A Neural Network Model for Efficient Antonymy-Synonymy Classification by Exploiting Co-occurrence Contexts and Word-Structure Patterns

Van-Tan Bui[1]   Phuong-Thai Nguyen[2]*   Van-Lam Pham[3]   Thanh-Quy Ngo[4]

[1]*University of Economic and Technical Industries, Hanoi, Vietnam*
[2]*VNU University of Engineering and Technology, Hanoi, Vietnam*
[3]*Institute of Linguistics, Vietnam Academy of Social Sciences, Hanoi, Vietnam*
[4]*Thai Nguyen University, Vietnam*
* Corresponding author's Email: thainp@vnu.edu.vn

Abstract: Antonymy and synonymy are basic semantic relations between words. Automatically distinguishing between antonymy and synonymy is an important task in natural language processing. This task is hard because antonyms and synonyms tend to occur in highly similar contexts. Recent studies often focus on exploiting dense-vector representations of words to deal with this problem. In this paper, we present a study on antonymy-synonymy discrimination for the Vietnamese language. We propose a deep neural network model (DVASNet) that can utilize not only embedding representations of words but also co-occurrence contexts and specific patterns of Vietnamese word structure. Our experimental results showed that the proposed method achieved significant improvements in comparison with a number of state-of-the-art methods by 14% to 17% in terms of F1 score for both benchmark datasets.

Keywords: Soft computing, Natural language processing, Machine learning, Semantic relation classification.

1. Introduction

Antonymy and synonymy are lexical semantic relations that play an important role in the organization of lexical databases [1, 2]. The synonymy is a semantic relation between two words that have the same meaning in some or all contexts. In contrast, the antonymy is a semantic relation between two words that have opposite meanings [3]. Distinguishing between antonymy and synonymy is important in natural language processing (NLP) especially for applications such as machine translation, sentiment analysis and information retrieval.

Paradigmatic relations such as synonymy, antonymy and hypernymy are notoriously difficult to distinguish because distributions of context words tend to be very similar across the relations [4, 5]. An interesting example in [6], with regard to the sentence *The boy/girl/person loves/hates the cat*, the nominal co-hyponyms *boy*, *girl* and their hypernym *person* as well as the verbal antonyms *love* and *hate* occur in identical contexts, respectively.

1.1 Previous studies

Prior studies on this problem can be classified into four approaches including pattern-based, vector semantics, unsupervised measure and syntactic structure based. In the pattern-based approach, Lin and co-authors [7] presented a method to retrieve antonyms from distributionally similar words. The method makes use of two patterns *"either X or Y"* and *"from X to Y"*. It relies on a hypothesis that if two words *X* and *Y* appear in one of these patterns, they are very unlikely synonymous. Mohammad and co-authors [8] indicated that Lin's patterns have a low coverage for their antonym set. Turney and co-authors [9] proposed a unified method to analogies, synonyms, antonyms, and associations. The method consists of PairClass, a feature extraction algorithm, and a supervised classification model based on Support Vector Machines. Sabine Schulte im Walde

el al. [6] presented a pattern-based model for distinguishing paradigmatic relations for German words. The first step is to extract lexico-syntactic patterns between word pairs. The second step is to compute pattern frequency vectors. Then a nearest-centroid algorithm is used to distinguish antonyms from synonyms.

In the vector semantics approach, word embedding methods are investigated. Word2Vec, a well-known word representation method proposed by Mikolov and co-authors [10], learns representation by training a classifier to distinguish nearby and far-away words. While Word2Vec tries to capture co-occurrence one window at a time, GloVe proposed by Pennington and co-authors [11] tries to capture the counts of overall statistics. This model learns by constructing a co-occurrence matrix that basically counts how frequently a word appears in contexts. To generate better word embeddings for rare words, Bojanowski and co-authors [12] introduced fastText, a variant of the Word2Vec skip-gram model which builds word vectors as the sum of their constituent character n-gram vectors. Therein, each word is represented by a real-valued feature vector. Two models WE-T and WE-TD to learn word embeddings for the task of identifying antonymy were introduced by Ono and co-authors [13]. Pham and co-authors [14] introduced the multitask Lexical Contrast Model (mLCM), an extension of the skip-gram method that optimizes semantic vectors to predict contexts. Nguyen and co-authors [15] proposed dLCE, a word embedding model integrating external lexical resources. Lexical contrast information is integrated into the skip-gram model to learn word embeddings. This model is effective in predicting degrees of similarity and classifying antonymy and synonymy. ATTRACT–REPEL, another method proposed by Mrksic and co-authors [16], injects lexical contrast information into a pre-trained word embedding model. This method can induce high-quality vectors for lower resource languages by using cross-lingual synonymy and antonymy constraints for creating unified cross-lingual vector spaces.

In the unsupervised approach, distributional measures are used to distinguish synonymy from antonymy in an unsupervised manner. Scheible and co-authors [17] introduced a distributional measure to distinguish adjectival synonyms and antonyms embedded in a vector space. Their distributional measure is based on assumptions that "*the contexts of adjectival synonyms and antonyms are not distributionally similar*", and "*not all word classes are useful for modeling the contextual differences between adjectival synonyms and antonyms*". APAnt,

another distributional measure proposed by Enrico Santus [18] to distinguish antonymy from synonymy, is based on an observation that antonyms are often very similar except in one dimension of meaning. For example, both *giant* and *dwarf* refer to a person, with a head, two legs and two feet, but their size is different. APAnt is an adaption of the Average Precision measure. It compares the N most salient contexts of a pair of antonyms/synonyms. Another hypothesis that this model is based on is that the number of salient contexts shared by synonyms are significantly higher than the number of the ones shared by antonyms.

Recently, a number of neural network models have been used to to distinguish antonymy from other relations [4, 7]. These methods can exploit syntactic structures (in the form of dependency or constituency tree) of co-occurrence contexts of word pairs. For inducing semantic relations between pairs of words (x, y), lexico-syntactic paths that connect x and y in a tree are considered. These studies proved the usefulness of lexico-syntactic information. Fundel and co-authors [19] pointed out that the shortest dependency paths (SDP) between two words in their co-occurrence context are informative for recognizing relations. More specifically, a number of other studies used recurrent neural networks (RNN) to process dependency paths edge-by-edge. Xu and co-authors [20] applied a separate long short term memory network to capture the indicative information in such paths. Good performance achieved by deep learning techniques is due to the ability of automatically learning lexico-syntactic patterns from syntactic parse trees.

1.2 A Summary of our work

Recently, neural network models exploiting lexico-syntactic information of co-occurrence contexts outperform other approaches. However, these methods require parsed corpora which could bear errors of syntactic parsers. Besides, the dependence on syntactic structures could omit useful information (i.e., negation, adverbs, prepositions, etc.) as shown by Can and co-authors [21]. Therefore, Mandar Joshi and co-authors [22] proposed pair2vec model exploiting surface form of contexts instead of dependency paths. From the review of previous studies in subsection 1.1, there are still two open problems as follows:
- Using neural network models to exploit co-occurrence contexts of word pairs without relying on syntactic parse trees: For low-resource languages like Vietnamese, syntactic parsers can only achieve an accuracy much lower than those for English [23].

158

Besides, in Vietnamese interpretation manners are often used for constructing sentences. In that way, related words can be used to semantically complement for each other. Therefore there exist words not belonging to SDP but containing useful information. For example, in a co-occurrence context of *khỏe_mạnh - yếu_đuối* as "*con của ông từ một thanh_niên **khoẻ_mạnh** giờ trở_thành người **yếu_đuối** bệnh_tật*"<his son from a healthy young man becomes a weak and sick person>, *bệnh_tật* is a near-synonym of *yếu_đuối* but it is not in SDP.

- Exploiting specific lexical characteristics of the Vietnamese language: In Vietnamese, compound words take a large proportion of Vietnamese vocabulary. In this paper, we used (_) character to associate syllables of a compound word in Vietnamese. Table 1 shows a number of word length statistics from a popular Vietnamese dictionary conducted by Nguyen and co-authors [24]. Vietnamese word-structure characteristics such as the relationship between syllables of a word and the use of Sino-Vietnamese words [25, 26] have not been exploited in previous studies of the antonymy-synonymy distinction task.

The study closest to our work is proposed by Nguyen and co-authors [27]. This study presents a pattern-based neural network model (AntSynNET) that exploits lexico-syntactic patterns from syntactic parse trees for the task of distinguishing between antonymy and synonymy. The distance between related words along a syntactic path is used as a new feature. In comparison with the AntSynNET model, our method has two important differences. Firstly, instead of using dependency paths as the representation of context, we only use the surface form. Dependency trees are not used to avoid the accumulated error from dependency parsers. Secondly, our model exploits specific Vietnamese language characteristics as mentioned above. Besides, measures of mutual information and similarity between words are also made use of to classify antonymy and synonymy.

The main contributions of this paper are as follows:

Table 1.Word length statistics from a popular Vietnamese dictionary according to the number of syllables [24]

Length	Words	Percentage
1	6303	15.69
2	28,416	70.72
3	2,259	5.62
4	2,784	6.93
5	419	1.04
total	40,181	100.0

- A deep neural network model exploiting features in multi-types to distinguish antonymy from synonymy in Vietnamese.
- A set of Vietnamese word-structure patterns which serve as clues to the determination of relation.
- A new dataset constructed by considering various Vietnamese linguistic criteria for the antonymy-synonymy classification task.

The rest of this paper is structured as follows. Section 2 presents our method. Word structure patterns of antonyms and synonyms are introduced in section 3. Section 4 describes the construction of a novel Vietnamese Dataset for the antonymy-synonymy classification task. Section 5 presents experiments and evaluations. The last section gives a number of conclusions.

2. Proposed method

In this section, we propose a framework that can exploit rich Vietnamese features to classify antonyms and synonyms. We first present Long Short-Term Memory Architecture (LSTM). A bidirectional LSTM is the important module of DVASNet used to encode co-occurrence contexts within a vector representation (section 2.1). Then we describe the architecture of DVASNet which combines multiple features to deal with the problem (section 2.2). The hard encoded feature introduced in section 2.3 is identified as a unified vector that is formed by features pre-extracted such as word structure patterns, mutual information, and word semantic similarity without training. Our framework is demonstrated in Fig. 1.

In our framework, we first extract a set of antonym/synonym pairs from Vietnamese WordNet [28] and Vietnamese Computational Lexicon (VCL - *https://vlsp.hpda.vn/demo/?page=vcl*). Then this set is used to extract a set of triplets (triplet set) from a corpus. A triplet is a tuple <*u, v, contextual words*> co-co-occurred in a sentence. *u* and *v* are synonyms/antonyms. *Contextual words* includes the other words in the sentence (excluding *u* and *v*).
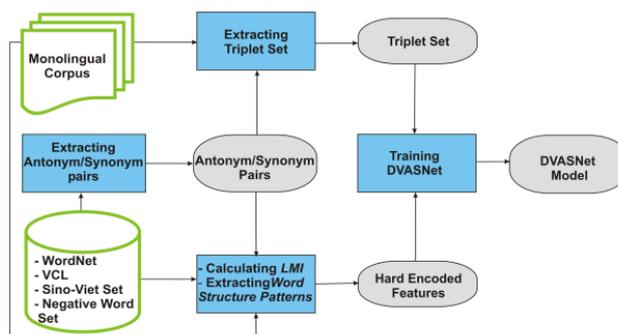

Figure. 1 Overview of the proposed framework

Three features of antonyms and synonyms including lexicographers mutual information, Vietnamese word-structure patterns, and word similarity score are also extracted as hard encoded features. Both the soft/hard encoded features are fed into the DVASNet for training as shown in Fig.1.

2.1 Long Short-Term Memory Architecture

The recurrent neural network (RNN) is suitable for modeling sequential data by nature, as it keeps a hidden state vector h, which changes with input data at each step accordingly. One problem of RNN model is known as gradient vanishing or exploding. Long short term memory (LSTM) units are proposed by Hochreiter and Schmidhuber [29] to overcome this problem. The main idea is to introduce an adaptive gating mechanism, which decides the degree to which LSTM units keep the previous state and memorize the extracted features of the current data input. LSTM models seem well-suited for modeling sequential data by a vector representation. Some recent studies show that LSTM is used effectively to the semantic relation recognition problem. Several papers show improved performance using LSTM that process a dependency path edge-by-edge. Xu and co-authors [20] apply a separate long shortterm memory (LSTM) network to each sequence of words, POS tags, dependency labels and WordNet hypernyms along the path. More recently, Nguyen and co-authors [27] exploited a recurrent neural network with long short-term memory units to encode the patterns as vector representations to improve performance for the semantic relation discrimination task. In this study, we hypothesize that lexico-syntactic patterns appeared with antonym/synonym pairs can be represented as a co-occurrence context embedded vector which encoded by a bidirectional long-short term memory model (biLSTM).

The LSTM-based recurrent neural network comprises four components: an input gate i_t, a forget gate f_t, an output gate o_t, and a memory cell c_t. The three adaptive gates i_t, f_t, and o_t depend on the previous state h_{t-1} and the current input x_t (Eq. (1)). An extracted feature vector g_t is also computed, by Eq. (2), serving as the candidate memory cell.

$$i_t = \sigma(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i)$$
$$f_t = \sigma(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f) \qquad (1)$$
$$o_t = \sigma(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o)$$

$$g_t = tanh(W_g \cdot x_t + U_g \cdot h_{t-1} + b_g) \qquad (2)$$

The current memory cell c_t is a combination of the previous cell content c_{t-1} and the candidate content g_t, weighted by the input gate i_t and forget gate f_t, respectively (Eq. (3)).

$$c_t = i_t \otimes g_t + f_t \otimes c_{t-1} \qquad (3)$$

The output of LSTM units is the the recurrent network's hidden state, which is computed by Eq. (4) as follows.

$$h_t = o_t \otimes tanh(c_t) \qquad (4)$$

In the above equations, σ denotes a sigmoid function; $\otimes$ denotes element-wise multiplication.

2.2 The DVASNet Architecture

In this section, we present a model to distinguish antonyms from synonyms for the Vietnamese (DVASNet). The proposed model makes use of a recurrent neural network with LSTM units to encode the context of word pairs. Fig. 2 illustrates the DVASNet mode. Give a triplet <u, v, contextual words>. Each word is represented by a concatenation vector of fastText embedding and part of speech embedding. Contextual words w1:n are fed into the bidirectional LSTM module and the contextual vector defined as the following vector concatenation:

$$\vec{v}_{BiLSTM} = \left[\vec{v}_{lLSTM(w_{1:n})}, \vec{v}_{rLSTM(w_{n:1})} \right] \qquad (5)$$

Where lLSTM/rLSTM represent distinct left-to-right/right-to-left word embeddings of the contextual words. Next, we apply the following non-linear function on the concatenation of the left and right context representations:

$$MLP(\vec{v}_{BiLSTM}) = L_2 \left(ReLU(L_1(\vec{v}_{BiLSTM})) \right) \qquad (6)$$

Where MLP stands for Multi Layer Perceptron, ReLU is the Rectified Linear Unit activation function, and $L_i(x) = W_i x + b_i$ is a fully connected linear operation. MLP aims to learn a combination vector from left/right output vectors of BiLSTM. Let denote $\vec{v}_c$ is the representation vector for contextual words, it can be calculated as follows:

$$\vec{v}_c = MLP(\vec{v}_{BiLSTM}) \qquad (7)$$

Denote $\vec{v}_{soft}$ is the concatenation of $\vec{v}_c$ and representation vectors of $\vec{v}_u$ and $\vec{v}_v$. $\vec{v}_{soft}$ becomes a
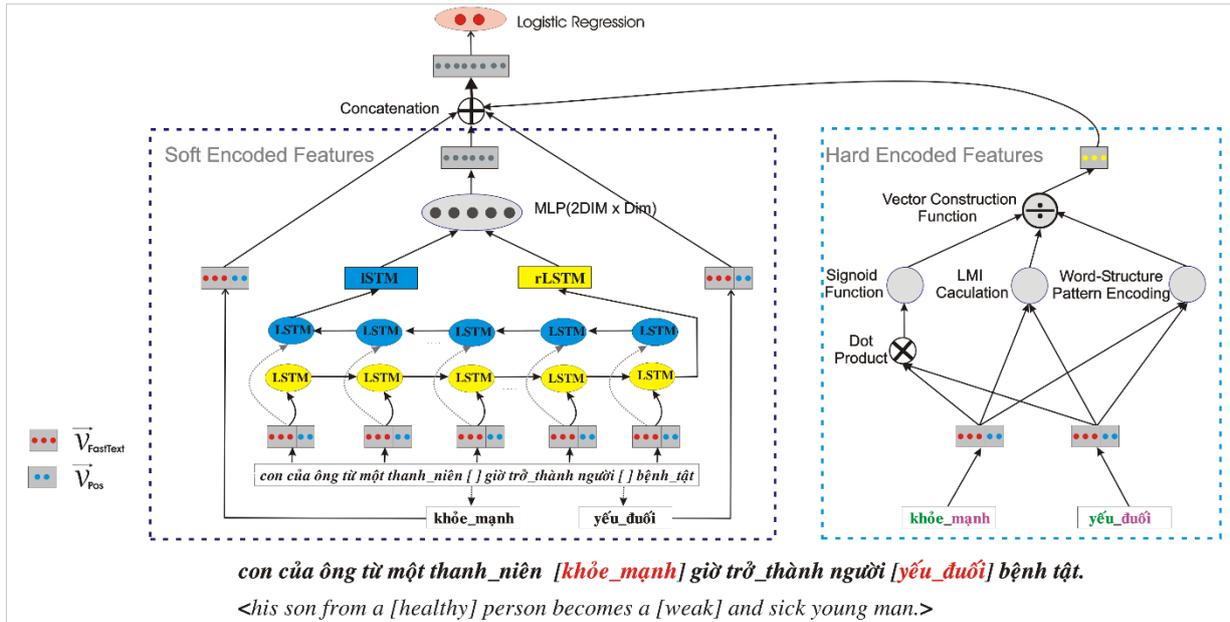
Figure. 2 The DVASNet architecture for Vietnamese antonyms and synonyms classification

soft encoded feature which is learned through by training the neural network:

$$\vec{v}_{soft} = [\vec{v}_c, \vec{v}_v, \vec{v}_u] \qquad (8)$$

Next, we denote the hard encoded feature vector as $\vec{v}_{hard}$.

It is a k-dimensions vector which is generated from three values by vector construction function (VCF) including a dot-product similarity value between two embedded vectors transformed by the sigmoid function (SimScore), an LMI score (LMI), and a word-structure pattern encoded value (WLP). Antonymy word-structure patterns are encoded by value in the range from 0 to 8. In contrary, synonymy word-structure patterns are encoded by value in the range from -7 to 0. Where values are different from 0 representing a corresponding word-structure pattern. Word pairs do not correspond to any pattern assigned a value of 0.

$$\vec{v}_{hard} = VCF(SimScore, LMI, WLP) \qquad (9)$$

Finally, $\vec{v}_{triplet}$ is a unified feature vector which represents a triplet is fed into the logistic regression layer to classify antonyms and synonyms. $\vec{v}_{triplet}$ is the concatenation of $\vec{v}_{hard}$ and $\vec{v}_{soft}$ as follows:

$$\vec{v}_{triplet} = [\vec{v}_{hard}, \vec{v}_{soft}] \qquad (10)$$

2.3 Hard encoded features

2.3.1. Word structure

The Vietnamese language has a number of specific lexical characteristics such as existing the relationship between the components in a word, Sino-Vietnamese words [25, 26]. Therefore, along with encoding lexico-syntactic patterns, we also aim to exploit effectively these characteristics. Since compound words take a large proportion in Vietnamese vocabulary (Table 1), it is interesting to analyse the structure of compound synonyms and antonyms. Linguistic analyses of compound antonyms and synonyms show that semantic relations between syllables are a strong indication of antonymy and synonymy. For example, in many cases, compound antonyms contain syllables that are single antonyms. For instance, the pair of compound antonyms *vui_vẻ*<merry> - *buồn_rầu*<moody> contain syllables *vui*<happy>, *buồn*<sad> that are single antonyms. Similarly, *phụ*<dad> - *cha*<father> and *mẫu*<mom> - *mẹ*<mother> are pairs of single synonyms. Their compounds *cha_mẹ*<parent>, *phụ_mẫu*<parent> are also synonyms. Word structure patterns will be described in section 3.

2.3.2. Mutual information between words

Several prior studies have pointed out that the co-occurrence of an antonym pair is more often than expected by chance. Charles and Miller [1] proposed a co-occurrence hypothesis that antonyms relatively frequently co-occur in the same sentence. However

this hypothesis is also true for other semantic relations such as near synonymy and hypernymy. Therefore, co-occurrence is not a sufficient condition for detecting antonyms, but it is useful. Fellbaum [2], Justeson and co-authors [30] counted the appearance of antonyms in a corpus. Experimental results showed that antonym pairs had significant numbers of co-occurrences. Moreover, by computing Point Mutual Information (PMI) scores of antonym pairs, synonym pairs and random pairs from the Princeton WordNet, the average PMI scores of antonym pairs were significantly higher than that of synonym pairs and random pairs [8]. PMI is biased towards infrequent words. This weakness of PMI may be addressed by using LMI [31] (Eq. (11)).

$$LMI(u,v) = n_{u,v} \times PMI(u,v) \qquad (11)$$

2.3.3. Word semantic similarity

The similarity of two words (word similarity) quantifies the cognitive distance between two words with respect to their type (e.g., *horse* is very similar to *donkey* because they are both equine animals) or function (e.g., *car* is somewhat similar to *motorbike* as both can be used for traffic). The similarity score of a word pair can be exploited as a strengthened feature to determine the semantic relations of this pair. The word similarity can be quantified as the distance between two embedded vectors. In this study, the word similarity obtained by calculating dot product of two embedded vectors. The sigmoid function is used to normalize dot product score to a range of values from 0 to 1.

3. Vietnamese word-structure patterns

By surveying linguistic literatures, we found that *there exist antonym pairs and synonym pairs whose components of one word have a cross semantic relation with components of the other*. Fig 3 represents a visual illustration of this property. In (A-a), two compound antonyms are composed by single antonyms. In (A-b), two compound antonyms are composed by single antonyms and single synonyms. In (S-a) two compound synonyms share one syllable and the other syllables are synonymous. In (S-b) a pair of compound synonyms is composed of two pairs of single synonyms among them one pair of Sino-Vietnamese synonyms.

To find out a specific set of Vietnamese word-structure patterns, we manually analysed a sample of compound antonym pairs and compound synonym pairs extracted from Vietnamese WordNet and VCL.

For each word pair we identified the semantic relation between components of one word with components of the other. As a result, we found 8 word-structure patterns of antonyms and 7 word-structure patterns of synonyms.

In sub-section 5.3.1, we will present a number of statistics showing how popular these word-structure patterns are. In order to do that, we collected two sets of single antonym pairs such as *to*<big> - *nhỏ*<small>, and single synonym pairs such as *buồn*<sad> - *sầu*<sorrowful>. The antonym set consists of 1,200 pairs, the synonym set contains 9,045 pairs. These sets are extracted from the Vietnamese WordNet and the VCL. Furthermore, we manually collect 3,003 Sino-Vietnamese synonym pairs to construct a set of pairs which each comprises a Sino-Vietnamese word and its meaning in Vietnamese (e.g., *tử*<children> - *con*<children>, *mã*<horse>   - *ngựa*<horse>,...). We rely on these sets to determine semantic relation between components of words.

3.1 Word structure patterns of antonyms

In total, we found 8 word-structure patterns of antonyms as follows.
u₁_* - v₁_*, where u_1 is an antonym of v_1, the remaining components do not hold any relation: *ác_độc*<sinister> - *hiền_từ*<gentle>.
u₁_u₂ - v₁_v₂: where u_1 is an antonym of v_1, u_2 is an antonym of v_2:
cao_sang<of high rank> - *thấp_hèn*<lowly>.
u₁_u₂ - v₁_v₂, where u_1 is an antonym of v_1, u_2 is a synonym of v_2: *hữu_ích*<useful>-*vô_dụng*<useless>.
u₁_x - v₁_x, where u_1 is an antonym of v_1, the remaining components are identical: *ác_tính*<malignant> - *lành_tính*<benign>.
u₁_* - v, where u_1 is an antonym of *v*: *bẩn_thỉu*<dirty> - *sạch*<clean>.
u - NW_u, where *NW* is a word in the negative word set {*vô*, *phi*, *bất*, *chẳng*, *không*}. *NW* reverses the meaning of *u*: *thành_văn*<explicit> - *bất_thành_văn*<implicit>.
u₁_* - NW_u₁, where NW is a word in the negative word set: *minh_bạch*<transparent> - *bất_minh*<suspicious>.
u₁_u₂ - v₁_v₂, where u_1 is a synonym of v_1, u_2 is an antonym of v_2: *bôi_đen*<blacken> - *tô_hồng*<embellish>.

3.2 Word structure patterns of synonyms

In total, we found 7 word-structure patterns of synonyms as follows.
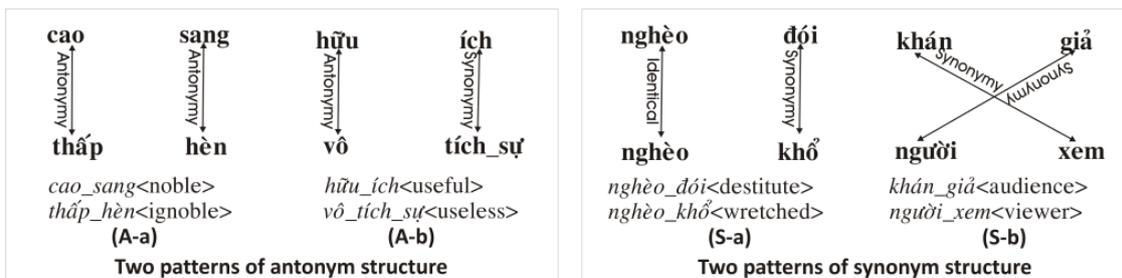**x_* - x_***, where two words share the first

Figure. 3 Illustrations of antonym/synonym structure patterns

component *x*, remaining syllables do not hold any relation: *chật_hẹp*<cramped> - *chật_chội*<cramped>.

u - u_*, where one word is the first component of the other word: *dối*<deceitful> - *dối_trá*<deceitful>.

u - *_u, where one word is the second component of the other word: *hiếm*<rare> - *khan_hiếm*<rare>.

u₁_x - v₁_x, where u_1 is a synonym of v_1, the other syllables are identical: *kính_trọng*<respect> - *tôn_trọng*<respect>.

u₁_* - v₁_*, where u_1 is a synonym of v_1, the remaining components do not hold any relation: *dối_trá*<deceitful> - *lừa_lọc*<deceitful>.

x_y - y_x, where components are in an inverse order: *giảm_sút*<decrease> - *sút_giảm*<decrease>.

u₁_u₂ - v₁_v₂, where u_1 - v_2 and u_2 - v_1 are are Sino-Vietnamese pairs: *thính_giả*<hearer> - *người_nghe*<hearer>.

4. A new Vietnamese dataset for testing

ViCon dataset was introduced in a prior study [32]. It is a reliable Vietnamese dataset of lexical contrast pairs for evaluation of models that distinguish similarity and dissimilarity. The dataset consists of 400 noun pairs, 400 verb pairs, and 600 adjective pairs randomly selected from VCL. Though this dataset was built elaborately and meticulously, it still has a number of disadvantages. Firstly, it does not guarantee that the data set can cover circumstances of antonymy and synonymy according to various criteria in Vietnamese linguistics because word pairs are randomly selected. Secondly, the ratio of synonym/antonym pairs via the part of speech (POS) does not correspond to the natural ratio. Thirdly, several pairs contain words with different POS such as *xuất_hiện*verb<appear> - *thiếu*adjective<absent>, *thanh_danh*noun<repute> - *ô_nhục*adjective<ignoble>.

In this section, we introduce a novel Vietnamese Dataset for the antonymy-synonymy distinction task (ViAS-1000 dataset - https://github.com/BuiVanTan2017/DVASNet).   Antonym/synonym

Table 2. Statistics of the ViAS-1000 dataset

	#Adjective	#Verb	#Noun	Total
Antonyms	250	90	60	400
Synonyms	250	200	150	600

pairs are manually selected from Vietnamese WordNet and VCL. Firstly, we extracted all antonym and synonym pairs belonging to three part-of-speech categories: noun, verb and adjective. Then, 400 antonym pairs and 600 synonym pairs were carefully selected with respect for a number of constraints:

- Words in a pair have the same POS.
- The POS ratio follows the natural ratio. In the Vietnamese vocabulary, the popularity of antonymy and synonymy decreases in the order of adjective, verb, noun. Besides, synonyms are also much more popular than antonyms [25].
- Antonym/synonym pairs cover sufficiently Vietnamese word structures such as single/compound words, subordinating/coordinating words, reduplicative words, reverse words.
- The dataset should contain words in different domains (e.g. sport, emotion, vehicle).

The ViAS-1000 data set contains 400 antonym pairs, and 600 synonym pairs, with the proportion of pairs according to the part of speech as shown in Table 2.

5. Experiments

5.1 Baseline models

Since the proposed model can exploit not only embedded vectors of candidate words but also information about co-occurrence contexts, the usefulness of this information for distinguishing Vietnamese antonymy and synonymy must be demonstrated. Besides, the use of hard encoded features is also evaluated against baselines. Therefore, in our work baseline systems only utilize embedded vectors of candidate words. Word embedding methods selected for baseline systems
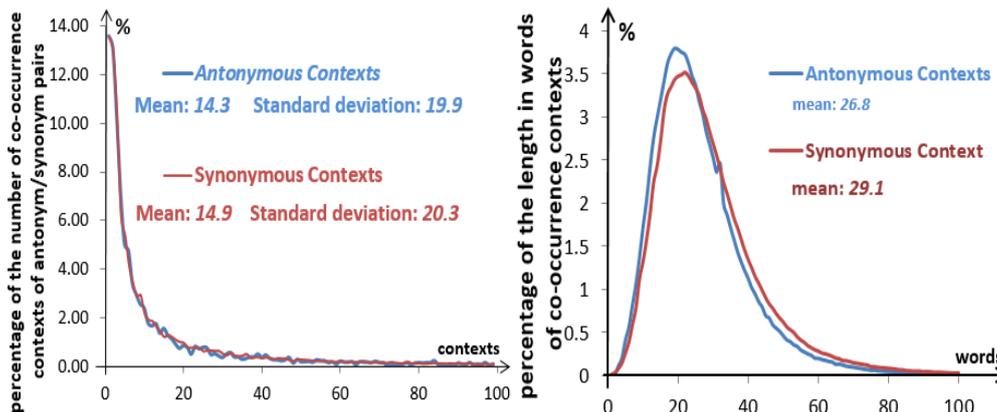
Figure. 4 An illustration of statistical results on triplets

include both general-purpose models (Word2Vec [10], fastText[12], GloVe[11]) and specialized purpose models (ATTRACT–REPEL[16], dLCE[15]). The logistic regression method is used for relation classification.

5.2 Experimental settings

5.2.1. Resources for training

To train embedding models and extract the triplet set, we used a monolingual Vietnamese corpus (Vcorpus - is collected from Vietnamese Wikipedia and https://baomoi.com/). Any words that appear less than 5 times are excluded. We used the Python Vietnamese Toolkit (https://github.com/ trungtv/pyvi) for tokenizing and POS tagging. Vcorpus includes 28,767,455 sentences with 567,473,442 tokens, where 1,603,857 tokens are distinct.

The set of antonym/synonym pairs was extracted from Vietnamese WordNet and VCL. We excluded from this set any pair in the test set (ViCon, ViAS-1000). As a result, the total number of antonym and synonym pairs are 24,347 and 156,847, respectively. To obtain triplets, we pick sentences in the Vcorpus which contain any antonym/synonym pair. Note that words in a pair must have the same POS tag. The extracted triplet set includes 3,809,864 antonym triplets and 23,500,420 synonym triplets. Several examples are presented in Table 3.

To better understanding the extracted triplet set, several statistics on this set are illustrated in Fig. 4. In the left side of this figure, the distribution in the percentage of the number of co-occurrence contexts of antonym/synonym pairs is visualized. This figure shows only co-occurrence frequency up to a threshold of 100. Possible values of frequency are much larger than this threshold, but these values

Table 3. Examples of co-occurrence contexts

*Bà càng **mạnh_mẽ** bao_nhiêu thì con_gái bà lại càng **yếu_đuối** bấy_nhiêu* <the **stronger** she was, the **weaker** her daughter became>.
*người_dân mua đất **hợp_pháp** giờ lại thành **bất_hợp_pháp*** <people bought the **legal** land and now it becomes **illegal**>.
*tôi không **giàu** cũng chẳng **nghèo*** <I am neither **rich** nor **poor**>
*thời gian tới chúng_tôi vẫn xác_định **tiến** chứ không **lùi*** <forthcoming we still determined to move **forward** rather **backward**>.
*cô ấy là người kén **bạn** nên không đàn_đúm như **bạn_bè** cùng trang_lứa* <she is a picky person about **friends** so she is not a reveler as her **schoolmate**>
*lãnh_đạo tỉnh **chủ_trương** phải đẩy_mạnh công_tác đôn_đốc kiểm_tra và giám_sát để sớm đưa những **chính_sách** mới vào cuộc sống* <chief of the province has an **idea** that thrusts supervisory and supervisory activities to bring new **policies** to life soon>

take only a small percentage. 80% of antonym pairs co-occur less than 30 times and 80% of synonym pairs co-occur less than 60 times. However, the standard deviation is high and the frequency range is also large. In the right side of Fig. 4, the distribution in the percentage of the length in words of co-occurrence contexts is visualized. The average lengths of co-occurrence contexts for antonym/synonym pairs are 26.8 and 29.1, respectively.

All relation classification methods based on co-occurrence contexts suffer from data sparseness as they depend on word pair co-occurrences in a corpus Koki Washio and co-authors [33]. According to Zipf's law, the frequency of any word is inversely proportional to its rank in the frequency table. In other words, most words occur very infrequently [34]. Therefore, we cannot observe all co-occurrences of antonym/synonym pairs even with a very large corpus. In this study, although the number of triplets extracted is massive, but only 25% of

antonym pairs appear together in the corpus, this percentage for synonym pairs is 26%.

5.2.2. Parameter settings

In training phase of the DVASNet model, a word is represented by a concatenation vector of a 300-dimensions fastText embedded vector and a 5-dimensions POS embedded vector. The AdaGrad algorithm is used with the number of the epoch set to 20, and the initial learning rate of AdaGrad set to 0.05. When evaluating the DVASNet model, for each word pair we use d co-occurrence contexts to evaluate, the predicted result is the average of candidate values. Through experiments, we choose d=10. For word pairs that do not co-occur in Vcorpus, only embedded vectors and hard encode representation are used.

5.3 Experimental results

5.3.1. Word-structure pattern statistics

To estimate the proportion of synonym/antonym pairs that match word-structure patterns. We conducted automatically analyses on datasets including ViCon [32], ViAS-1000 represented in Section 4, as well as the whole antonym/synonym pairs extracted from Vietnamese WordNet and VCL. Statistical results in Table 4 show that a large proportion of antonym/synonym pairs match word-structure patterns. Furthermore, patterns are more common in synonyms than antonyms.

5.3.2. Mutual information comparison between antonyms and synonyms

We computed the Pointwise Mutual Information and Lexicographers Mutual Information (LMI) [31] on the Vcorpus of 1000 synonym pairs and 1000 antonym pairs which randomly selected from antonym/synonym pairs of ViCon and ViAS-1000. Statistical results are shown in Table 5. These results show that the average PMI score of antonym pairs is 3.6 times higher than that of synonym pairs. Similarly, the average LMI score of antonym pairs is more than 5 times higher than that of synonym pairs.

Table 4. The rate of the word-structure pattern of the antonymy synonymy in Vietnamese

Dataset	#Pairs	#Patterns	(%)
ViCon	1,398	622	44.5
ViAS-1000	1,000	453	45.3
Antonym Pairs	24,347	15,126	62.1
Synonym Pairs	156,847	88,382	56.3

Table 5. The average of PMI/LMI score of antonyms/synonyms pairs

Relation	Average PMI	Average LMI
Antonymy	2.86	72.78
Synonymy	0.79	13.24

Table 6. Performance of the DVASNet model in comparison to the baseline models

Model	ViCon			ViAS-1000		
	P	R	F1	P	R	F1
Word2Vec [10]	0.75	0.62	0.68	0.77	0.47	0.58
GloVe [11]	0.78	0.63	0.70	**0.84**	0.45	0.59
fastText [12]	0.76	0.63	0.69	0.79	0.44	0.57
dLCE[15]	**0.78**	0.67	0.72	0.80	0.55	0.65
Attract-Repel[16]	0.76	0.67	0.71	0.75	0.47	0.58
DVASNet	0.72	0.80	0.76	0.75	0.70	0.72
DVASNet+	0.74	**0.82**	**0.78**	0.79	**0.73**	**0.76**

5.3.3. Antonymy-synonymy classification

We used the F-score (**F1**) as a primary evaluation metric. The F1 is the harmonic mean of precision (**P**) and recall (**R**). In experimental results shown in Table 6, DVASNet labels the case which the model only uses soft encoded features. DVASNet[+] labels the case the model exploiting both soft and hard encoded features. Table 6 shows the significant performance of our models in comparison to the baselines.

Experimental results in Table 6 show that most baseline methods achieved higher precision scores than our method, but recall scores are lower. Since the DVASNet model improves the recall score significantly, the F1-score is also improved in comparison to baseline models. Furthermore, the DVASNet+ model which exploits both soft and hard encoded features consistently achieved a higher performance than DVASNet.

Table 6 also shows that DVASNet achieved performance outperform comparing to baseline models in the R score. This proves that exploiting co-occurrence contexts can increase the ability to recognize both antonyms and synonyms. In other words, DVASNet reduces omitting positive pairs than baseline models which only use distributional semantic features. Further, incorporate hard encoded features with soft encoded features in DVASNet[+] again increased the R score.

6. Conclusion

This paper introduces DVASNet, a deep neural network model that can effectively exploit specific characteristics of the Vietnamese language for the antonymy-synonymy classification task. The model can utilize not only lexico-syntactic information

captured from co-occurrence contexts of word pairs in a corpus, but also word structures and distribution features as well. The word-structure patterns were first exploited as a useful feature to recognize the relations. In addition, the ViAS-1000 dataset was constructed considering various criteria in Vietnamese linguistics for the evaluation of the task. Our proposed model significantly outperformed five baseline methods by 22% to 25% in terms of recall score and from 14% to 17% of the F1 score for benchmark datasets. In the future, we intend to apply the proposed method to discriminate between other semantic relations and to automatically extract antonymy-related pairs for the ontologies and lexical resources construction.

Acknowledgments

References

[1] W.G. Charles and G.A. Miller, "Contexts of antonymous adjectives", *Applied Psycholinguistics*, pp.357-375, 1989.

[2] C. Fellbaum, "Co-Occurrence and Antonymy", *International Journal of Lexicography*, pp.281-303, 1995.

[3] J. Lyons, "Semantics", *Cambridge University Press,* volume 1, 1977.

[4] M. Roth and S.S. im Walde, "Combining word patterns and discourse markers for paradigmatic relation classification", In: *Proc. of NAACL,* pp. 524-530, 2014.

[5] V.T. Bui, P.T, Nguyen, and M.T. Nguyen, "Enhancing performance of lexical entailment recognition for vietnamese based on exploiting lexical structure features", In: *Proc. of KSE*, pp.341-346, 2018.

[6] S.S. Im Walde and M. K¨oper, "Pattern-based distinction of paradigmatic relations for german nouns, verbs, adjectives", In: *Lecture Notes in Computer Science*, Vol. 8105, pp.184-198, 2013.

[7] D. Lin, S. Zhao, L. Qin, and M. Zhou, "Identifying synonyms among distributionally similar words", In: *Proc. of IJCAI*, pp.1492-1493, 2003.

[8] S. Mohammad, B.J Dorr, G. Hirst, and P.D. Turney, "Computing lexical contrast", *Computational Linguistics*, Vol. 39, No.3, pp.555-590, 2013.

[9] P.D. Turney, "A uniform approach to analogies, synonyms, antonyms, and associations", In: *Proc. of COLING,* pp.905-912, 2008.

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space", In: *Proc. of ICLR,* 2013.

[11] J. Pennington, R. Socher, and C.D. Manning, "Glove: Global vectors for word representation", In: *Proc. of EMNLP*, Vol. 14, pp.1532-1543, 2014.

[12] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information", In: *Proc. of TACL* 5, pp.135-146, 2017.

[13] M. Ono, M. Miwa, and Y. Sasaki, "Word embedding-based antonym detection using thesauri and distributional information", In: *Proc. of HLT-NAACL*, pp. 984-989, 2015.

[14] N.T. Pham, A. Lazaridou, and M. Baroni, "A multitask objective to inject lexical contrast into distributional semantics", In: *Proc. of NAACL*, pp.21-26, 2015.

[15] K.A. Nguyen, S.S. im Walde, and N.T. Vu, "Integrating distributional lexical contrast into word embeddings for antonym-synonym distinction", In: *Proc. of NAACL*, p. 454-459, 2016.

[16] N. Mrksic, I. Vulic, D. S´eaghdha, I. Leviant, R. Reichart, M. Gasic, A. Korhonen, and S.J. Young, "Semantic specialisation of distributional word vector spaces using monolingual and cross-lingual constraints", *Transactions of the ACL,* 2017.

[17] S. Scheible, S.S. im Walde, and S. Springorum, "Uncovering distributional differences between synonyms and antonyms in a word space model", In: *Proc. of IJCNLP*, pp.489-497, 2013.

[18] E. Santus, Q. Lu, A. Lenci, and C.R. Huang, "Unsupervised antonym-synonymdiscrimination in vector space", In: *Proc. of CLiC-it,* 2014.

[19] K. Fundel, R. K¨uffner, and R. Zimmer, "Relex -relation extraction using dependency parse trees", *Bioinformatics*, Vol. 23, No. 3, pp.365-371, 2007.

[20] Y. Xu, L. Mou, G. Li, Y. Chen, H. Peng, and Z. Jin, "Classifying relations via long short term memory networks along shortest dependency paths", In: *Proc. of EMNLP,* pp.1785-1794, 2015.

[21] D.C. Can, H.Q. Le, Q.T. Ha, and N. Collier, "A richer-but-smarter shortest dependency path with attentive augmentation for relation

extraction", In: *Proc. of NAACL-HLT(1)*, pp.2902-2912, 2019.

[22] M. Joshi, E. Choi, O. Levy, D.S. Weld, and L. Zettlemoyer, "pair2vec: Compositional word-pair embeddings for cross-sentence inference", In: *Proc. of NAACL*, pp.3597- 3608, 2019.

[23] D.Q. Nguyen, M. Dras, and M. Johnson, "An empirical study for Vietnamese dependency parsing", In: *Proc. of ALTA*, pp.143-149, 2016.

[24] P.T. Nguyen, A.C. Le, T.B. Ho, and V.H. Nguyen, "Vietnamese treebank construction and entropy-based error detection", *Language Resources and Evaluation*, pp.487-519 2015.

[25] V.L. Pham, "Study antonyms in vietnamese", *doctoral thesis*, 2019.

[26] V.T. Bui, P.T. Thai, and V.L. Pham, "Construction of a word similarity dataset and evaluation of word similarity techniques for Vietnamese", In: *Proc. of KSE*, pp. 65-70, 2017.

[27] K.A. Nguyen, S.S. im Walde, and N.T. Vu, "Distinguishing antonyms and synonyms in a pattern-based neural network", In: *Proc. of EACL*, pp.76-85, 2017.

[28] P.T. Nguyen, V.L. Pham, H.H. Vu, and N.A. Tran, "A twophase approach for building vietnamese wordnet", In: *Proc. of the 8th Global Wordnet Conference,* pp. 259 – 264, 2015.

[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory", *Neural computation*, pp.1735-1780, 1997.

[30] J.S. Justeson and S.M. Katz, "Co-occurrences of antonymous adjectives and their contexts", *Computational Linguistics*, Vol. 17, No. 1, pp.1-19, 1991.

[31] S. Bordag, "A comparison of co-occurrence and similarity measures as simulations of context.", In: *Proc. of CICLing*, *Lecture Notes in Computer Science*, Vol. 4919, pp. 52-63. Springer, 2008.

[32] K.A. Nguyen, S.S. im Walde, and N.T. Vu, "Introducing two vietnamese datasets for evaluating semantic models of (dis-)similarity and relatedness", In: *Proc. of NAACL-HLT (2)*, pp.199-205, 2018.

[33] K. Washio and T. Kato, "*Filling missing paths: Modeling co-occurrences of word pairs and dependency paths for recognizing lexical semantic relations*" In: *Proc. of NAACL-HLT*, pp. 1123-1133, 2018.

[34] P. Hanks, "The impact of corpora on dictionaries", *Contemporary Corpus Linguistics,* p.214-236, 2009.