

Ontology Enrichment for Aspect-Oriented Sentiment Analysis with Deep Learning Using Logical Concept Analysis

Tho Quan*, Trung Mai

Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, Vietnam.

* Corresponding author. Email: {qttho, mdtrung}@hcmut.edu.vn

Manuscript submitted September 8, 2019; accepted November 8, 2019.

doi: 10.17706/jcp.14.12.650-661

Abstract: Aspect-oriented sentiment analysis is a problem that have been attracting much attention from the research community. In the past, using ontologies to represent domain knowledge helped increase the performance of this task. Recently, with the rapid development of deep learning techniques, combination of deep learning with ontology has become very promising. A natural approach is to use aspects represented by ontologies to form input vectors s for deep learning models. However, ontologies often only represent concrete concepts, whereas machine learning techniques often focus on latent features. To address this, we propose using Logical Concept Analysis (LCA), an extension of Formal Concept Analysis (FCA), to enrich the ontology by generating more abstract concepts from existing concrete concepts. We have applied this approach to real data and achieved promising results.

Keywords: Aspect-oriented sentiment analysis, ontology enrichment, deep learning, logical concept analysis.

1. Introduction

Opinion, as defined by the Oxford dictionary, is a person's perceptions and thoughts about something, and those feelings and thoughts are not necessarily the truth. Therefore, opinion is always an important reference for decisions made by individuals and organizations. Before the Internet, opinions were consulted through friends, family or consumer opinion surveys of businesses. After that, the explosive growth of the Internet and information changed drastically the way users or businesses consulted and shared opinions and reference information from various sources such as e-commerce websites, Social networks, blogs, online shopping make information more complete, objective, diverse and does not cost too much cost and manpower for the survey process as before. In addition, it also helps users to have a better overview, to help make more accurate decisions.

Therefore, the field of *opinion mining / sentiment analysis* was born and is developing very fast, strong and attracts a lot of attention and research. According to [1], opinions are of great importance in the daily activities of people, because everyone needs to make their decisions more or less based on consulting with other people. There are plenty of practical examples of this, which are businesses that always need to learn about consumer tastes, as well as collect their feedback on their products or services. For an individual who needs to buy a particular product, he or she will need to consult with other consumers.

Studies on this issue are conducted at different levels: *word level* [2], *phrase level* [3], *sentence level* [4] and *document level* [5], [6]. However, if we analyze the feelings of a review on a product, we only know the

general feelings of the user about the product, it is difficult to know whether the user likes or dislikes certain *aspect* on the product. Because a product is positively reviewed does not mean that all features of this product are positive and vice versa.

Therefore, there are much research focusing on identifying sentiment of an object in the text based on the aspects, or *features* of the product. This approach is known as *aspect-oriented sentiment analysis*. Various works have been proposed to solve this problem. *Topic modeling* is often applied in this case [7]. [8] proposed the use of *Probabilistic Latent Semantic Allocation (PLSA)*, while most recent studies are based on *Latent Dirichlet Allocation (LDA)* [9]-[11]. *Insight analysis* [12] improves performance by inferring the characteristics of analytical aspects.

Recently, deep learning has been used widely for sentiment analysis, as discussed subsequently. After some remarkable results of applying deep learning for NLP problems, it now urges the research community to consider combining deep learning with domain knowledge, in order to inherit previous results on this field.

To combine background knowledge with deep learning, a commonly used approach is to represent knowledge by domain ontology. The ontological concepts will then be expressed as feature vectors, which will serve as inputs for deep learning. However, this way does not exploit the *latent features* of knowledge. To solve this problem, we propose an approach of ontology enrichment using Logical Concept Analysis (LCA). This is an extension of the FCA (Formal Concept Analysis) technique, in which logical formulas are used to represent concepts and generate new concepts. We used this approach with actual data sets and obtained initial promising results.

2. Related Works

2.1. Deep Learning for Sentiment Analysis

Recently, with the emergence of *TreeBank*, especially *Stanford Sentiment Treebank* [12], sentiment analysis techniques using deep learning are increasingly being used. Suggested by [13] the idea of feeding a neural network with inputs through multiple-way interactions, parameterized by a tensor have been proposed for relation classification [14]. In the famous Stanford NLP tool [15] widely used by the NLP research community, *RTNN techniques* have been used to apply on the Stanford Sentiment Treebank and gain high performance [16]. The convolution-based technique continues to be developed for sentiment analysis at sentence level [17]. To store relationship dependency in the order in which the features appear, *recurrent neural network (RNN)* systems such as *Long Short Term Memory (LSTM)* have been used in conjunction with convolution to process sentiment analysis for short text [18].

2.2. Ontology Enrichment

Ontology enrichment means updating new semantic words and phrases and their relationship to the ontology. Generally, an ontology has two parts: structure (called *T-Box*) and data (called *A-Box*) [19]. Most of work in ontology enrichments are interested in enriching A-Box. Ontology enrichment has been interested by many research projects and has many different approaches. The typical method is ontology learning [20] and depending on the input data type of the method there will be different approaches.

- Building and enriching ontologies from text, applying natural language analysis techniques such as model-based extraction [21], conceptual clustering [22], etc.
- Enrichment based on dictionary [23], [24]
- Enrichment based on knowledge [25]
- Machine learning and statistics [26]
- Enrichment based on the database [27]

The methods of constructing and enriching ontology are all about identifying words and phrases that capture significant concepts in the corresponding domain and the relationships among them. It is to improve both quality and quantity of the knowledge represented by the ontology. Ontology enrichment using LCA.

2.3. Sentiment Ontology

To give a formal description of aspect-based sentiment analysis, in this section we first present a definition of *Sentiment Ontology*. Then, we formally define the problem of aspect-based sentiment analysis based on Sentiment Ontology

Definition 1 (Sentiment Ontology). A *sentiment ontology* S_0 is a pair of $\{C,R\}$; where $C = (C^A,C^S)$ represents a set of *concepts*, which consists of 2 elements: C^A is a set of *aspect concepts*, and C^S is a set of *sentiment concepts*; $R = (R^T, R^N, R^S)$ represents a set of relationships, which consists of 3 elements: R^N is a set of *non-taxonomic relationships*, R^T is a set of *taxonomic relationships*, R^S is a *sentiment relationships*. Each concept c_i in C represents a set of *objects*, or *instances*, of the same kind, denoted as *instance-of*(c_i). Each relationship $r_i(c_p,c_q)$ in R represents a binary association between concepts c_p and c_q , and the instances of such a relationship, denoted as *instance-of*(r_i), are pairs of (c_p,c_q) concept objects. Especially, an instance $r^s_i(a,s)$ in R^S implies a relationship between an aspect $a \in A$ and a sentiment term $s \in S$.

Example 1. The *Generic Ontology* $G_0 = \{(C^A,C^S), (R^T, R^N, R^S)\}$ is a sentiment ontology where its components are endowed as the following Listing 1.

Listing 1. The Formal Representation of Generic Ontology

```

CA = {"Thing"}
CS = {"Sentiment Term", "Negative Term", "Positive Term"}
RN = {}
RT = {subconcept-of("Positive Term", "Sentiment Term"),
subconcept-of("Negative Term", "Sentiment Term")}
RS = {mentioned-by("Thing", "Sentiment Term")}
instances-of("Positive Term") = {"like"}
instances-of("Negative Term") = {"hate"}
    
```

Generally speaking, G_0 consists of one aspect concept of *Thing*, whose instances can be any real-life concepts. An instance of *Thing* can be *mentioned* by a *Sentiment Term*, which can be either *Positive Term* or *Negative Term*. In this example, G_0 does not present any instance of aspect concept, non-taxonomic nor sentiment relation; whereas two terms "like" and "hate" are example instances of sentiment concepts *Positive Term* and *Negative Term*, respectively.

To graphically visualize an ontology, we rely on the idea of T-Box and A-Box. Basically, a T-Box captures the relations between *concepts* and an A-Box describes *instances* of concepts. Fig. 2 presents the T-Box and A-Box of Generic Ontology G_0 .

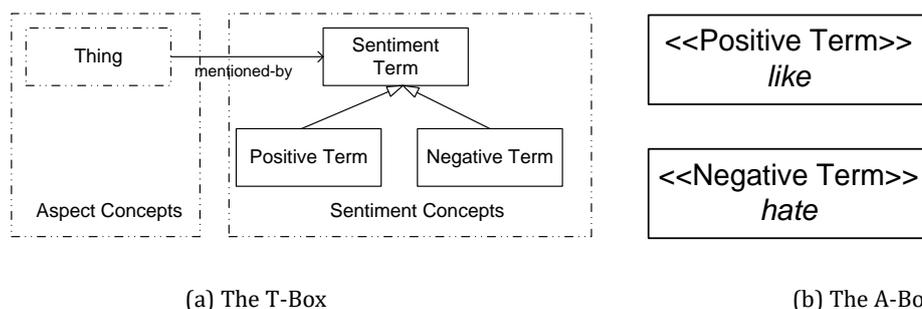
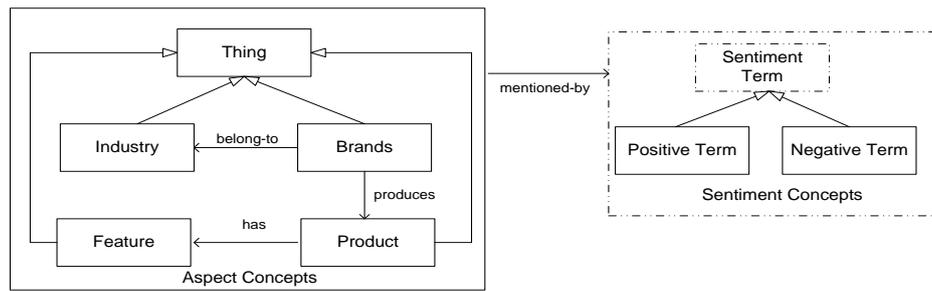
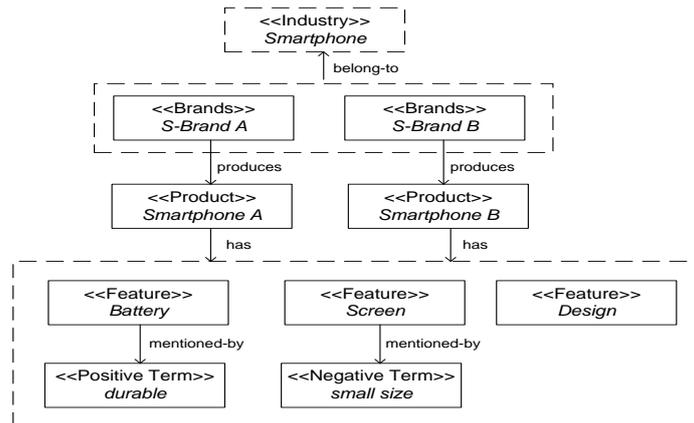


Fig. 1. An example of generic ontology.



(a) The T-Box



(b) The A-Box

Fig. 2. An example of industry ontology.

Example 2. Fig. 3 gives an example of an *Industry Ontology* O_s . The T-Box shows that an *Industry* may have some *Brands*. Each *Brand* can produce many *Products* and each *Product* has various *Features*. All of these concepts are *subconcepts* of *Thing* in the *Generic Ontology*, i.e. they can be mentioned by positive or negative terms.

Listing 2. The Formal Representation of Generic Ontology

```
//Concepts
CA = {"Thing", "Industry", "Brand", "Product", "Feature" }
CS = {"Sentiment Term", "Negative Term", "Positive Term"}

//Relations
RN = {belong-to("Brand,Industry"), produce("Brand","Product"), has("Product","Feature")}
RT = {subconcept-of("Industry","Thing"), subconcept-of("Brand","Thing"), subconcept-of("Product","Thing"),
subconcept-of("Feature","Thing"), subconcept-of("Positive Term","Sentiment Term"), subconcept-of("Negative Term",
"Sentiment Term")}
RS = {mentioned-by("Thing", "Sentiment Term")}

//Concept instances
instances-of("Positive Term") = {"like","durable"}
instances-of("Negative Term") = {"hate","small size"}
instances-of("Industry") = {"Smartphone"}
instances-of("Brands") = {"S-Brand A", "S-Brand B"}
instances-of("Features") = {"Battery", "Screen", "Design"}

//relation instances
instances-of("produces") = {"S-Brand A","Smartphone A"}, {"S-Brand B","Smartphone B"}
instances-of("has") = {"Smartphone A","Battery", "Screen", "Design"}, {"Smartphone A","Battery", "Screen",
"Design"}}
instances-of("mentioned-by") = {"Battery","durable"}, {"Design","small size"}
```

The A-Box of this *Industry Ontology* shows that this ontology describes concepts in the industry of smartphones. There are two brands, namely *S-Brand A* and *S-Brand B*, which produce products of *Smartphone A* and *Smartphone B* respectively. A smartphone product has certain features, such as *Battery* and *Screen*. Though sentiment terms are inherited from the concept of *Thing*, this *Industry Ontology* also introduces some new domain-specific sentiment terms in a manner such that *durable* is a positive term with regard to *Battery* whereas *small size* is a negative term with respect to *Screen*.

The formal information of this *Industry Ontology* can be captured as presented in Listing 2.

3. Sentiment Ontology Enrichment Using LCA

Formal Concept Analysis [28] is a data analysis technique that defines formal concepts of a formal context and builds a corresponding conceptual framework. In the field of data mining, this technique has many rich applications such as biological information [29], [30], social network analysis [31], semantic web search [32], coin tracking, technology orientation [33], understanding the association's law [34], assessing terrorist risk [35], document compatibility [36] and software technology [37]. More specifically, [38] used the FCA technique to analyze the publishing status of articles in this field.

The central role of FCA is to generate a *concept lattice*. To build a more effective and meaningful conceptual lattice, many improvements have been proposed. [39] combines FCA, Chu Space and Domain Theory to analyze dependencies between attributes. [40] suggests using closure operators to analyze dependencies between attributes. The conceptual positioning method is proposed by [41]. In the proposal in [42], an algebraic structure has been proposed to construct a conceptual hierarchical tree and associated ontology from the concept lattice.

However, since the concepts created by FCA are not sufficient to capture the semantics of complex concepts such as *features* of the object, during the study, we propose to use *Logical Concept Analysis* (LCA) [43] as an extension of FCA, to build a concept lattice, where each concept has a characteristic represented by a logical formula. This technique is presented as follows.

Definition 1 (Formal context). A formal context is a triple $K = (G, M, I)$ where G is a set of objects, M is a set of attributes, and I is a set of binary relations between G and M , $I \subseteq G \times M$. An object $g \in G$ has an attribute $m \in M$ is denoted as $(g, m) \in I$ or gIm .

Definition 2 (Derivation operator). In a context (G, M, I) , derivation objects $A \subseteq G$, the set of attributes that are shared by all these objects, and for a set of attributes $B \subseteq M$, the set of objects that share all these attributes and is denoted as follows.

$$\begin{aligned} A \subseteq G, A' &= \{m \in M \mid \forall g \in A, \exists (g, m)\} \\ B \subseteq M, B' &= \{g \in G \mid \forall m \in B, \exists (g, m)\} \end{aligned}$$

Definition 3 (Formal concept). In a context (G, M, I) , (A, B) is a formal concept if and only if $A \subseteq G, B \subseteq M, A = B'$ and $B = A'$. A is called the extent of the formal concept (A, B) and B is called the intent of the formal concept (A, B) .

Definition 4 (Concept relation). In a context (G, M, I) , $(A_1, B_1) \leq (A_2, B_2)$ if and only if $A_1 \subseteq A_2$ or $B_2 \subseteq B_1$. (A_1, B_1) is called the subconcept of the formal concept (A_2, B_2) or (A_2, B_2) is called the superconcept of the formal concept (A_1, B_1) .

Definition 5 (Concept lattice). In a context (G, M, I) , the set of all formal concepts ordered by concept relations is called the concept lattice.

Note that for one concept lattice generated by an FCA, there would be two special concepts called *supremum* S and *infimum* I . or any concept C on the lattice, we always have $C \subseteq S$ and $I \subseteq C$. The supremum and infimum are associated with the concepts denoted by \top and \perp , respectively.

Definition 6 (Logic formal context). A logic formal context is a triple (O, L, i) , where:

- O is a finite set of objects.
- (L, \models) is a lattice of formulas, whose supremum is \vee and infimum is \wedge ; L denotes a logic whose deduction relation is \models , and whose disjunctive and conjunctive operations are respectively \vee and \wedge .
- i is a mapping from O to L that associates to each object a formula that describes the object.

The main difference between LCA and the standard FCA is that the intent is now a formula of the logic L . O is a finite set of objects.

- (L, \models) is a lattice of formulas, whose supremum is \vee and infimum is \wedge .
- i is a mapping from O to L that associates to each object a formula that describes the object.

Definition 7 (Logic derivation operator). Let (O, L, i) be a logic formal context, logic derivation operator is defined as follows.

- $O \subseteq O', O' = \bigvee_{o \in O} i(o)$
- $f \subseteq L, f' = \{o \in O \mid i(o) \models f\}$

Definition 8 (Logic concept). In a context (O, L, i) , a logic concept is a pair $c = (O, f)$ where $O \subseteq O$ and $f \subseteq L$, such that $O' = f$ and $f' = O$.

The set of all concepts that can be built in a context (O, L, i) is denoted by $C(O, L, i)$ and is partially ordered by \leq^C , which is defined as follows.

Definition 9 (Order \leq^C). Let (O_1, f_1) and (O_2, f_2) be in $C(O, L, i)$
 $(O_1, f_1) \leq^C (O_2, f_2) \Leftrightarrow O_1 \subseteq O_2 \vee (f_2 \models f_1)$

Moreover, it is necessary to note that the supremum and infimum of this logic lattice are \top and \perp , respectively, where \top is achieved by applying the widening operator to all logic formulas, f_i , on a lattice, and \perp is the result of applying conjunctive operations on all f_i . Obviously, $\top \models f_i$, and $f_i \models \perp$.

To apply LCA to ontology enrichment, in Table 1, we describe the logical formulas to bring out new concepts from existing ones. Fig. 3 illustrates the concepts formed from the existing ontological concepts from the Sentiment Ontology, represented as a concept lattice.

Table 1. Abstract Operators on Concept Generation Using LCA

No	Logic formulas	Purpose	Examples
1	AND	Provide general attributes of a product line based on specific products	From $C1 = \{iPhone_design, "beautiful", AND "featured border"\}$ and $C2 = \{Samsung_design, "beautiful" AND "no home button"\}$ we can introduce $C3 = \{smartphone_design, "beautiful"\}$

2	OR	Provide product descriptions from attribute descriptions	From $C4 = \{Samsung_price, "expensive"\}$ and $C2 = \{Samsung_design, "beautiful" \text{ AND } "no\ home\ button"\}$ We can introduce new concept with higher abstraction $C5 = \{Samsung, "beautiful" \text{ OR } ("expensive" \text{ AND } "no\ home\ button")\}$
3	XOR	Building product attribute characteristics	From $C6 = \{sentiment_general, "beautiful"\}$ And $C1 = \{iPhone_design, "beautiful" \text{ and } "bordered\ features"\}$ and $C2 = \{Samsung_design, "beautiful" \text{ AND } "no\ home\ button"\}$ We can introduce $C7 = \{smartphone_design_sent, "featured\ border" \text{ OR } "no\ home\ button"\}$

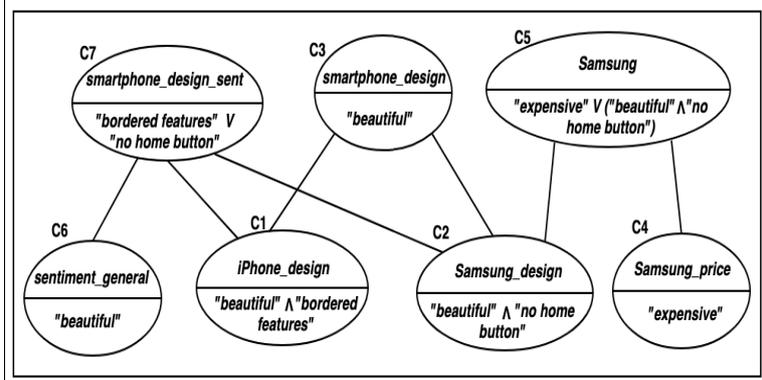


Fig. 3. Generation of new abstract concepts from the existing feature.

4. Experiments

4.1. The Deep Learning Model

Fig. 4 presents an overview of our proposed deep architecture for aspect-level sentiment analysis on news articles. The system includes the following modules.

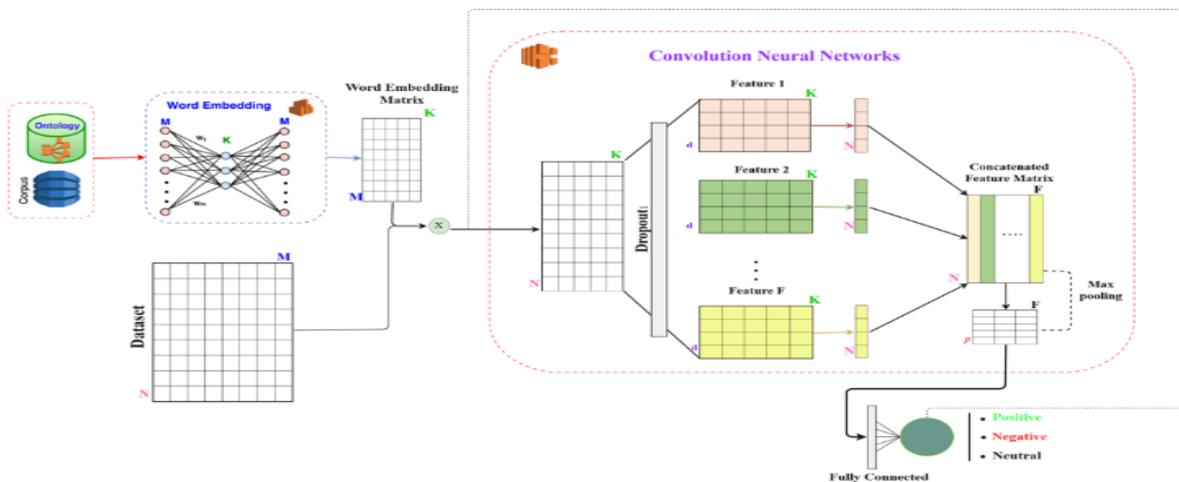


Fig. 4. The overall deep architecture.

Word embedding module: This is a three-layer neural network. The input layer consists of M words where

M is the number of words in the dictionary. The hidden layer consists of K nodes with K being quite small compared to N . The output layer also includes M nodes. This network will be trained from an M -word dictionary. Each word w in the dictionary is passed to the input layer of W as a *one-hot vector* corresponding to w . The W network will be trained to recognize the words w' close to w , to be activated by the corresponding nodes in the output layer. w' words can be determined from a predefined domain ontology built by the expert, or learned from a co-occurrence between (w, w') in a large corpus in the domain being handled. After W is trained, the w_{ij} weights from the input node i to the intermediate node j will form a word embedding matrix $W_{M \times K}$.

Training dataset: A set of collected documents. Each of these documents has been labeled $\{positive, neutral, negative\}$ w.r.t to an object obj that needs to be sentiment-rated. A document with N words will be represented as matrix $D_{N \times K}$, in which the row of i^{th} being the one-hot vector corresponding to the i^{th} word in the document. When performing the matrix multiplication $D \times W$, we can obtain an embedded matrix $E_{N \times K}$. Matrix E will be used as the input for next *Convolution Neural Network* module.

Convolution Neural Network: At this stage, the matrix E will be collapsed with a convolutional window, which is a matrix $F_{d \times K}$. The meaning of this matrix F is to extract an *abstract feature* from the d -gram analysis of the original text. The system will use f matrices $F_{d \times K}$ as an attempt to learn f abstract features. With the convolution between two matrices E and F being a $N \times 1$ column matrix, we will obtain the last matrix $C_{N \times f}$ by concatenating these column matrices together.

Next, matrix C will be fed into a pooling layer by a window $p \times f$. The meaning of this process is to keep the important d -gram sets in consecutive p d -gram. Finally, we obtain the matrix $Q_{q \times f}$ with $q = N/p$.

Finally, a fully connected layer will be implemented to aggregate the results and from there conducted back-propagation process.

4.2. Experimental Results

In order to conduct experiment from real-life data, we obtain real datasets of user reviews on smartphone products from YouNet Media (YNM), a company dedicated for social listening and market intelligence¹. The dataset covers of 32 smartphone brands, 234 products. It consists of 2809, 3098 and 365 *negative*, *neutral* and *positive mentions*, respectively. The experts of YNM also helped us to defined aspect (attribute) of the Smartphone domain, as depicted in Table 3. The aspect terms and sentiments are also defined, as illustrated in Table. Totally, there were 1782 positive term and 1469 negative terms defined for the Smartphone domain.

Table 2. Some Example of Aspect and Sentiment Terms

Attribute	Aspect Term	Sent.Term (Positive)	Sent.Term (Negative)
Design	Design, shape,	attractive, eye-catching,	cloddish, flat
Screen	inch, pixel	sharp, anti-glare	opaque, stained screen
Camera	lens, autofocus	wide, bright	blur, light-interference

We then measured the accuracy of our sentiment analysis approach. We did compare the performance of various sentiment analysis strategies as follows.

- SEN-LCA: we applied our full framework of LCA
- SEN-FCA: we only use FCA, with no extension of LCA
- SEN-NO-FCA: we only use Sentiment Ontology, no FCA techniques applied.

¹ <http://www.younetmedia.com/>

- SVM: we used SVM for sentiment classification, as this technique was employed by various related works.

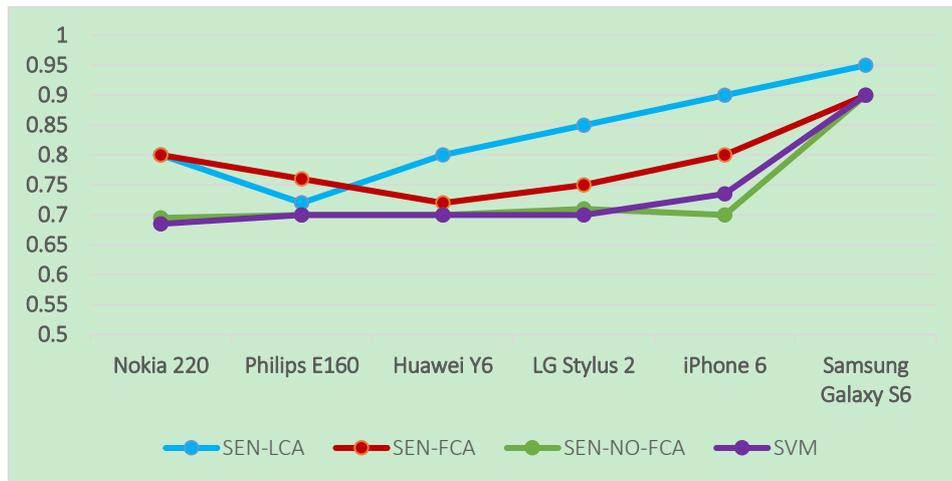


Fig. 5. Accuracy performance of sentiment analysis strategies.

Fig. 5 presents the accuracy percentage when we applied those analysis strategies on the collected datasets. It can be observed that our approach SEN-LCA enjoys the best performance.

5. Conclusion

In this paper, we proposed a solution to solve the problem of ontology enrichment. This approach is intended to apply to a specific problem: aspect-oriented sentiment analysis using deep learning. From the explicit attributes described by the ontology, we propose LCA, an extension of the FCA technique, to further enrich the ontologies and detect potential attributes on them. We have tested this approach on real data sets and obtained encouraging results.

Acknowledgment

This research is funded by Vietnam National University Ho Chi Minh City (VNU-HCM) under Grant B2018-20-07.

References

- [1] Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.
- [2] Liu, B. (2010). Sentiment analysis and subjectivity. *Handbook of Natural Language Processing*, 2, 627-666.
- [3] Kim, J., Li, J. J., & Lee, J. H. (2009). Discovering the discriminative views: Measuring term weights for sentiment analysis. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Vol. 1* (pp. 253-261).
- [4] Wilson, T., Wiebe, J., & Hwa, R. (2004). Just how mad are you? finding strong and weak opinion clauses. *Proceedings of AAAI: Vol. 4* (pp. 761-769).
- [5] Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing: Vol. 10* (pp. 79-86).
- [6] Turney, P. D. (2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (pp. 417-424).

- [7] Qiu, G., Liu, B., Bu, J., & Chen, C. (2011). Opinion word expansion and target extraction through double propagation. *Computational Linguistics*, 9-27.
- [8] Mei, Q., Ling, X., Wondra, M., Su, H., & Zhai, C. (2007). Topic sentiment mixture: modeling facets and opinions in weblogs. *Proceedings of the 16th International Conference on World Wide Web*, 171-180.
- [9] Zhao, W. X., Jiang, J., Yan, H., & Li, X. (2010). Jointly modeling aspects and opinions with a MaxEnt-LDA hybrid. *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing* (pp. 56-65).
- [10] Sauper, C., Haghighi, A., & Barzilay, R. (2011). Content models with attitude. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*.
- [11] Mukherjee, A., & Liu, B. (2012). Aspect extraction through semi-supervised modeling. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers, 1*, 339-348.
- [12] Pang, B., & Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *Proceeding of the 43rd Annual Meeting of the ACL* (pp. 115-124).
- [13] Hinton, G. E. (1990). Mapping part-whole hierarchies into connectionist networks. *Artificial Intelligence*, 46(1-2).
- [14] Jenatton, R., Roux, N. L., Bordes, A., & Obozinski, G. R. (2012). A latent factor model for highly multi-relational data. *Proceeding of Advances in Neural Information Processing Systems* (pp. 3167-3175).
- [15] Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. *Proceeding of ACL (System Demonstrations)* (pp. 55-60).
- [16] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng, A., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. *Proceeding of Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*.
- [17] Kim, Y. (2014). Convolutional neural networks for sentence classification. *CoRRabs/1408.5882*.
- [18] Wang, P., Xu, B., Xu, J., Tian, G., Liu, C. L., & Hao, H. (2016). Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. *Neurocomputing*, 174, 806-814.
- [19] Tung, T. N., Tho, T. Q., & Tuoi, T. P. (2014). Sentiment search: An emerging trend on social media monitoring systems. *ASLIB Journal of Information Management*, 66(5).
- [20] Maedche, A., & Staab, S. (2003). Ontology learning. In S. Staab, & R. Studer (Eds.), *Handbook on Ontologies in Information Systems*. Retrieved from <http://www.aifb.uni-karlsruhe.de/WBS/sst/Research/Publications/handbook-ontology-learning.pdf>
- [21] Morin, E. (1999). Automatic acquisition of semantic relations between terms from technical corpora. *Proceedings of the Fifth Int. Congress on Terminology and Knowledge Engineering (TKE-99)*, TermNetVerlag, Vienna.
- [22] Faure, D., & Poibeau, T. (2000). First experiments of using semantic knowledge learned by ASIUM for information extraction task using INTEX. In S. Staab, A. Maedche, C. Nedellec, & P. Wiemer-Hastings (Eds.). *Proceedings of the Workshop on Ontology Learning, 14th European Conference on Artificial Intelligence ECAI'00*, Berlin, German.
- [23] Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. *Proceedings of the Fourteenth International Conference on Computational Linguistic*, Nantes, France.
- [24] Jannink, J., & Wiederhold, G. (1999). Ontology maintenance with an algebraic methodology: A case study. *Proceedings of AAAI workshop on Ontology Management*.
- [25] Suryanto, H., & Compton, P. (2001). Discovery of ontologies from knowledge bases. In Y. Gil, M. Musen, J. S. Victoria (Eds.), *Proceedings of the First International Conference on Knowledge Capture* (pp. 171-178).

New York, USA: British Columbia Canada, The Association for Computing Machinery.

- [26] Hwang, C. H., (1999). Incompletely and imprecisely speaking: Using dynamic ontologies for representing and retrieving information. *Proceedings of the 6th International Workshop on Knowledge Representation meets Databases (KRDB'99)*, Linköping, Sweden.
- [27] Johannesson, P. (1994). A method for transforming relational schemas into conceptual schemas. *Proceeding of 10th International Conference on Data Engineering* (pp. 115-122). Houston, IEEE Press.
- [28] Ganter, B., & Wille, R. (1999). *Formal Concept Analysis - Mathematical Foundations*, Springer.
- [29] Kaytoue, M., Kuznetsov, S. O., Napoli, A., & Duplessis, S. (2011). Mining gene expression data with pattern structures in formal concept analysis. *Inf. Sci.*, 181(10), 1989-2001.
- [30] Coste, F., Garet, G., Groisillier, A., Nicolas, J., & Tonon, T. (2014). Automated enzyme classification by formal concept analysis. *Proceedings of Formal Concept Analysis - 12th International Conference, ICFCA 2014, Cluj-Napoca, Romania*.
- [31] Obiedkov, S. A. (2013). Modeling ceteris paribus preferences in formal concept analysis. *Proceedings of 11th International Conference Formal Concept Analysis, ICFCA 2013* (pp. 188-202). Dresden, Germany.
- [32] Dubois, D., & Prade, H. (2012). Possibility theory and formal concept analysis: Characterizing independent sub-contexts. *Fuzzy Sets and Systems*, 196, 4-16.
- [33] Lee, C., Jeon, J., & Park, Y. (2011). Monitoring trends of technological changes based on the dynamic patent lattice: A modified formal concept analysis approach. *Technological Forecasting and Social Change*, 78(4), 690-702.
- [34] Du, Y., & Li, H. (2010). Strategy for mining association rules for web pages based on formal concept analysis. *Appl. Soft Comput.*, 10(3), 772-783.
- [35] Elzinga, P., Poelmans, J., Viaene, S., Dedene, G., & Morsing, S. (2010). Terrorist threat assessment with formal concept analysis. *Proceedings of IEEE International Conference on Intelligence and Security Informatics, ISI 2010* (pp. 77-82). Vancouver, BC, Canada.
- [36] Dufour-Lussier, V., Lieber, J., Nauer, E., & Toussaint, Y. (2010). Text adaptation using formal concept analysis. *Proceedings of 18th International Conference on Case-Based Reasoning, Research and Development, ICCBR 2010* (pp. 96-110). Alessandria, Italy.
- [37] Doerfel, S., Jäschke, R., & Stumme, G. (2012). Publication analysis of the formal concept analysis community. *Proceedings of 10th International Conference on Formal Concept Analysis, ICFCA 2012* (pp. 77-95). Leuven, Belgium.
- [38] He, N., Rümmer, P., & Kroening, D. (2011). Test-case generation for embedded simulink via formal concept analysis. *Proceedings of the 48th Design Automation Conference, DAC 2011* (pp. 224-229). San Diego, California, USA.
- [39] Zhang, G. Q. (2003). Chu spaces, concept lattices, and domains. *Electronic Notes in Theoretical Computer Science*, 83, 287-302.
- [40] Ganter, B. (2010). Two basic algorithms in concept analysis. *Proceedings of 8th International Conference Formal Concept Analysis, ICFCA 2010* (pp. 312-340). Agadir, Morocco.
- [41] Poshyvanyk, D., & Marcus, M. (2012). Concept location using formal concept analysis and information retrieval. *ACM Trans. Softw. Eng. Methodol.*, 21(4), 23.
- [42] Wang, L., Liu, X., & Cao, J. (2010). A new algebraic structure for formal concept analysis. *Inf. Sci.*, 180(24), 4865-4876.
- [43] Ferr, S., & Ridoux, O. (2000). A logical generalization of formal concept analysis. *Proceedings of 8th International Conference on Conceptual Structures: Logical, Linguistic, and Computational Issues, ICCS 2000* (pp. 371-384). Darmstadt, Germany.



Tho Quan is an associate professor in the Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology (HCMUT), Vietnam. He received his B.Eng. degree in information technology from HCMUT in 1998 and received Ph.D degree in 2006 from Nanyang Technological University, Singapore. His current research interests include formal methods, program analysis/verification, the semantic web, machine learning/data mining and intelligent systems. Currently, he heads the Department of Software Engineering of the Faculty. He is also serving as the vice dean of Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology.



Trung Mai is currently working as a researcher in the Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology. He received his M.Eng in 2015 in computer science from James Cook University, Australia and B.Eng in 2013 in computer engineering from Auckland University of Technology, New Zealand. His current research interests involve machine learning and deep learning.