of the algorithm.

7. Conclusions and future work

In this paper we presented a novel text categorization method based on ontological knowledge that does not require a training set. The tests performed using an RDF ontology derived from Wikipedia demonstrated its effectiveness and practical value. In comparison with one of the statistical methods trained on the documents from the categorization ontology, our classification algorithm achieved nearly identical overall accuracy.

The presented approach and our experiments confirm that a rich and comprehensive ontology can be successfully used as a text classifier. The selection of a proper mapping between the ontology classes and user defined categories remains as an open question. In the near future, we plan to concentrate on defining a *categorization context* that could be used to specify perceived areas of interest for the user.

Another direction of future work is in including more semantics from the analyzed text. We plan to investigate the usefulness of NLP methods in discovering named relationships between the identified entities in the document itself. The relationships would be used for categorization in order to either strengthen the existing relationships in the knowledge base or to add additional information, not yet existing in the ontology.

References

- [1] WordNet: An Electronic Lexical Database. The MIT Press (1998)
- [2] Auer, S., Lehmann, J.: What have Innsbruck and Leipzig in common? Extracting Semantics from Wiki Content.: European Semantic Web Conference (ESWC'07). Springer, Innsbruck, Austria (2007) 503-517
- [3] Bada, M., Stevens, R., Goble, C., Gil, Y., Ashburner, M., Blake, J.A., Cherry, J.M., Harris, M., Lewis, S.: A Short Study on the Success of the Gene Ontology. *Journal of Web Semantics* **1** (2004)
- [4] Bloehdorn, S., Hotho, A.: Text Classification by Boosting Weak Learners based on Terms and Concepts. 4th IEEE International Conference on Data Mining (ICDM'04) (2004)
- [5] Brickley, D., Guha, R.V.: RDF Vocabulary Description Language 1.0: RDF Schema. In: McBride, B. (ed.): <http://www.w3.org/TR/rdf-schema/> (10 Feb 2004)
- [6] Buccella, A., Cechich, A., Brisaboa, N.R.: Ontology-Based Data Integration. In: Rivero, L.C., Doorn, J.H., Ferraggine, V.E. (eds.): *Encyclopedia of Database Technologies and Applications*. Information Science Reference (2005)
- [7] Deerwester, S., Dumais, S., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by Latent Semantic Analysis. *Journal of the Society for Information Science* (1990) **41** (1990) 391-407
- [8] Eccher, C., Purin, B., Pisanelli, D.M., Battaglia, M., Apolloni, I., Forti, S.: Ontologies supporting continuity of care: The case of heart failure. *Computers in Biology and Medicine* (2006) **Jul-Aug; 36** (2006) 789-801
- [9] Gabrilovich, E., Markovitch, S.: Overcoming the Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge. 21th National Conference on Artificial Intelligence, Boston, MA, USA (2006)
- [10] Gruber, T.: A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition* **5** (1993) 199-220, 1993
- [11] Hammond, B., Sheth, A.P., Kochut, K.J.: Semantic Enhancement Engine: A Modular Document Enhancement Platform for Semantic Applications over Heterogeneous Content. *Real World Semantic Web Applications*, IOS Press, 2002 (2002)
- [12] Janik, M., Kochut, K.J.: BRAHMS: A WorkBench RDF Store And High Performance Memory System for Semantic Association Discovery. Fourth International Semantic Web Conference (ISWC 2005), Galway, Ireland (2005)
- [13] Kleinberg, J.M.: *Authoritative Sources in a Hyperlinked Environment*. ACM-SIAM Symposium on Discrete Algorithms (1998)
- [14] Lewis, D.D.: Naive (Bayes) at forty: The independence assumption in information retrieval.: ECML-98, 10th European Conference on Machine Learning, Chemnitz, DE (1998)
- [15] McCallum, A.K.: Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering.: <http://www.cs.cmu.edu/~mccallum/bow> (1996)
- [16] Nagarajan, M., Sheth, A.P., Aguilera, M., Keeton, K., Merchant, A., Uysal, M.: Altering Document Term Vectors for Classification - Ontologies as Expectations of Co-occurrence. LSDIS Technical Report (November, 2006)
- [17] Ossenbruggen, J.v., Amin, A., Hardman, L., Hildebrand, M., Assem, M.v., Omelayenko, B., Schreiber, G., Tordai, A., Boer, V.d., Wielinga, B., Wielemaker, J., Niet, M.d., Taekema, J., Orsouw, M.-F.v., Teesing, a.A.: Searching and Annotating Virtual Heritage Collections with Semantic-Web Techniques. *Museums and the Web 2007*, San Francisco, California (2007)
- [18] Patel-Schneider, P.F., Hayes, P., Horrocks, I.: OWL Web Ontology Language Semantics and Abstract Syntax. <http://www.w3.org/TR/owl-semantics/> (10 Feb 2004)
- [19] Ponzetto, S.P., Strube, M.: Deriving a Large Scale Taxonomy from Wikipedia. Twenty-Second Conference on Artificial Intelligence (AAAI'07), Vancouver, Canada (2007)
- [20] Rosso, P., Ferretti, E., Jiménez, D., Vidal, V.: Text Categorization and Information Retrieval Using WordNet Senses. 2nd Global WordNet Int. Conf., GWN-2004, Brno, Czech Republic (2004)
- [21] Schonhofen, P.: Identifying document topics using the Wikipedia category network.: ACM International Conference on Web Intelligence (WI 2006), Hong Kong (2006)
- [22] Sebastiani, F.: Machine learning in automated text categorization. *ACM Computing Surveys (CSUR)* **34** (2002) 1 - 47
- [23] Semagix: Anti-Money Laundering - CIRAS. http://www.semagix.com/solutions_ciras.html.
- [24] Sheth, A.P., Arpinar, I.B., Kashyap, V.: Relationships at the Heart of Semantic Web: Modeling,

Discovering, and Exploiting Complex Semantic Relationships. In: Nikravesh, M., Azvin, B., Yager, R., Zadeh, L. (eds.): Enhancing the Power of the Internet: Studies in Fuzziness and Soft Computing. Springer Verlag (2003)

[25] Sheth, A.P., Bertram, C., Avant, D., Hammond, B., Kochut, K.J., Warke, Y.: Semantic Content Management for Enterprises and the Web. IEEE Internet Computing **July/August 2002** (2002)

[26] Strube, M., Ponzetto, S.P.: Wikirelate! Computing Semantic Relatedness Using Wikipedia.: Twenty-First Conference on Artificial Intelligence (AAAI'06), Boston, Massachusetts (2006)

[27] Vapnik, V.: The nature of statistical learning theory. Springer Verlag (1995)

[28] Voss, J.: Collaborative thesaurus tagging the Wikipedia way. ArXiv Computer Science e-prints **cs/0604036** (2006)

[29] Wu, S.-H., Tsai, T.-H., Hsu, W.-L.: Text categorization using automatically acquired domain ontology. 6th international workshop on Information retrieval with Asian languages - Volume 11, Sapporo, Japan (2003)

[30] Zesch, T., Gurevych, I.: Analysis of the Wikipedia Category Graph for NLP Application. Workshop - TextGraphs-2: Graph-based Methods for Natural Language Processing at NAACL Human Language Technologies Conference, Rochester, New York (2007)

Appendix – categorization example

Example text (downloaded from WikiNews²):

The **Boston Red Sox** are once again headed to the **World Series** after being down three games to one. **The Red Sox** were most recently in the 2004 **World Series**, which they won.

Last night's game ended with **the Red Sox** winning 11-2 over the **Cleveland Indians**. The \$103 million rookie import from Japan, **Daisuke Matsuzaka** (nicknamed "**Dice-K**"), pitched five innings for **Boston**, allowing two runs on six hits. Cleveland's **Jake Westbrook** started and took the loss. This proved to be a much better showing for **Boston's "Dice-K"** than his previous outing, which **Boston** lost.

The Sox jumped out to a quick lead, scoring a run in each of the first three innings on a single by **Manny Ramirez**, a sacrifice ground-out by **Julio Lugo**, and a sacrifice fly by **Mike Lowell**. The **Indians** scored their first run on a **Ryan Garko** double in the fourth inning, and a **Grady Sizemore** sacrifice fly in the fifth made the score 3-2 in favor of **the Sox**.

In the sixth inning, **Hideki Okajima** came in to relieve "**Dice-K**" and pitched two scoreless innings before **Jonathan Papelbon** came in to close in the eighth. He entered the game with runners on first and second and no outs, but quickly retired the side and in the ninth managed to maintain the nine run lead, once again giving fans a performance of his **Riverdance style victory dance**.

In the seventh inning, **Red Sox** rookie second baseman **Dustin Pedroia** hit a two-run home run off of **Rafael Betancourt** that drove **Boston's Fenway Park** wild. **Boston** scored a total of 6 runs in a crazy eighth inning, on a single by **J.D. Drew**, a three-run double by **Pedroia**, and a two-run **Kevin Youkilis** home run which bounced off a large Coke bottle advertisement. **Betancourt** gave up the first four runs, with the home run allowed by **Jenson Lewis**. **Youkilis** was the first batter he faced in relief of **Betancourt**.

The Sox will go on to face the **Colorado Rockies**, the surging **National League Champions**. The series will begin October 24th, with the first game at **Fenway Park**.

In this text, underlined words and phrases were recognized as entities in Wikipedia, but only the ones

in bold were selected to thematic graph for further categorization.

Created thematic graph is presented in Figure 2 on the next page. Most important and central entities are shaded in gray. Majority of relationships between selected entities are simple page references (href) represented as black arrows. Blue, bold arrows represent relationships via templates included in pages. They carry some more information than simple page references. Finally, red, bold arrows represent relationships discovered in Wikipedia's infoboxes. They are the most important connections between entities, as have specific semantic meaning, defined by infobox specification.

Most important and most central entities discovered in the thematic graph, which became the core entities for the categorization process:

- Boston_Red_Sox
- Home_run
- Run_(baseball)
- Single

After analysis of the thematic graph with special attention to the core entities, our algorithm assigned following Wikipedia categories (presented top 15):

- Category:Major_League_Baseball_teams
- Category:Sports_clubs_established_in_1901
- Category:Boston_Red_Sox
- Category:Major_League_Baseball
- Category:Sports_in_Boston
- Category:Singles
- Category:Baseball
- Category:Boston,_Massachusetts
- Category:Baseball_in_the_United_States
- Category:Sports_in_the_United_States
- Category:Baseball_teams
- Category:Sports_leagues_in_the_United_States
- Category:Sports_leagues_in_Canada
- Category:Sports_in_the_United_States_by_city
- Category:Baseball_leagues

The categories shown above are assigned using only document text and ontological knowledge. This is the categorization result of the proposed algorithm. Partial graph of Wikipedia categories associated with selected entities from the thematic graph is presented in Figure 3.

Using prepared mapping external to ontology-based categorization algorithm, the document was assigned to the CNN category *Sport MLB*.

² http://en.wikinews.org/wiki/Boston_Red_Sox_win_American_League_Championship

