Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 176 (2020) 723–732

Procedia Computer Science

www.elsevier.com/locate/procedia

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

Contextual ontology for tonality assessment

Wojciech Waloszek[a]*, Nina Rizun[b]

[a]*Faculty of Electronic, Telecommunication and Informatics, Gdansk University of Technology, Narutowicza 11/12, 80-233 Gdansk, Poland*
[b]*Faculty of Management and Economics, Gdansk University of Technology, Narutowicza 11/12, 80-233 Gdansk, Poland*

Abstract

In the paper we discuss the possibilities of using hierarchical contextual ontologies for supporting sentiment classification tasks. The discussion focuses on two important research hypotheses: (1) whether it is possible to construct such an ontology from a corpus of textual document, and (2) whether it is possible and beneficial to use inferencing from this ontology to support the process of sentiment classification. To support the first hypothesis we present a method of extraction of hierarchy of contexts from a set of textual documents and encoding this hierarchy into a multi-level contextual ontology. To support the second hypothesis, we present a method of reasoning from the ontology, and results of experimental verification, which show that use of this reasoning method can increase the accuracy of sentiment classification for longer text documents.

Keywords: knowledge bases; contexts; sentiment classification; topic modelling; hierarchical sentiment dictionary

1. Introduction

Recent years have seen massive bloom of the sentiment analysis, field of study connected with assessing tonality of textual information [1]. Sentiment classification proved itself as an important method, uses in practice for supporting business decisions [2,3], ranking products and sellers [3,4], identifying clients satisfaction and their suggestions [5-8], products and clients classification [9,10], service quality assessment [11,12] or creating strategies

* Corresponding author: Wojciech Waloszek. Tel.: +48583472475 ; fax: +48583472727 .
E-mail address: wowal@eti.pg.edu.pl

10.1016/j.procs.2020.09.045

of stock market investments [13]. Within sentiment analysis advanced method of text mining are used. One of the most important tasks in the field is building a sentiment dictionary. Sentiment dictionaries hold information about tonality of words or expressions, which can then be used to estimate the sentiment of longer texts.

Building such a dictionary is often a major challenge for sentiment classification tasks. Some of the more advanced approaches take into consideration the fact that specific words (like *sharp*) can have differing tonality in various contexts (very positive in the context of knives, and very negative in the context of children toys [1]). As a result, for the domains where occurrences of various contexts are likely, contextual dictionaries are used [14, 15].

The second field of study we draw our inspiration from is semantic knowledge management, basing firmly on the fundamental ideas of Semantic Web [16]. In this field the stress is put on creating white-box specifications of conceptualizations that assume the form of formal ontologies [17]. In the case of more complicated ontologies, there are methods of dividing them into modules (which often represent contexts). Information is usually managed in the form of a knowledge graph [18] in which individual objects, connected with relationships (thus graph), are assigned various concepts.

Knowledge graphs, as it was their original Semantic Web purpose, are commonly used to manage large collection of documents (information objects). These documents frequently have textual character, and thus text analysis methods are often used to support the process of building or querying such a graph [18]. However, the words are not usually considered information objects by themselves, and their use is constrained to label the elements of knowledge graphs.

In this paper we propose to use methods from the field of Semantic Web to manage knowledge about sentiment dictionaries. Our postulate is to treat dictionary elements (words and bigrams) as first-class citizens of an ontology to transcend their usual use as simple labels. Moreover, we propose here to employ a contextual knowledge base, to check if use of contexts in semantic knowledge management can also be beneficial in text analysis.

Specifically we formulate the hypotheses that (H1) it is possible to construct a contextual ontology, whose domain embraces words used in a specific domain of interest, from a corpus of textual document, and (H2) it is possible and beneficial to use inferencing from this ontology to support the process of sentiment classification. What we focus therefore is the *relative* performance of the methods with the stress put on use of a contextual ontology.

The rest of the paper is devoted to discuss these hypotheses. To support the first hypothesis we present a method of automated extraction of hierarchy of contexts from a set of textual documents and encoding this hierarchy into a multi-level contextual ontology. To support the second hypothesis we present a method of reasoning from the knowledge base, and results of its experimental verification which show that use of this reasoning method can increase the accuracy of sentiment classification for longer text documents.

The subsequent Sections are organized as follows. In Section 2 we describe the idea of using contextual ontology for sentiment dictionaries, in Section 3 we describe the procedure of building the ontology, Section 4 contains the results of experimental evaluation, and Section 5 discusses the related work.

2. Contextual ontology for sentiment dictionary

In the field of Semantic Web, ontologies are usually built over a structure $S = (\mathbf{C}, \mathbf{R}, \mathbf{A})$ which is called a *signature* and represents a vocabulary. Notions in **C** represent *concepts*, in **R** *roles* (or properties), and in **A** individual objects (*individuals*).

The ontology itself consists of assertions and axioms. Assertions are usually of the form $C(a)$, $C \in \mathbf{C}$, $a \in \mathbf{A}$ which assigns the individual a to a concept C, or $R(a, b)$, $R \in \mathbf{R}$, a, $b \in \mathbf{A}$ which relate two individuals with the role (property) R. Axioms in turn, allow for expressing interrelationships between concepts, like subsumption or disjointness. An example of an axiom is $\mathit{MedicalProfessional} \sqsubseteq \mathit{Professional}$. Existence of axioms allows for reasoning, so from the ontology which contains this axiom and the assertion $\mathit{MedicalProfessional}(\mathit{johnSmith})$ we can infer that it is true that $\mathit{Professional}(\mathit{johnSmith})$.

Contextual ontologies are relatively newer field of study. They usually follow the idea that the ontology is divided into smaller pieces called *modules* or *contexts*. Throughout these contexts the notions conveyed by concepts might change their meaning, especially if the concepts themselves are highly contextual (like *Neighbor* for countries or *Winner* for a match). The axioms and assertions are typically placed within such contexts, so the same assertion might be entailed by some contexts and not entailed by others.

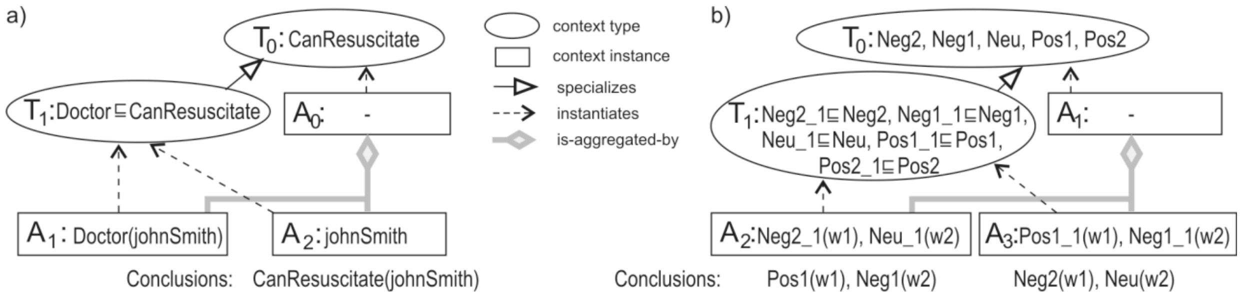

Fig. 1. (a) an example of a SIM ontology; (b) use of SIM ontology to represent sentiment dictionary.

2.1. SIM ontology

SIM is a model for contextual ontologies proposed in [19]. This is our model of choice, because of its innately hierarchical nature put together with a set of inference rules which enable flow of conclusions between the portions of the ontology (modules/contexts).

SIM ontology is organized into modules, which are of two kinds: terminological ones, which contain axioms thus introducing concepts and relationships between them, and assertional ones, which contain assertions thus assigning individual objects to concepts. Both terminological modules, and assertional modules, form their own hierarchy (of inheritance and aggregation resp.), and every assertional module has to *instantiate* one of terminological modules (meaning that it inherits all the terminology from this specific module). Terminological modules are called *context types*, and assertional ones *context instances*. A *context instance* together with the *context type* it instantiates form a *context*.

Contexts are therefore arranged hierarchically, and the assumption here is that contexts higher in the hierarchy are more general, while contexts lower in the hierarchy are more specific. The inference rules for concepts (SIM also defines rules for roles, but they are not relevant in the scope of this paper) state that all the conclusions flow down the hierarchy of context instances, and those conclusions that fit in the concept vocabulary of the higher context instances also flow up (more details can be found in [19]).

An example of reasoning with such rules is presented in Fig. 1a. Context type T_0 introduces a single concept *CanResuscitate*. In the context A_1 we learn that *johnSmith* is a doctor, but this conclusion cannot flow directly to A_0, since *Doctor* is not in its vocabulary. Instead, only the conclusion that *johnSmith* can resuscitate flows freely and reaches A_2. This accounts for the situations where A_1 and A_2 represent different countries which might not respect their medical diplomas (however, of course, people retain their abilities while traveling among countries).

2.2. Use of SIM ontology for representing sentiment dictionary

The idea of using SIM ontology to represent a hierarchy of contexts bases profoundly on the rule of flow illustrated in Fig 1a. Just like the contexts A_1 and A_2 could have their own doctors, contexts in text corpora can assign their own tonalities to specific words (like in the example with *sharp* in the contexts of knives and children toys) or, more generally, *n*-grams. However, to make the full use of conclusion flows, we would like to enrich the reasoning with the flow of information about these different tonalities.

This idea is illustrated in Fig. 1b. In the ontology for representing sentiment dictionary the individual objects are *n*-grams (here *w1* and *w2*). These *n*-grams can be assigned varying tonalities in different contexts (represented by context instances A_1 and A_2 in the SIM ontology). These tonalities are expressed by concepts *Neg2_1*, *Neg1_1*, *Neu_1*, *Pos1_1*, *Pos2_1*, where the digit before the underscore represents the strength of tonality (*Neg2_1* is more negative than *Neg1_1*) and the digit after the underscore the level in the hierarchy (0 when omitted). This, in connection with the rules of conclusion flow, gives us the enriched information in both the contexts A_1 and A_2, for instance for the *n*-gram *w1* in A_1 we can state that it is strongly negative in our context but slightly positive in one of the other context of our domain of interest.

This enriched information can be a basis for assigning a numerical value for tonality of a specific *n*-gram *w* in a specific context *A*. For that it suffices to establish a function *weight*: $\mathbf{C} \rightarrow [-x, x]$ which returns a number for each of the concepts in the ontology, and to sum the value for this function for all the concepts of *w* in *A*. Continuing our example we can notice that it probably makes more sense to assign higher absolute numbers to concepts introduced lower in the hierarchy. Therefore, for the *n*-gram *w1* in A_1, assuming *weight*(*Neg2_1*) = –2 and *weight*(*Pos1*) = 0.5, we can assign it the number of 2 + 0.5 = –1.5.

The proposed model is therefore highly elastic, and has several degrees of freedom: the number of tones *k* (here 2), the number *l* of hierarchy levels (here 1), the numbers of context instances at each level of the hierarchy, and finally the function *weight* for assigning weights to the concepts.

3. Building the contextual ontology

In this Section we introduce the procedure we used for building a contextual ontology representing a sentiment dictionary. For the experimental evaluation of the approach proposed here we used the set of about 8000 Polish-language movie reviews acquired from the Internet. Each of the reviews was accompanied by a numerical rating in scale of 0-10 (stars). The rating was given by the author of the review.

The choice of movie reviews was not incidental. Our assumption was that character of the text corpus should be specific, as we assume that the ontology-assisted method is best suited for longer texts containing contextually varied contents. Movie reviews have both the features.

Due to the assumption we decided to analyze the texts at the level of single paragraphs. These paragraphs represent for us indivisible units that can be assigned to specific contexts basing on their contents.

The procedure we executed included the following steps:

1. Establishing the hierarchy of contexts with use of unsupervised learning,
2. Building the contextual sentiment dictionary.
3. Constructing and filling the knowledge base, and determining *weight* function.

We elaborate upon these steps and their results in the subsequent subsections.

3.1. Establishing the hierarchy of contexts

To establish the hierarchy of contexts we use algorithms well-suited for discovering latent semantic relations, which are hidden inside the documents in the corpus. These relations are used to identify the document's context as the set of topics and to group the documents based on their semantic proximity correspondingly. For this task we use a fusion of Latent Semantic Analysis (LSA) [20, 21] and Latent Dirichlet Allocation (LDA) [22-24] in the style described in more details in [25].

Latent Dirichlet Allocation (LDA) is a generative probabilistic graphical model based on a three-level hierarchical Bayesian modeling approach. In the LDA model, each text is generated independently, by randomly selecting its topic distribution, then randomly selecting a topic from the distribution, and finally randomly selecting a word from the distribution of words in the selected topic.

In the classic LDA model, the number of topics is fixed and initially set by the parameter *t*. However, the quality of topic models can be evaluated with use of perplexity index [22]. It can be used to discover the optimal number of topics, by picking the value of *t* whose further tuning does not significantly change the value of the index.

Latent Semantic Analysis (LSA), also known as Latent Semantic Indexing (LSI), strives to determine the degree of closeness of documents (in our case paragraphs) and visualizing it in a space of a lower dimension by identifying and interpreting hidden semantic relations existing between them. The most well-known version of LSA is based on the algorithm of singular value decomposition of a term-document matrix.

Both of the mentioned have their own limitations. Effective application of LSA [26] requires that texts in the corpus should have the same style of writing, each document be focused on one topic, and words should have a high probability of belonging to one topic but low probability of belonging to other topics (these requirements are unrealistic in the light of our previous assumptions). In turn, the main limitation of LDA is that the number of topics that gives the optimal value of the perplexity index does not guarantee that a particular document can be clearly included within a specific topic [27].

Due to the limitations we decided to use the combination of both the algorithms. In the first step the number of topics *n* is determined by using LDA method and perplexity index. In the second step LSA is executed against the corpus and its results are clustered with use of *k*-means for the previously determined number of clusters.

Each of the clusters/topics (for the brevity we can also call LDA topics clusters) is now treated as a context, and paragraphs of texts are assigned to a context basing on the results of both the algorithms. If LSA and LDA clusters disagree, the LDA probability is decisive, with it being larger than 0.7 means that LDA cluster is assigned, while its lower value means that LSA cluster should be used.

The procedure described above gives us one level of contextual hierarchy. To obtain the deeper hierarchical division of the corpus into context the same algorithm should be performed recursively.

3.2. Establishing the hierarchy of contexts - results

In our experiment we decided to separate from our learning set two subsets: of 2500 reviews with the rating of above 5 (subjectively positive corpora sample) and of 2500 reviews with the rating below 5 (subjectively negative corpora sample). For both of these samples we have performed the contextual division obtaining two hierarchies of contexts. Owing to use of LSA we were also able to associate the contexts in the hierarchy with one or two most distinctive term included in them.

In the first iteration we performed one level contextualization, obtaining 5 and 4 contexts for the two corpora respectively. We decide that to test the method more thoroughly the number of context should be increased, so we proceeded to building a two-level hierarchy.

For the positive corpora sample we obtained 5 first level contexts, and 4 second level in each:

- *Protagonist* (2nd level *Actor/Play*, *Story/Movie*, *Picture/Scene*, *Director/Creator*),
- *Director* (*Movie/Director*, *Scene/Story*, *Style*, *Creator/Author*),
- *Script* (*Movie/Director*, *Story/Protagonist*, *Author/Creator*, *Role/Actor*),
- *Plot* (*Movie/Effects*, *Portrait/Picture*, *Director/Production*, *Script/Story*),
- *Spectator* (*Movie/Aspect*, *Protagonist/Fan*, *Role/Person*, *Scene/Director*)

For the negative corpora sample we obtained 4 first level contexts, and 3 second level in each:

- *Protagonist* (2nd level *Action/Story*, *Director/Theater*, *Scene/Actor*),
- *Actor* (*Protagonist/Image*, *Role/Scene*, *Script/Story*),
- *Creator* (*Protagonist/Scene*, *Movie/Script*, *Picture/Actor*),
- *Plot* (*Story/ Protagonist*, *Director/Picture*, *Creator/Movie*)

3.3. Building the contextual sentiment dictionary

The contextual dictionary has been built in several steps. First, the common bigrams were generated for each of the contexts. The bigrams have been generated from the set *S* unigrams (terms) that have been determined in the previous phase as characteristic for subsequent contexts. All bigrams have been assigned sentiment score with use of Relevancy Frequency measure applied to the classes of *truly positive* (with the rating 9 or 10) reviews and *truly negative* (with the rating 1 or 2) reviews. The measure was calculated with the formula (*C* is truly positive or truly negative, *a* is the number of documents within *C* and containing this bigram, and *b* is the number of documents within the class opposite to *C* and containing this bigram):

$$RF_C = log_2\left(2 + \frac{a}{max(1,b)}\right) \tag{1}$$

This procedure allowed us to obtain the dictionary **DP** of 19163 bigrams and their weights for the positive corpora and the dictionary **DN** of 12227 bigrams for the negative corpora. Within **DP** predominance of positive and neutral bigrams could be noticed, while in **DN** the distribution of bigrams was much more uniform (see Tab. 1).

Table 1. Distribution of positive, neutral, and negative bigrams in sentiment dictionaries.

Dictionary	Positive bigrams	Neutral bigrams	Negative bigrams
DP	43.70%	46.30%	9.91%
DN	20.75%	37.53%	41.72%

3.4. Constructing and filling the SIM ontology

According to the approach described in Section 2.2, the building of the contextual ontologies was relatively straightforward. Each of the ontologies (one for the positive, one for the negative corpora), had two levels of contexts, and thus three context types. The number of context instances was equal to the number of contexts (plus one top level context instance).

In the first iteration (ontologies $\mathbf{O}_{1P}$ and $\mathbf{O}_{1N}$ for the dictionaries **DP** and **DN** respectively) we decided to settle on three tones of positive and negative tonality (therefore creating the concepts of the pattern [*Pos/Neg*][1..3]_[0..2] and [*Neu*]_[0..2], jointly creating the set $\mathbf{C}_{3,2}$). In the further part of the experiment (see Section 4.2) we proceeded to four tones, changing the ontologies to $\mathbf{O}_{2P}$ and $\mathbf{O}_{2N}$ with the set of concepts $\mathbf{C}_{4,2}$.

The most complicated part of this step was determining the exact form of the *weight* function. We have done this by training a linear regression model in order to reflect most closely with *weight* function the original weights contained within a dictionary. The vector of $weight(C)$, $C \in \mathbf{C}_{3,2}$ (or $\mathbf{C}_{4,2}$ in the further part of the experiment), was in fact the vector of regression coefficients being learnt in the process, while the input variables $x_C \in \{0,1\}$ represent the fact of a specific bigram being an instance of a concept C (1 when yes, 0 otherwise).

4. Experimental evaluation

For the purposes of evaluation a test set of movie reviews has been prepared containing 1500 of subjectively positive (rating >5) reviews and 1500 subjectively negative reviews. Each of those sets has been then further divided into three subclasses:

- *HP* (Highly Positive, rating 8-10), *QP* (Quite Positive, rating 6-7), *RP* (Rather Positive, rating 5),
- *RN* (Rather Negative, rating 4), *OP* (Obviously Negative, rating 2-3), *AN* (Absolutely Negative, rating 0-1).

The classification procedure was based on calculating polarity scores for bigrams contained in each review. Polarity scores were calculated on the basis of contextual sentiment dictionary. Contexts were determined for each paragraph in a review *T*. The outline of the procedure is presented below:

1. For each paragraph *p* determine its context *c*, by placing *p* in the space created by LSA in the first phase,
2. For each bigram *b* found in *p* calculate its sentiment score:
   a. in the ontology **O** find the context instance *A* which represents the context *c*,
   b. in the ontology **O** find the individual *a* which represents the bigram *b*,
   c. sum all the $weight(C)$, $C \in \mathbf{C}$ where it is true that $C(a)$ in *A*.
3. Sum the sentiment scores for all bigrams found in all paragraphs.
4. Compare the score with the score for classes in the training set to estimate the final class.

The third step seems straightforward, however, following [28], we decided to use a modified sum here, to account for relatively smaller number of negative bigrams (see Table 1). Therefore, the final summed score was calculated as follows (w_i is a polarity score of *i*-th bigram, N_{pos}, N_{neu}, N_{neg} are number of bigrams of particular tonality, and k_{neg} is the coefficient, compensating the fact of the prevalence of positive vocabulary in the texts):

$$score = \sum_{i=1}^{N_{pos}} w_i^{pos} + \sum_{i=1}^{N_{neu}} w_i^{neu} + k_{neg} \sum_{i=1}^{N_{neg}} w_i^{neg} \quad (2)$$

4.1. First experiment

In the first experiment we decided to make a preliminary comparison of ontology based 7-concept contextual classifier *O7* (we used here the ontologies $\mathbf{O}_{1P}$ and $\mathbf{O}_{1N}$, 3 tonalities, 7 concepts) versus a simple non-contextual baseline. The dictionary for the baseline procedure was not divided into contexts and was prepared as described in Section 3.3 but for the whole set of documents. Therefore, the steps 1 and 2 of the procedure, for the baseline were simply replaced with taking the weight from this non-contextual dictionary.

The results are presented in Tab. 2. It can be seen that *O7* outperformed the simple baseline, therefore confirming its feasibility to be used in classification. This is also supports the hypothesis (H1), as the knowledge base constructed with the described algorithm could be successfully used for processing the corpora.

4.2. Second experiment

In the second experiment we decided to make more thorough comparison with other algorithms and take into consideration also the length of the review being classified. Basing on observations from the preparation to the previous experiment, we also made the decision to increase the number of tonality concepts and use the ontologies $\mathbf{O}_{2P}$ and $\mathbf{O}_{2N}$.

The algorithms used in the experiment were as follows:

- (*O9*) 9-concept contextual classifier working as described in Section 3.3 and using $\mathbf{O}_{2P}$ and $\mathbf{O}_{2N}$ ontologies,
- (*NO*) non-ontological contextual classifier, in which the step 2 has been replaced with taking the appropriate weights directly from the dictionaries **DP** and **DN** (without using the contextual ontology),
- (*SO*) state-of-the-art commercial classifier we obtained from our business project partners *SentiOne*, one of the leading companies in sentiment analysis for Polish language (due to use of this language, comparison with other state-of-the-art classifiers was difficult).

The setting was arranged like this in hope that the comparison between *O9* and *NO* would allow us to assess the influence of using a contextual ontology, while the comparison between *O9* and *SO* would give us some insight about the performance of the algorithm in general.

The results of the experiment are shown in Fig. 3 and Tab. 3. Short reviews were those with the size of maximum 50 characters, medium-sized reviews between 51 and 200 characters and long reviews over 200 characters.

4.3. Discussion of the results

Analysis of the second experiment allows for saying that *SO* excels in classifying short reviews. However, for long reviews *O9* gives better results than any of the other classifiers.

An encouraging observation is that *O9* generally outperforms *NO*. This might indicate that the flow of conclusions between contexts in SIM, and consequent possibility of accounting for tonality of bigrams in several contexts, can be beneficial for sentiment analysis.

In the light of this discussion we can say that the results of the second experiment at least partially support the second hypothesis (*H2*). However, one have to bear in mind that contextual ontologies have some limitations:

- in the case of short reviews, especially negative ones, the algorithm using the ontology gave worse results than the two other algorithm tested; this might also be the effect of specifics of negative reviews in which many positive and negative terms are mixed, and should be investigated further,
- in our setting better results were yielded when the number of sentiment concepts in the ontology was larger (9 in the case of the experiment), this leaves us with the question of how to choose this number, and the observation that when it grows larger, the readability of the ontology decreases.

Table 2. Precision, recall, and accuracy for the first experiment (O7 vs. baseline).

	Positive corpora			Negative corpora		
Method	Precision	Recall	Accuracy	Precision	Recall	Accuracy
O7	62.50%	69.85%	65.49%	67.29%	70.55%	39.67%
baseline	45.60%	45.16%	43.00%	39.32%	39.00%	39.00%

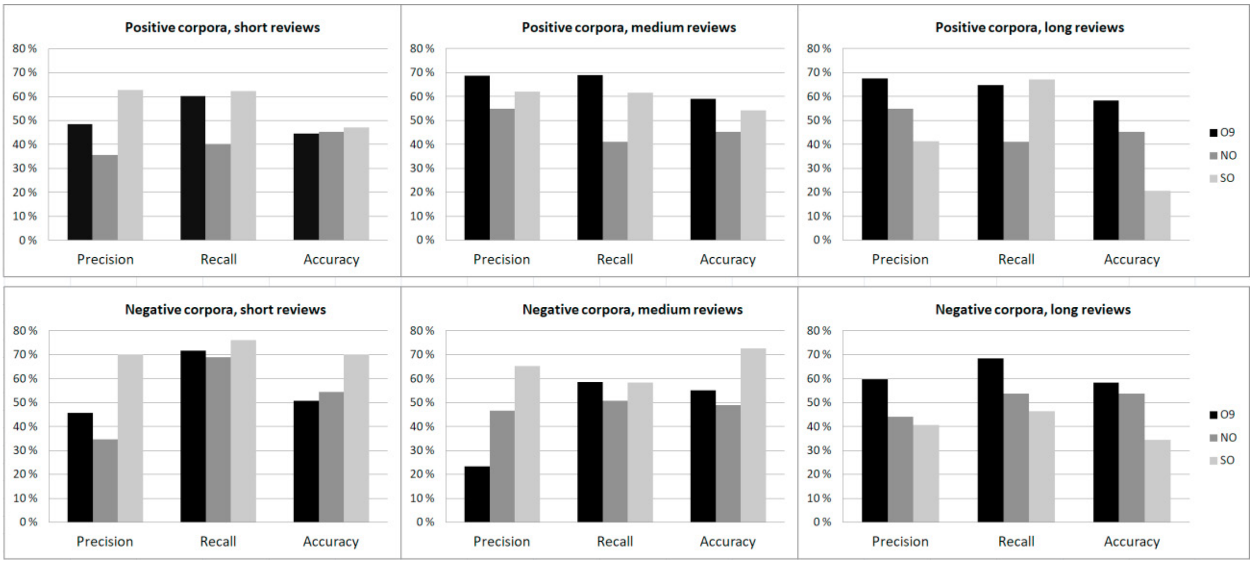

Fig. 2. Charts illustrating the results of the second experiment.

Table 3. Precision, recall, and accuracy for the second experiment (O9 vs. NO vs. SO).

		Positive corpora			Negative corpora		
	Method	Precision	Recall	Accuracy	Precision	Recall	Accuracy
Short reviews	O9	48.48%	60.34%	44.56%	45.61%	71.77%	50.74%
	NO	35.54%	40.26%	45.26%	34.69%	68.86%	54.41%
	SO	62.79%	62.42%	47.02%	70.12%	76.10%	70.00%
Medium reviews	O9	68.61%	68.85%	58.95%	23.37%	58.65%	55.14%
	NO	54.84%	41.06%	45.14%	46.71%	50.76%	48.94%
	SO	62.02%	61.55%	54.24%	65.33%	58.42%	72.58%
Long reviews	O9	67.60%	64.79%	58.29%	59.79%	68.46%	58.44%
	NO	54.84%	41.05%	45.14%	44.19%	53.83%	53.75%
	SO	41.39%	67.05%	20.57%	40.75%	46.40%	34.38%

Nevertheless, our hope for better performance in the case of longer reviews, in which there was a much larger probability of the author referring to different contexts, turned out to be justified. Additionally to the improvement of the classification result, the creation of the ontology containing *n*-grams used within a domain of interest opens significant possibilities. One of them, quite straightforward, is facilitating assessment and development of sentiment dictionaries by human experts. Another is the fact that the contextual ontology, as a knowledge graph itself, can be combined with other knowledge graphs, like, WordNet [29], to be enriched by relationships of homonymy or synonymy in order to use them to refine the text processing even further.

5. Related work

To the best of our knowledge the idea of building a contextual ontology for sentiment dictionaries is novel and not discussed in the literature. However, the work presented in this paper is strongly related to several prominent fields of studies.

Contextual knowledge bases are an important topic of interest in the Semantic Web community. Approaches alternative to SIM have been proposed, including Contextual Knowledge Repositories (CKR) [30] and Description Logics of Contexts (DLC) [31]. These contextual knowledge bases offer similar set of functionalities as SIM, and it could be advantageous to asses also their use in sentiment analysis.

Use of semantic information for supporting text and sentiment analysis is prominent in the works where knowledge graphs are used to leverage processing of corpora of documents [32]. In particular, knowledge graph embeddings play an important role in this process [18]. Especially embeddings to WordNet performed for sentiment analysis [33] carry similarities to our method. While related, it should be however noted, that the approach presented here differs in the aspect of building and using the resulting knowledge graph (arranging *n*-grams in contexts, using conclusion flow), moreover the use of contextual ontology is distinctive.

Finally our work can be perceived as an attempt in increasing the accuracy of sentiment classification. In this aspect it can be compared to [34] or [35]. The accuracies reported by these works (82.9%–87.2%) are higher than those in our experiment. However, it has to be noted that it is difficult to compare these works to our setting (use of Polish language movie reviews, and six target classes in our classifier), and in our work we primarily focused on relative performance of methods which use a contextual ontology. To give more informed opinion about the usefulness of contextual ontological approach in a broader range of situations, further experiments are needed.

6. Conclusions and future work

In this paper we presented the idea of using contextual ontologies to model sentiment dictionaries and in consequence to support the process of sentiment analysis. We proposed a method of building such an ontology (with use of SIM model [19]) on the basis of a corpus of textual documents and using the ontology for estimating tonality of *n*-grams.

The process and the method were illustrated by a case study. In the study we analyzed a set of movie reviews, and such documents tend to have a wide palette of topics and sub-topics. Performed experiments, in which we compared our classifier to non-contextual one, and the one used for commercial sentiment analysis, allowed us to confirm two formulated research hypotheses: about the possibility of constructing a contextual ontology for sentiment dictionary, and about its usefulness in classifying longer documents. Moreover, the resulting ontology might be a useful artifact on its own, which can be used for facilitating development of sentiment dictionaries or combined with other knowledge graphs.

There are two main further directions of work we would like to follow in our research. The first one embraces broadening the range of experiments. It will allow us to more carefully assess the scope of use of our approach, and explain in more details the influence of the exact form of the ontology (like the number of different tonalities) on reasoning and classification. Use of different datasets (also for a corpus in English language) could enable us to compare the results with other studies

The second direction consists in pursuing the new possibilities that open with use of sentiment dictionary expressed as a contextual ontology. Experiments in this area may include enriching the ontology by adding properties (relationships between individuals), to further increase the performance of classification, at first using WordNet to introduce relations between *n*-grams. The fact that we create a contextual ontology gives us also a unique opportunity to integrate the sentiment dictionary with domain knowledge (also expressed by ontological means), by using ontology engineering methods.

References

[1] Liu, B. (2012) “Sentiment Analysis and Opinion Mining”, Morgan & Claypool.

[2] Jahanzeb, J. (2019) "Real-Time Sentiment Analysis On E-Commerce Application," *in IEEE 16th International Conference on Networking, Sensing and Control (ICNSC)*: 391–396.

[3] Luo, Y. (2018), "What Airbnb Reviews can Tell us? An Advanced Latent Aspect Rating Analysis Approach," *Grad. Theses Diss.*: 137

[4] McGlohon, M., Glance, N., Reiter, Z. (2010) "Star quality: Aggregating reviews to rank products and merchants", in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM-2010)*.

[5] Zhao, Y., Xu, X., Wang, M. (2019) "Predicting overall customer satisfaction: Big data evidence from hotel online textual reviews," *Int. J. Hosp. Manag.,* **76**: 111–121.

[6] Gottipati, S., Shankararaman, V., Lin, J. R. (2018) "Text analytics approach to extract course improvement suggestions from students' feedback," Res. *Pract. Technol. Enhanc. Learn*. **13 (1)**

[7] Cleary, P. D., McNeil, B. J. (2018) "Patient Satisfaction as an Indicator of Quality Care", *Inquiry* **25 (1):** 25–36.

[8] Negi S., Daudert T., Buitelaar P. (2019) "SemEval-2019 Task 9 : Suggestion Mining from Online Reviews and Forums," *in Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019*): 877–887

[9] Mtetwa, N., Awukam, A. O., Yousefi, M. (2019) "Feature Extraction and Classification of Movie Reviews," *in Proceedings of the 5th Int. Conf. Soft Comput. Mach. Intell (ISCMI 2018):* 67–71

[10] Othman, R., Belkaroui, R., Faiz, R. (2017) "Extracting Product Features for Opinion Mining Using Public Conversations in Twitter," Procedia Comput. Sci., **112**: 927–935

[11] Al-Hussami, M., Al-Momani, M., Hammad, S., Maharmeh, M., Darawad, M. (2018) "Patients' perception of the quality of nursing care and related hospital services," *Heal. Prim. Care* **1 (2)**: 1–6.

[12] Ojo, A., Rizun, N. (2019) "Enabling Deeper Linguistic-Based Text Analytics—Construct Development for the Criticality of Negative Service Experience," in *IEEE Access* **7**: 169217-169256

[13] Bollen, J., Mao, H., Zeng, X-J. (2011) "Twitter mood predicts the stock market." *Journal of Computational Science* **2 (1)**: 1–8

[14] Ding, X., Liu, B., Yu, P. S. (2008) "A holistic lexicon-based approach to opinion mining", in *Proceedings of the Conference on Web Search and Web Data Mining (WSDM-2008)*.

[15] Lu, Y., Castellanos, M., Dayal, U., Zhai, C.-X. (2011) "Automatic construction of a context-aware sentiment lexicon: an optimization approach", in *Proceedings of the 20th international conference on World Wide Web (WWW-2011)*.

[16] Berners-Lee, T., Hendler, J., Lassila, O. (2001) "The Semantic Web." *Scientific American*, May 2001.

[17] Gruber, T. R. (1993) "A translation approach to portable ontologies." *Knowledge Acquisition*, **5 (2)**:199–220

[18] Nickel, M., Murphy, K., Tresp V., Gabrilovich E. (2015) "A Review of Relational Machine Learning for Knowledge Graphs." *Proceedings of the IEEE*, **104 (1)**: 11–33

[19] Goczyła, K., Waloszek, A., Waloszek, W. (2007) "Contextualization of a DL knowledge base", in: *Proceedings of DL Workshop DL'2007*

[20] Dumais, S.T., Furnas, G.W., Landauer, T.K., Deerwester, S. (1988) "Using latent semantic analysis to improve information retrieval", in *Proceedings of the CHI'88: Conference on Human Factors in Computing*

[21] Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R. (1990) "Indexing by Latent Semantic Analysis." *Journal of the American Society for Information Science* **41 (6)**: 391-407

[22] Blei, D. M., Ng, A. Y., Jordan, M. I. (2003) "Latent dirichlet allocation." *The Journal of Machine Learning Research* **3**: 993–1022

[23] Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., & Den Hartog, D. N. (2018). Text Classification for Organizational Researchers: A Tutorial. Organizational Research Methods, **21(3)**: 766–799

[24] Srinivas, S., & Rajendran, S. (2019). Topic-based knowledge mining of online student reviews for strategic planning in universities. Computers and Industrial Engineering, **128**: 974–984

[25] Rizun, N., Waloszek W. (2018) "Improving the Accuracy in Sentiment Classification in the Light of Modelling the Latent Semantic Relations" *Information* **9 (12)**: 307

[26] Anaya, L.H. (2011) "Comparing Latent Dirichlet Allocation and Latent Semantic Analysis as Classifiers." PhD Thesis

[27] Blei, D. (2012) "Introduction to Probabilistic Topic Models." *Communications of the ACM* **55**: 77–84

[28] Pang, B. (2008) "Opinion Mining and Sentiment Analysis." *Foundations and Trends in Information Retrieval* **2 (1–2)**: 18–22

[29] Miller, G.A. (1995) "WordNet: A Lexical Database for English." *Communications of the ACM* **38 (11)**: 39–41

[30] Homola, M., Serafini, L. (2012) "Contextualized knowledge repositories for the Semantic Web." *Journal of Web Semantics* **12–13**: 64–87

[31] Klarman, S. (2013) "Reasoning with Contexts in Description Logics." Doctoral Thesis, Free University of Amsterdam

[32] Han, X., Liu, Z., Sun, M. (2018) "Neural Knowledge Acquisition via Mutual Attention Between Knowledge Graph and Text", in *Proceedings of Thirty-Second AAAI Conference on Artificial Intelligence AAAI18*

[33] Wallaart, O., Frasincar, F. (2019) "A Hybrid Approach for Aspect-Based Sentiment Analysis Using a Lexicalized Domain Ontology and Attentional Neural Models", in *The Semantic Web, Proceedings of 16th International Conference, ESWC 2019*

[34] Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M. (2011) "Lexicon-based methods for sentiment analysis." *Computational Linguistics* **37 (2)**: 267–307

[35] Lin, C., He, Y. (2009) "Joint sentiment/topic model for sentiment analysis", in: *Proceedings of the 18th ACM Conference on Information and Knowledge Management*