An explainable data-driven approach to web directory taxonomy mapping

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 176 (2020) 1101–1110

Procedia Computer Science

www.elsevier.com/locate/procedia

24th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

An explainable data-driven approach to web directory taxonomy mapping

Elena Daraio[a,*], Luca Cagliero[a], Silvia Chiusano[a], Paolo Garza[a], Giuseppe Ricupero[a]

[a]*Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Turin, Italy*

Abstract

The spread of e-commerce and web applications has fostered the integration of cross-domain business activities. To efficiently retrieve products and services, web directories allow customers to browse multiple-level taxonomies to find specific products or services according to a predefined categorization. Providers need to periodically update web directory lists by aligning in-house taxonomies to domain-specific hierarchies coming from external sources. However, such taxonomy mapping procedures are often semi-automatic and rely on traditional word disambiguation techniques to capture the semantics behind categories and products descriptions. Hence, the flexibility and explainability of the underlying models are quite limited.

This paper proposes an automated, explainable approach to web directory taxonomy mapping based on text categorization. It exploits two complementary word-based text representations: a frequency-based representation, which captures syntactic text similarities, and an embedding one, which highlights the underlying semantic relationships among words. Since the proposed solution is purely data-driven, it can be successfully applied to business domains where there is a lack of semantic models. The frequency-based text representation has shown to be particularly suitable for driving the automated taxonomy mapping procedure, whereas the embedding space has been profitably used to provide local explanations of the category assignments.

Keywords: automated taxonomy mapping; web directories; text Categorization; vector representations of text

1. Introduction

In the era of Industry 4.0, companies need to tightly integrate business activities managed by different providers and related to various domains. With the advent of e-commerce and Internet of Things technologies, one of the key aspects of product/service provision is the availability of a multiplicity of web platforms [1]. For example, it is very common to find the same product offered by various e-commerce platforms, to book the same hotel using different online travel agencies, and to receive personalized item recommendations on social media based on the activities

* Corresponding author. Tel.: +39-011-090-7084.
E-mail address: elena.daraio@polito.it

1877-0509
10.1016/j.procs.2020.09.106

carried out in other virtual environments. Due to the pervasive nature of web-based applications, getting a unified view of the actual demand and supply of specific service and product categories is particularly challenging.

With such new market scenarios that have been opening, web directories pose themselves as a concrete support to meet customer needs. They consist of hierarchies of websites providing a multiple-level and multi-faceted categorization of business activities. Specifically, they list entries on people or businesses, classifying them as the corresponding categories and sub-categories [2]. Unlike search engines, they do not rely on web crawling, but they are often curated by human editors thus guaranteeing high-quality categorizations.

To keep their lists updated, web directory providers continuously acquire new information regarding domain-specific business activities from external partner organizations. To integrate such an external knowledge, ad hoc taxonomy mapping procedures have been proposed in literature [3]. Most of them rely on Natural Language Processing (NLP) techniques, aimed to disambiguate the meaning of the text of the category descriptions in order to find the best match. However, since, to our best knowledge, the currently available solutions rely on ad hoc ontology-based models, they encounter many issues related to model adaptability, portability, and explainability. First, they often rely on semi-automatic processes requiring human intervention. Secondly, they are, in most cases, not easily portable to different contexts where semantic knowledge is not available. Finally, due to the inherent characteristics of the models, explaining the reasons behind specific category assignments is often unfeasible. A more thorough comparison with the existing solutions is given in Section 2.

This paper presents a text categorization approach to automated web directory taxonomy mapping. It addresses the goal of automatically mapping products/services already classified in a coarser taxonomy to a predefined hierarchy of categories of a target fine-grained. The automated mapping strategy is based on the analysis of the textual descriptions of the products/services to be added and of the categories in the in-house taxonomy. Multiple-level category assignments are performed in a top-down fashion, i.e., from the upper-level categories to the most specific sub-categories. Textual descriptions are represented using two complementary word-based representations: (i) a *frequency-based model* [4], which counts word co-occurrences to measure text similarities, and (ii) a *latent embedding model* [5], which exploits a high-dimensional vector representation to discover the underlying semantic text relationships. Notably, the latter representation enables local explanations, thus providing promptly usable feedback on the quality of the category assignments. The experiments show the effectiveness of the frequency-based model to accurately predict category membership and the usability of the latent model to explain the achieved results.

The rest of the paper is organized as follows. Section 2 presents a literature review. Section 3 states the problem addressed in the present study. Section 4 describes the proposed approach. Section 5 reports the results of the performed experiments, while Section 6 draws conclusions and discusses the future developments of this research study.

2. Related work

The present paper addresses the task of automated taxonomy mapping, which is a subcase of the more general ontology mapping problem. Notably, the problem is still open despite an extensive research has already been carried out. As further proof of the fact that the problem is still open, several companies still address web directory mapping in a semi-automatic way with the aid of specialized software tools (e.g., PROMPT [6]). However, the aforesaid approach turns out to be ineffective while coping with multiple-level, complex hierarchies. For example, a list of restaurants can be divided in at least 25 categories, where each category is further split into many sub-categories. A manual inspection of all the taxonomy categories to find the best match could be extremely time-consuming. According to Falconer et al. [7], finding the optimal solution to the ontology mapping problem is often practically unfeasible due to language ambiguity and inherent complexity of the Natural Language Processing models.

Thor et al. [3] have classified the main ontology mapping strategy as: (i) *metadata-based*, which exclusively rely on the information contained in the ontology; (ii) *instance-based*, when the specific instances of each concept are leveraged in the process; (iii) *mixed*, when metadata information and instances are jointly exploited.

To the best of our knowledge, the majority of the proposed solutions rely on Natural Language Processing (NLP) techniques, including word- and graph-embedding approaches (e.g., [8, 9, 10]). They consider the semantic relationships between ontological concepts to automate the process of category mapping. Conversely, the present study is purely data-driven, i.e., it does not rely on any ontological model. Such a property is particularly desirable when there is a lack of domain-specific knowledge.

Agrawal & Srikant [11] have proposed an instance-based Bayesian classification approach, which maps the content of a source taxonomy to that of a reference model (i.e., the master taxonomy). According to the concept of locality, documents of the same category are likely to be assigned to the same target in the reference model. A mixed graph-based approach, called *SimilarityFlooding*, has been proposed by Melnik et al. [12]. Despite the proposed ontology mapping approach can be adapted to map taxonomies, its structure has shown to be not efficient while coping with hierarchies of concepts [7]. More recently, in [13], Skinner et al. have proposed a strategy to map e-commerce queries to product categories in the pre-defined taxonomy. Due to the lack of training data, the approach relies on a transfer learning strategy: a predictive model able to forecast the category of a product given its title is trained first. Then, the model is applied to predict the category of the input queries. The idea behind is that product titles and e-commerce queries are likely to be similar. The approach presented in this work is not query-based, but rather focuses on mapping the content of two taxonomy models. Since a partial taxonomy mapping is given, in our context the use of a supervised approach is fully justified.

3. Web directory taxonomy mapping: problem statement

Web directory editors periodically update the lists with external content. Let $\mathcal{T}$ be the in-house web directory taxonomy aggregating products/services into their upper-level categories. Categories are organized in fine-grained hierarchies, where sub-categories grouping a subset of products/services are further generalized as an upper level category. Leaf nodes of the taxonomy represent arbitrary elements e of the web directory list (i.e., single products or services). Upper-level elements in $\mathcal{T}$ represent instances of hypernym categories (i.e., product/service categories). Both low- and high-level elements are characterized by a textual description of the corresponding instances.

For example, the in-house taxonomy depicted in the right hand-side of Figure 1 splits category *Restaurants* into the lower-level specialization *Italian Restaurants* and *Chinese Restaurants*, which in turn can be further split into lower-level sub-categories. According to the level of generalization, descendant categories of the same ancestor are characterized by the same aggregation level (e.g., level l=2 for *Italian Restaurant*, l=1 for *Restaurants*).

External content is commonly retrieved from third-party taxonomies. An external taxonomy $\mathcal{T}_e$ aggregates new elements e^*, not present in $\mathcal{T}$, typically based on coarser aggregation hierarchies. Upper-level categories in $\mathcal{T}_e$ can be straightforwardly mapped to a specific aggregation level l^* in $\mathcal{T}$. However, mapping elements e^* to categories in $\mathcal{T}$ is a challenging (and potentially time-consuming) task.

The mapping procedure aims at assigning a new element e^* in $\mathcal{T}_e$ to the most pertinent categories in the in-house fine-grained taxonomy $\mathcal{T}$. To this end, the in-house taxonomy is explored in a top-down fashion, starting from the reference level l^*. Once e^* is assigned to the most likely level-l^* category, then the procedure iterates on the hyponyms of level l^*-1 until either an upper bound of the prediction error is exceeded or the taxonomy is fully explored. The idea behind is that identifying the upper-level category first simplifies the task of deciding which lower-level categories the new element is most likely to belong to.

For example, to add a new restaurant W to the in-house fine-grained taxonomy, we exploit the mapping between the Category *Restaurants* in the external, coarser taxonomy (see the left hand-side taxonomy in Figure 1) and the level-1 category *Restaurants* in the in-house taxonomy. This triggers the exploration of the in-house taxonomy starting from the level-1 descendant nodes of *Restaurants*.

Given an arbitrary aggregation level l, the procedure first identifies the subset of level-l candidate elements C^l={e^l_1, e^l_2, …, e^l_m} and then estimates the probability $p(\mathcal{F}(e^*, l) = x)$, where $\mathcal{F}$ is a function mapping an arbitrary element e^* to a specific candidate element $x \in C^l$.

The automated taxonomy mapping process at level l entails assigning the most likely candidate element. It can be formulated as a single-label classification task $arg_{j\,|1\leq j\leq m}\ max\ p(\mathcal{F}(e^*, l) = e^l_j)$.

4. Methodology

The automated taxonomy mapping strategy proposed in this study is summarized in Figure 2. It entails the following steps:

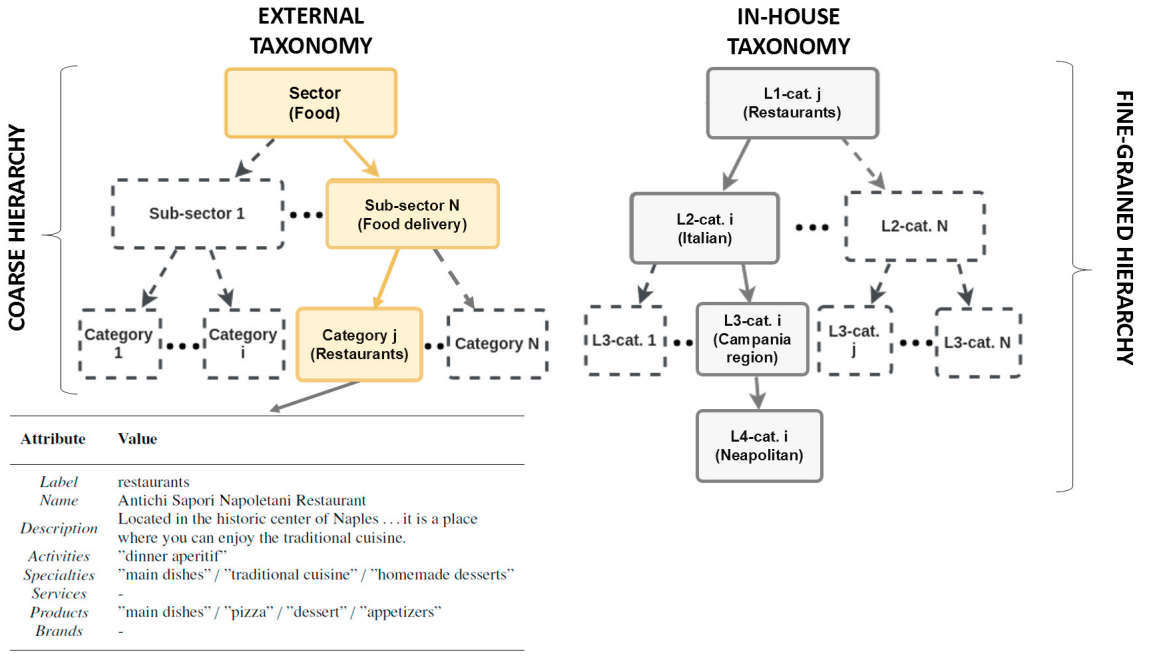


Fig. 1: The automated taxonomy mapping process.

1. *Text preparation*: It entails cleaning the textual descriptions of the taxonomy elements to enable the next analytical steps. It includes tokenization of text into words and stopword elimination to remove irrelevant words.
2. *Data modelling*: It models data using two complementary word-based representations, i.e., the occurrence-based and the vector representations.
3. *Text categorization*: It trains and applies classification algorithms on the input data to automatically assign the most likely categories to each new element retrieved from the external taxonomy.
4. *Local and global model explanation*: It provides evidences of the reasons behind a specific category assignment as well as a description of the word-based prediction model.

The remaining part of this section is organized as follows. Sections 4.1 and 4.2 detail the occurrence-based and the vector representations of text, respectively. Sections 4.3 thoroughly describes the text categorization steps, while Sections 4.4 details the steps related to local and global explainations.

4.1. Occurrence-based representation of text

The occurrence-based text representation describes taxonomy elements by counting the number of occurrences of words in the corresponding descriptions [4]. It can be used either to measure the syntactic similarity between different snippets of textual description or to identify the most frequently occurring words in the descriptions of single product/services and of the corresponding categories.

The key idea is to describe taxonomy categories by using the words that most frequently occur in the corresponding product/service descriptions. Such a popular text representation is particularly suitable for text categorization because word occurrences are likely to be correlated with the assignment of a specific category.

Let $D = \{d_1, \ldots, d_n\}$ be a collection of textual descriptions of the in-house taxonomy elements and let $\mathcal{W} = \{w_1, \ldots, w_k\}$ be the set of occurring words in D. The *Term Frequency* (TF) of word w_j in d_i, denoted as TF_{d_i,t_j}, indicates the relative frequency of occurrence of word w_j in d_i. It is computed as $f_{d_i,w_j} / \sum_{1 \le k \le |\sum|} f_{d_i,w_j}$, where f_{d_i,w_j} indicates the number of times w_j is contained in d_i and $\sum_{1 \le k \le |\sum|} f_{d_i,w_j}$ is the total number of words in d_i.

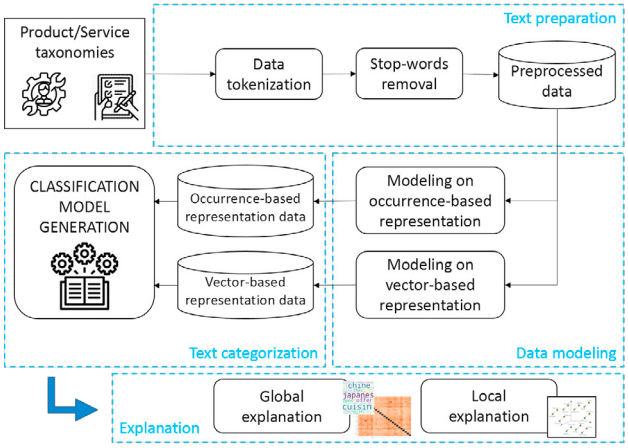


Fig. 2: The automated taxonomy mapping strategy.

The *Inverse Document Frequency* (IDF) of word w_j in d_i, denoted as IDF_{d_i,w_j}, is expressed as $\log(\frac{|D|}{|d_k \in D: f_{d_k,w_j} \neq 0|})$ where $|D|$ is the total number of descriptions and $|d_k \in D : f_{d_k,w_j} \neq 0|$ is the number of descriptions containing word w_j at least once. Mathematically speaking, the base of the log function for IDF computation is irrelevant and just represents a scaling factor.

The *Term Frequency-Inverse Document Frequency* (TF-IDF) w_{d_i,w_j} of word w_j in description d_i is computed as $w_{d_i,w_j} = TF_{d_i,w_j} * IDF_{w_j}$, where TF_{d_i,w_j} is the *Term Frequency* and IDF_{w_j} is the *Inverse Document Frequency* [4]. Notice that the TF-IDF value is high when the word w_j frequently occurs in d_i, whereas its frequency in the descriptions of D is low. When word w_j occurs in many descriptions, the log function tends to one and the IDF_{d_i} value and TF-IDF w_{d_i,w_j} reduces to zero. Hence, the most common words are automatically filtered out since they are deemed as not discriminating.

4.2. Vector representation of text

We adopt a Deep Natural Language Processing (NLP) model to explain the correlations between service/product descriptions and taxonomy categories. Deep NLP studies the application of Deep Learning techniques to accomplish Natural Language Processing tasks [14]. In the present study we exploit word-based embedding models to get semantically richer text descriptions. Embedding models are directly mapped to latent vector spaces, where the learned vectors inherently encode linguistic regularities. Hence, the key reasons behind their use can be summarized as follows: (i) the ability to capture the underlying semantic relationships among text (independently of the language) and (ii) the inherent explainability of the embedding models, which can be easily adapted to produce explainable results.

An embedding function $f: \mathbb{W} \rightarrow \mathbb{V}$ maps words $w \in \mathbb{W}$ occurring in the textual descriptions to a high-dimensional vector space V, where each vector $V^w \in \mathbb{V}$ corresponds to a specific word w (hereafter denoted as *word embedding* for the sake of simplicity). Word vectors are generated using the established FastText embedding [15].

Thanks to the properties of vector representations, word embeddings can be usefully combined to get vectors representing specific products/services or the corresponding categories. Hereafter, they will be denoted as *product/service embeddings* V^{ps} and *category embeddings* V^c, respectively. Specifically, textual descriptions are first tokenized to get the corresponding words. Then, the resulting description embedding is generated by computing the pointwise average over the vector dimensions of the token embeddings, i.e., $V^{ps}=avg_i\ v_i^w$.

Product/service embeddings V^{ps} describe the products/services listed in the web directory by reflecting the content of the corresponding textual description. Similarly, category embeddings V^c describe specific categories according to their description. Notably, the produced vector representations can be easily compared with each other by computing their pairwise distance in the vector space. Descriptions including semantically related content are likely to produce similar latent representations.

4.3. Text categorization

Given a training dataset consisting of a set of textual descriptions labeled with the corresponding taxonomy categories, the aim of the text categorization step is to build a predictive model able to automatically assign the most likely categories to a new taxonomy element (for which the label is unknown). The problem, formally stated in Section 4.2, is accomplished by applying established multi-class classification models, among which Support Vector Machines (SVM), Decision Trees, Random Forest, Gaussian Naive Bayes (GaussianNB), K-Nearest Neighbor (KNN) [4].

To build the training dataset on top of the occurrence-based text representation, a document-term matrix M is generated, where rows correspond to taxonomy element descriptions (labeled with the membership category in the in-house taxonomy) whereas columns corresponds to distinct words. Each cell m_{ij} contains the TF-IDF of word w_j in the element description d_j. To apply classification algorithms on the vector representation of text, each row in the dataset corresponds to a different product/service embedding e^{ps} and is labeled with the corresponding category in the in-house taxonomy.

4.4. Local and global model explanation

EXplainable Artificial Intelligence (XAI) is a recently opened branch of AI specifically addressing the lack of transparency of many popular Machine Learning models [16]. In the context of taxonomy mapping, the problem of making models easily interpretable by domain experts is particularly relevant. In fact, the majority of the classification models that have shown to be effective in text categorization (e.g., Support Vector Machines, Neural Networks) are inherently not explainable. Therefore, the aim of this study is to exploit the occurrence- and vector-based representations of text to leverage the explainability of the text categorization models embedded into the automated taxonomy mapping strategy. To address the above-mentioned issue, we envision two complementary strategies, respectively falling into the *local* and *global* explainability categories.

Local explainability in the unified latent space. Local explainability entails providing a targeted explaination of a specific classifier assignment. In our context, it entails understanding the rationale behind the assignment of a specific category to a new taxonomy element. Getting an intuitive and promptly usable explaination of a category assignment is particularly helpful for web directory editors because it simplifies the validation process on top of the automatically generated mapping.

To our purpose, we explore a unified latent vector space $\mathbb{V}^* = \mathbb{V} \cup \mathbb{V}^{ps} \cup \mathbb{V}^c$ incorporating three complementary vector spaces:

- the embeddings of single words occurring in the textual descriptions (i.e., the vector space $\mathbb{V}$)
- the product/service embeddings representing the descriptions of the elements associated with a taxonomy leaf node (i.e., the vector space $\mathbb{V}^{ps}$)
- the category embeddings representing the target categories in the in-house taxonomy (i.e., the vector space $\mathbb{V}^c$).

Notice that, by construction, the three embedding spaces are fully compatible with each other [15]. Hence, the pairwise distances between the vectors in different spaces can be computed.

For each product/service category we explore the unified latent space to find (i) the K-Nearest words[1], occurring in the product/service description, to the actual category label and (ii) the K-Nearest words, occurring in the product/service description, to the predicted category label. The former words describe the product/service to be classified by reflecting the actual category, whereas the latter ones describe the same taxonomy element by reflecting the predicted category.

Global explainability of the text categorization models. Global explainability provides a detailed view on the text categorization model (as a whole) to highlight the most discriminating features. In our context, we exploit such an exploration to describe the target categories and the most significant word correlations.

To achieve our goal, we exploit two complementary model representations to visually explore the characteristics of the target categories, respectively the *graphical view* and a *tabular view*. The graphical view aims at highlights the

[1] Vector similarities in the latent space are estimated via cosine similarity [4]

peculiar characteristics of the text categorization model. It relies on established visual techniques, i.e., wordclouds and heatmaps, to provide a global explanation. Wordclouds are commonly used to highlight word relevance in a document collection. In our context, we exploit such a graphical representation to summarize the most frequently occurring words within a specific target category. Heatmaps are used to evaluate the reciprocal distance among the target categories, where each category is summarized by the centroid of its corresponding descriptions. This highlights the similarities among categories thus explaining eventual performance drops in text categorization. The tabular view provides editors with a keyword-based summary of each (actual and predicted) category consisting of the most relevant words in the corresponding element descriptions. Word relevance to a specific category is computed in the unified latent space $\mathbb{V}^*$ as previously described for the local explainability task. We exploit the tabular view to quickly identify the most discriminating words peculiar to each category.

5. Experimental results

We validated the performance of the proposed taxonomy mapping strategy on real data provided by an Italian web directory. The remaining part of this section describes the experimental design, while Sections 5.1 and 5.2 summarize the main classification results and exemplify the local and global explanations extracted from the real dataset, respectively.

Dataset We analyzed the English version of a web directory dataset provided by an authoritative Italian company. The dataset consists of 170,000 entries, among which approximately 4% of the entries were acquired from external sources. On the latter entries we applied the taxonomy mapping strategy to simulate the automated machine learning-based process and compare the automatically assigned categories with the actual (humanly generated) assignments.

The input dataset contains business activities' data related to eight different industrial fields. The goal of the automated taxonomy mapping process is to automatically assign the *label* descriptor of the entry, which corresponds to the taxonomy category. To this purpose, the text categorization models have been trained on the textual descriptions available in the *description* field.

To explore the capability of the approach to effectively handle multiple-level data, we evaluate the performance in the classification of top taxonomy levels categories, but we also deepened the analysis on the corresponding sub-categories. For example, the entries of category *restaurants* have been classified as 25 different sub-categories (e.g., *restaurants.italian*), which in turn have been split into 15 sub-categories (e.g., *restaurants.italian.altoatesine*). Sub-category *restaurants.italian* has been considered as representative case study of this in-depth analysis.

Text processing To process multilingual descriptions in textual form we rely on the *Natural Language Toolkit* library and the FastText project [17].

Text categorization We trained various classification models available in SK-Learn machine learning library, i.e., Support Vector Machines (SVM), Decision Trees (DT), Random Forest (RF), Gaussian Naive Bayes (GaussianNB), K-Nearest Neighbors (KNN). To properly set the algorithm parameters, we performed a grid search.

5.1. Text categorization performance

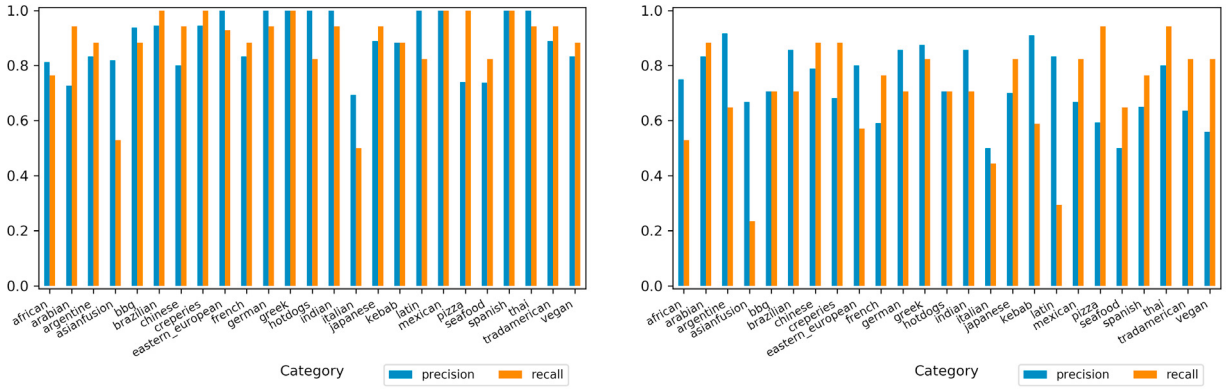
We evaluated text categorization performance in terms of classifier accuracy and per-class precision, recall, and F1-score (i.e., harmonic mean of the precision and recall) [4]. To perform performance validation, we applied a cross-validation process with leave-one-out sampling.

For the top taxonomy level categories, the average accuracy, precision, and recall values of the best performing classifier (i.e., SVM) are all around 89% using the occurrence-based text representation, whereas they drop to 72% (on average) using the latent vector representation. Such a performance gap is probably due to the limited capability of text categorization algorithms to capture significant correlations among embedding features and target categories. Conversely, word occurrences have shown to be particularly effective to discriminate among the candidate classes. Figures 3a and 3b detail the per-class performance analyses separately for each category in the top taxonomy level (e.g., *restaurant.mexican*, *restaurant.french*) for the SVM classifier. The performance gaps between the two model types remain unchanged even on the most complex categories (i.e., *asianfusion* and *italian*).

As for SVM, also for all the other evaluated classification algorithms best results are achieved for the top taxonomy level categories using the occurrence-based text representation. Results are summarized in Table 1.

Fairly good results were achieved by the SVM classifier on the occurrence-based text representation even for the first level of sub-categories of *restaurants.italian*, with average accuracy, precision, and recall values all around 74%.

Complexity SVM classifier training took around 600s on the occurrence-based text representation and 90s on the vector representation of text. The complexity gap is mainly due to data dimensionality, which vary from 3700 (with the TF-IDF scores) to 300 (with FastText). Classification time is negligible for all the classifiers and is roughly the same for both representations.



(a) Occurrence-based representation

(b) Vector representation

Fig. 3: Per-category performance comparison for SVM classifier.

Table 1: Model comparison on the occurrence-based representation. Top-level categories in the taxonomy

Model	Main parameters setting	Avg Accuracy	Avg Precision	Avg Recall	Avg F1-score
SVM	kernel = 'linear'	**88.6%**	**89.3%**	**88.8%**	**88.6%**
KNN	k = 3	77.8%	80.3%	77.9%	77.4%
DT	criterion = 'gini'	68.3%	69.3%	68.4%	68.5%
RF	criterion = 'gini'	63.4%	63.2%	63.5%	62.6%
GaussianNB	priors = None	62.4%	67.8%	62.4%	63.3%

5.2. *Example of explanations*

Table 2 reports some examples of misclassified elements. For each of them, it indicates the actual and assigned category as well as the most semantically related words occurring in the element description according to their similarity, in the latent space, with the actual and assigned category, respectively. Notice that the lists of related words in the misclassified examples contain many words in common or semantically related words. Hence, the misclassified examples seem to be related to both the actual and the predicted class. For instance, consider the first misclassification example reported in Table 2. Three out of five of the most semantically related words associated with the actual category occur also in the list of the most semantically related words associated with the predicted category, and also the words that are not in common are semantically related with each other. This means that the description of the product/service we are classifying contains many words that are semantically related to both classes (the actual and the predicted ones). Such information provides a local explanation of the wrong category prediction for this specific restaurant, which, to a

Table 2: Local Explainability: examples of wrong category assignments

Label		**Most semantically related words**	
Actual category	**Assigned category**	**Actual category**	**Assigned category**
Asianfusion	Japanese	['sushi', 'cuisin', 'restaur', 'japanes', 'sashimi']	['japan', 'japanes', 'sushi', 'sashimi', 'taiyo']
Bbq	Pizza	['grill', 'oven', 'meat', 'restaur', 'lunch']	['pizzeria', 'oven', 'lunch', 'restaur', 'takeaway']
Seafood	Italian	['seafood', 'dish', 'restaur', 'fresh', 'raw']	['restaur', 'homemad', 'gallipoli', 'local', 'seafood']

Correctly assigned category	**Most semantically related words**
Japanese	['japanes', 'masaki', 'cuisin', 'inoguchi', 'food']
Mexican	['mexican', 'mexico', 'fajita', 'burrito', 'chili']
Greek	['greek', 'souvlaki', 'taverna', 'athen', 'olympian']
Thai	['thailand', 'asian', 'rice', 'coconut', 'seafood']
Brazilian	['brazilian', 'picanha', 'feijoada', 'churrasco', 'moqueca']

Table 3: Local Explainability: examples of correct category assignments

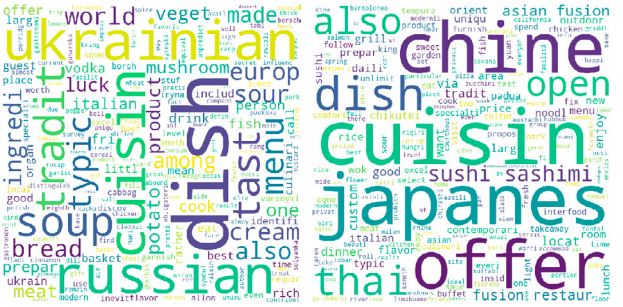


Fig. 4: Global Explainability: Wordclouds representation of categories Eastern European (left) and Asianfusion (right)

good approximation, could be assigned to both classes. Table 3 reports some examples of correctly classified elements (restaurants). For each of the considered restaurants, we report the 5 most semantically related words occurring in the restaurant description. We can notice a strong semantic correlation between the actual category name of each restaurant and the words selected in the description. Such information can be used to explain the correctness of aforesaid category assignments.

Figures 4 and 5 report the wordclouds associated with two restaurant categories (*eastern European* and *asianfusion*) and the global correlation matrix, respectively. The most frequent words occurring in the wordcloud of the eastern European (restaurant) category are all related to the expected category, whereas the wordcloud extracted for the *asianfusion* restaurant category contains several frequent words associated with other categories (e.g., Chinese, Japanese). This high-level description provides a global explanation of the prediction errors we achieved on the restaurants belonging to the *asianfusion* category. The heatmap reported in Figure 5 confirms the presence of a strong correlation between the *asianfusion* restaurants and the Chinese and Japanese ones. Hence, misclassifications among the restaurants of those three categories are not surprising, because many of those restaurants have similar descriptions and the classifier was not able to identify peculiar words.

6. Conclusions and future work

This paper has investigated the problem of automatically mapping external elements to existing categories of a web directory taxonomy using text categorization techniques. The problem is particularly challenging when either domain-specific semantically rich models are not available or there is a need to automate the category assignment process without lacking of model transparency. We adopt two complementary text representations, which provide complementary information about syntactic and semantic properties of the analyzed text.

We carried out a preliminary experimental analysis on a real web directory dataset to demonstrate the effectiveness and efficiency of the proposed approach. The results show that text categorizations relying on occurrence-based text representation achieved promising classification performance, whereas latent representations of text, based on word embedding techniques, provided intuitive interpretation of the classifier decisions.

Since the proposed approach is easily portable to text written in different languages, we plan to extend the current taxonomy mapping procedure to multilingual web directory data. Furthermore, we aim at extending the current analysis to other business domains, such as online travel agencies and online medical reviews.

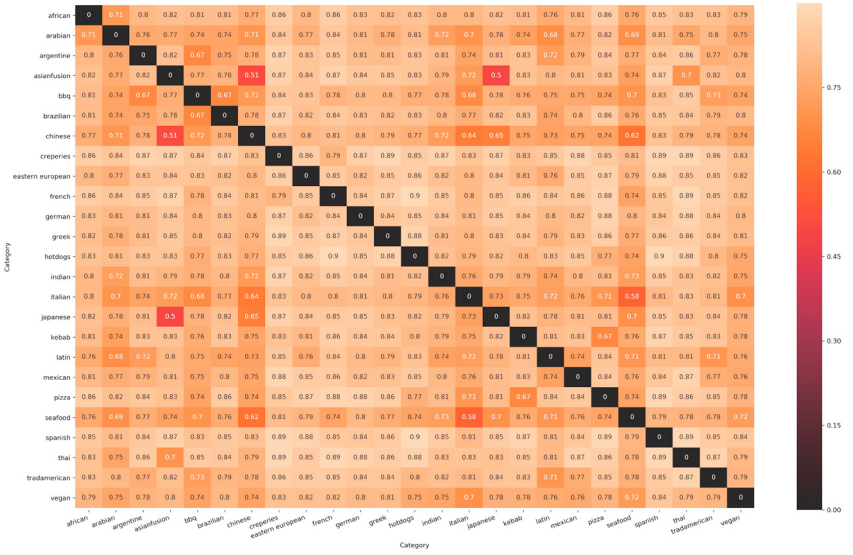


Fig. 5: Global explainability: Comparison among centroids of categories

References

[1] H.-N. Dai, H. Wang, G. Xu, J. Wan, M. Imran, Big data analytics for manufacturing internet of things: opportunities, challenges and enabling technologies, Enterprise Information Systems 0 (0) (2019) 1–25. doi:10.1080/17517575.2019.1633689.

[2] H.-C. Yang, C.-H. Lee, A text mining approach on automatic generation of web directories and hierarchies, Expert Systems with Applications 27 (4) (2004) 645 – 663. doi:https://doi.org/10.1016/j.eswa.2004.06.009.

[3] A. Thor, T. Kirsten, E. Rahm, Instance-based matching of hierarchical ontologies, in: Datenbanksysteme in Business, Technologie und Web (BTW 2007), 12. Fachtagung des GI-Fachbereichs "Datenbanken und Informationssysteme" (DBIS), Proceedings, 7.-9. März 2007, Aachen, Germany, 2007, pp. 436–448.

[4] P.-N. Tan, M. Steinbach, A. Karpatne, V. Kumar, Introduction to Data Mining (2Nd Edition), 2nd Edition, Pearson, 2018.

[5] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States, 2013, pp. 3111–3119.

[6] N. F. Noy, M. A. Musen, Using prompt ontology-comparison tools in the EON ontology alignment contest, in: EON 2004, Evaluation of Ontology-based Tools, Proceedings of the 3rd International Workshop on Evaluation of Ontology-based Tools, 2004.

[7] S. M. Falconer, N. F. Noy, M. D. Storey, Ontology mapping - a user survey, in: Proceedings of the 2nd International Workshop on Ontology Matching (OM-2007), Busan, Korea, November 11, 2007, 2007.

[8] S. Park, W. Kim, Ontology mapping between heterogeneous product taxonomies in an electronic commerce environment, Int. J. Electron. Commer. 12 (2) (2007) 69–87. doi:10.2753/JEC1086-4415120203.

[9] S. S. Aanen, D. Vandic, F. Frasincar, Automated product taxonomy mapping in an e-commerce environment, Expert Syst. Appl. 42 (3) (2015) 1298–1313. doi:10.1016/j.eswa.2014.09.032.

[10] T. Wu, D. Zhang, L. Zhang, G. Qi, Cross-lingual taxonomy alignment with bilingual knowledge graph embeddings, in: Z. Wang, A. Turhan, K. Wang, X. Zhang (Eds.), Semantic Technology - 7th Joint International Conference, JIST 2017, Proceedings, Vol. 10675 of Lecture Notes in Computer Science, Springer, 2017, pp. 251–258. doi:10.1007/978-3-319-70682-5_16.

[11] R. Agrawal, R. Srikant, On integrating catalogs, in: Proceedings of the 10th International Conference on World Wide Web, WWW '01, ACM, New York, NY, USA, 2001, pp. 603–612. doi:10.1145/371920.372163.

[12] S. Melnik, H. Garcia-Molina, E. Rahm, Similarity flooding: A versatile graph matching algorithm and its application to schema matching, in: Proceedings of the 18th International Conference on Data Engineering, San Jose, CA, USA, February 26 - March 1, 2002, 2002, pp. 117–128.

[13] M. Skinner, S. Kallumadi, E-commerce query classification using product taxonomy mapping: A transfer learning approach, in: Proceedings of the SIGIR 2019 Workshop on eCommerce, Vol. 2410 of CEUR Workshop Proceedings, CEUR-WS.org, 2019.

[14] Y. Bengio, R. Ducharme, P. Vincent, C. Janvin, A neural probabilistic language model, J. Mach. Learn. Res. 3 (2003) 1137–1155.

[15] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, arXiv preprint arXiv:1607.04606 (2016).

[16] F. K. Došilović, M. Brčić, N. Hlupić, Explainable artificial intelligence: A survey, in: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2018, pp. 0210–0215.

[17] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, T. Mikolov, Learning word vectors for 157 languages, in: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), 2018.