DS/SP 829 - Natural Language Processing
Lexical Entailment

Rahul Murali Shankar (IMT2017033)
Ram S (IMT2017521)
Prateksha U (IMT2017517)

International Institute of Information Technology, Bangalore

Abstract. In this report, we would like to document our efforts to solve the task of multi-lingual and cross-lingual lexical entailment for two different measurement criteria: binary and graded lexical entailment. We are using the latest techniques in pre-processing word vectors like Lexical Entailment Attract Repel (LEAR) and Multi-Step Linear Transformations (VecMap) to accomplish the task.

1 Introduction

Lexical Entailment is the process of deriving relationships between a pair of given words that satisfy certain properties. It aims to identify the *semantic* relationships between words, which can be broadly explained as a hyponym-hypernym relationship.

Given two words a and b, a lexically entails b if the meaning or context of b can be inferred from the meaning of a, or if the word b is a superset of the class of words in which the word a is present. An example would be that the word dog entails the word animal since a dog is an animal.

The concept of lexical entailment is different from the concept of textual entailment, in that the latter involves the comparison of two text fragments of any length. One could say that the problem of identifying lexical entailment is a subset of the textual entailment problem.

Formally, one can define lexical entailment as a relationship between two words that satisfies either one of the following cases (given two words a and b of the same language):

- The meaning of a possible sense of a implies a possible sense of b.
- Word b can substitute for word a in a given sentence, such that the meaning of the modified sentence entails the meaning of the original sentence.

This is what monolingual lexical entailment can be viewed as. We also have the additional task of cross-lingual lexical entailment, and it seems obvious that

the above definition cannot fully capture the nuances of the entailment due to various context clues (formally semantic relations and associations) inhibiting a direct conversion approach. Thus, we can give a formal definition for cross-lingual lexical entailment as follows, where word a is part of language A and word b is part of language B:

- The meaning of a possible sense of a implies a possible sense of b.

- Given a sentence in language B containing the word b, a can be substituted for b in the translation of the sentence to language A, such that the meaning of the modified sentence entails the meaning of the original sentence.

Thus, we are able to provide a definition for cross-lingual lexical entailment that helps us understand the semantic mappings across languages while also staying consistent with the method with which we have defined monolingual lexical entailment.

One can also broadly classify the pairs of words that lexically entail one another into three categories (note that these categories have been defined by us and may not be exhaustive in capturing all possible lexical relationships between a pair of words in any language):

1. *Hyponym-Hypernym relationship:* This category consists of words where the semantic field of one word is included within that of another word. The above example, where the word dog entails the word animal, is an example of a hyponym-hypernym relationship, since the class dog is a subset of the class animal (i.e. a dog is an animal). The consequence of this relationship is that the hyponym can be replaced by its corresponding hypernym in a sentence and still retain its meaning to some degree. Note that this is not a symmetric relation, since the word animal does not entail the word dog.

2. *Correlation:* This category consists of pairs of synonyms, where one word can be replaced by another in a given text fragment and retain a nearly identical context. An example would be the words smart and clever. In this case, the relation is symmetric since one word can be replaced by the other and still retain its meaning.

3. *Consequential relationship:* This category consists of words where the action constituted by one word is a consequence of the action constituted by the other. An example would be that the word buy entails the word own (given that if you buy something, you own it). The semantic meanings of the words are mutually exclusive in that one cannot be replaced by the other and still retain the context, but it is still a form of lexical entailment in that we can derive a semantic relationship between the words. Note that neither is this a symmetric relationship, since two words cannot be a consequence of each other, and nor is it a substitutable relationship, since one word cannot be replaced by the other and retain the context.

We will not be dealing with all of these classes in the process of completing the given task, but rather only depend on the dataset that we are using to train the model, which may not contain pairs of words that belong to one of the above classes. The most common form of lexical entailment is substitutable entailment, wherein one word can be replaced by another, as is the case in the first two categories of the above classification.

2 Given Task

The given task is to perform binary and graded lexical entailment on word pairs given multi-lingual and cross-lingual datasets.

The aim of the binary variant is to predict whether one word lexically entails another and output a 0 or 1, while the graded variant predicts a score (in the range of 0-5) based on to what degree one word entails another.

3 Related Work

We have come across many ways of solving the lexical entailment problem, mostly using the general technique of vector space models (VSMs), some of which are listed below:

1. *Directional Similarity:* This is an asymmetric similarity measure, which is an instance of the directional similarity strategy, designed to capture the degree to which the context of word a forms a subset of the context of word b [1]. The algorithm, called balAPinc (balanced average precision for distributional inclusion), attempts to design a measure that captures the context inclusion hypothesis. The context inclusion hypothesis simply states that if word a tends to ocur in a subset of contexts that word b occurs in, then a entails b. Basically, we take the context vectors for a and b and calculate a numerical score that measures the degree to which b contextually includes a.

2. *Relation Classification with context vectors:* This method, for two words a and b, represents a word pair $a : b$ where the feature vector is the concatenation of the context vectors of a and b. The algorithm then applies then applies supervised learning to a training set containing these feature vectors [1].

3. *Relation Classification with difference in similarity:* This method represents a word pair $a : b$ with the feature vector in which the features are the difference in the similarities of a and b to a set of reference words ([1]). Then a similar process of supervised learning with the constructed feature vectors is applied. This method and the one above attempt to recognize lexical entailment using techniques from research in semantic relation classification.

4. *Distributional Semantics:* This is based upon the distributional hypothesis which states that if two words appear in similar contexts, they can be assumed to have similar meaning [2]. There are several models that fall under this, like predicting specific ontology relations and entailment decisions between lexical items devoid of context and predicting specific lexical paraphrases in complete sentences.

5. *LEAR:* This is a post-processing method that transform any input word vector space to emphasize the asymmetric relation of lexical entailment [3]. The asymmetric distance measure used here adjusts the norms of word vectors to reflect the actual WordNet style hierarchy of concepts.

6. *Unsupervised Machine Translation:* More specifically, we focus on the concept of unsupervised SMT (statistical machine translation). The traditional SMT approach induced an initial phrase-table through cross-lingual embedding mappings, combining it with an n-gram language model and improving it though iterative back-translation. The unsupervised version incorporates subword information, a well defined tuning method and a joint refinement procedure while also using it to develop an unsupervised NMT (neural machine translation) model [4].

7. *Improving Bi-lingual Word Embedding Mappings through a multi-step framework of Linear Transformations:* Using a dictionary to map independently trained word embeddings to a shared space was shown to be an effective approach to learn bilingual word embeddings. In this work, a multi-step framework of linear transformations is proposed that generalizes a substantial body of previous work. The core step of the framework is an orthogonal transformation along with other pre and post-processing steps. The corresponding software is released as an open source project known as VecMap [5].

Now that we have a good grasp of some of the related work in this field, we have chosen to focus on and enhance the LEAR algorithm for solve the lexical entailment problem, aided by the concepts of learning principled bilingual mappings of word embeddings to accomplish the given task.

4 LEAR

LEAR - Lexical Entailment Attract-Repel is a post processing method to transform a given word vector space , where the distributional vectors are refined to emphasise the asymmetric relation of Lexical Entailment.

The key idea of LEAR is to pull (*attract*) desirable word pairs closer to each other, and push (*repel*) undesirable word pairs away from each other. Concurrently, it also re-arranges the vector norms so that the norm values in the Euclidean space reflect the hierarchy of concepts involved.

Fig. 1. Illustration of the mechanism of LEAR

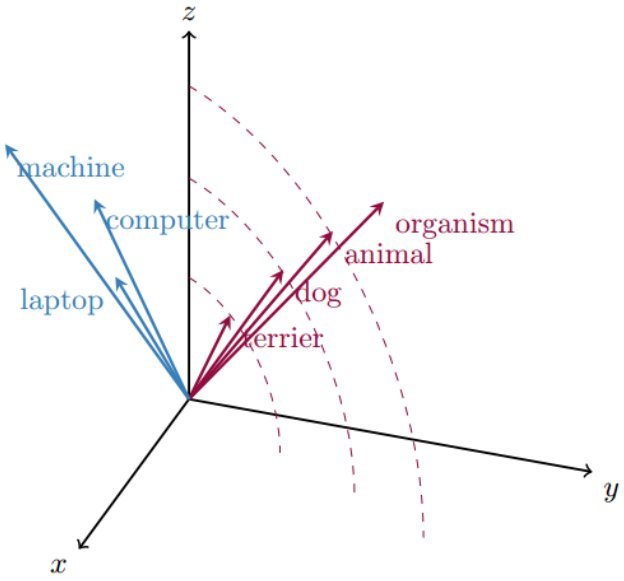

Figure 1 explains the two main concepts of LEAR:

1. Emphasising symmetric similarity of LE pairs through cosine distance. i.e., The angle between ($\overrightarrow{terrier}$ and $\overrightarrow{dog}$), ($\overrightarrow{dog}$ and $\overrightarrow{animal}$), ($\overrightarrow{animal}$ and $\overrightarrow{organism}$) is small indicating the similarity of concepts.
2. Imposing the LE ordering using vector norms. i.e., The order of norm values represent the hierarchy of concepts.
$|\overrightarrow{terrier}| < |\overrightarrow{dog}| < |\overrightarrow{animal}| < |\overrightarrow{organism}|$

LEAR Methodology:

1. The ATTRACT-REPEL Framework
V = Vocabulary,
A = Set of Attract Pairs (ex: intelligent and brilliant): Synonyms are considered as the Attract Pairs
R = Set of Repel Pairs (ex: vacant and occupied): Antonyms are considered as the Repel Pairs
Negative examples are found for each of the attract pairs and repel pairs.
The main idea of this framework is to
   (a) force the ATTRACT pairs to be closer to each other than to their respective negative examples.

(b) force the REPEL pairs to be further away from each other than they are from their negative examples.

The above idea idea is implemented in the below cost function
The first term pulls ATTRACT pairs together.

$$Att(B_A, T_A) = \sum_{i=1}^{k_1} [\tau(\delta_{att} + cos(x_l^i, t_l^i) - cos(x_l^i, x_r^i)) + \tau(\delta_{att} + cos(x_r^i, t_r^i) - cos(x_l^i, x_r^i))] \quad (1)$$

where cos denotes cosine similarity,
$\tau(x) = max(0, x)$ and δ_{att}

The second term pushes REPEL pairs away from each other.

$$Rep(B_R, T_R) = \sum_{i=1}^{k_2} [\tau(\delta_{rep} + cos(x_l^i, x_r^i) - cos(x_l^i, t_l^i)) + \tau(\delta_{rep} + cos(x_l^i, x_r^i) - cos(x_r^i, t_r^i))] \quad (2)$$

The third term is a regularization term.

$$Reg(B_A, B_R) = \sum_{x_i \in V(B_A \cup B_R)} \lambda_{reg} ||\hat{x_i} - x_i||_2 \quad (3)$$

where λ_{reg} is the L_2 regularization constant and
$\hat{x_i}$ denotes the original word vector for x_i

2. Encoding Lexical Entailment
To do this, an additional source of external lexical knowledge is used. B_L contains k_3 word pairs of lexical constraints. Ex: (dog, animal), (animal, organism). The lexical entailment term is defined as:

$$LE_j(B_L) = \sum_{i=1}^{k_3} D_j(x_i, y_i) \quad (4)$$

where $D_j$is one of the following asymmetric distance functions

$$D_1(x, y) = |x| - |y|$$
$$D_2(x, y) = \frac{|x| - |y|}{|x| + |y|}$$
$$D_3(x, y) = \frac{|x| - |y|}{max(|x|, |y|)}$$

3. The LEAR cost function =
$Att(B_S, T_S) + Rep(B_A, T_A) + Reg(B_A, B_R, B_L) + Att(B_L, T_L) + LE_j(B_L)$
4. The above defined cost function encodes semantic similarity as well as LE relations in the same vector space.

5. Finally, the metric used to determine whether a given word pair is lexically entailed is given below:
   $I_{LE}(x, y) = dcos(x, y) + D_j(x, y)$ The first term is the cosine distance and the second term is the asymmetric cost term.

5 Solving Cross-Lingual Lexical Entailment

To solve the problem of cross-lingual lexical entailment, we have to deal with two main problems. The problem of lexical entailment itself, and how we could solve it in the cross-lingual context. We have discussed in great length as to how we could solve the traditional lexical entailment problem (through LEAR), and so, in this section, we are going to explore the various methods by which we attempt to solve cross-lingual lexical entailment.

5.1 Method 1: Direct Translation

Since we already have a well performing method for solving lexical entailment in English, one of the simplest methods to solve this problem would be to directly translate words in other languages into the words of our choice, and perform the lexical entailment in English itself. This method is referred to as TRANS and while it performs well, there are many semantic aspects that this model simply will not capture. For example, let's take the given examples as mentioned in [6]:

English → English	**English → French**
affection → feeling	affection → sentiment
aspirin → drug	aspirin ↛ drogue
water → wet	water → humide
feeling ↛ nostalgia	feeling ↛ nostalgie

Table 1. Difference between Monolingual and Cross-Lingual Lexical Entailment

Here, we can clearly see that while *aspirin* entails the English word *drug*, it does not entail the French word *drogue*, which only refers to the narcotic sense of the word *drug* and not to its medical sense. But, generally the translation for *drug* is given as *drogue*. So, through examples like these, we can see that each language has its own dynamics that need to be understood by the model before we are able to perform cross-lingual lexical entailment.

5.2 Method 2: Bi-lingual Word Mappings

Now that we have shown that Direct Translation is unable to capture the intricacies of different languages, we need a better method to account for language specific constraints in the overall cross-lingual model. One way to achieve this would be to work with Bi-lingual word mappings. Essentially, these are word

mappings for two different languages, that are treated as if they occupy the same space i.e. the feature vectors for words of both languages are contained in the same space. It is for the building of this shared space that we investigated some systems such as VecMap [5] and CLEAR [7].

VecMap Authors over the past few years have proposed different methods to learn such word embedding mappings, but their approaches and motivations were often divergent, making it difficult to get a general understanding of the topic. In this work, the authors tackle this issue by generalizing previous works. The core step of the framework, which maps both languages to a shared space by using an orthogonal transformation, is shared by all variants, and the differences from previous methods are exclusively explained in terms of their normalization, whitening, re-weighting and dimensionality reduction behaviour. This is a novel variant combining these various factors that improves the state of the art in bilingual lexical extraction.

This framework is highly related to the zero-shot learning paradigm where a multi-class classifier trained over a subset of the labels learns to predict unseen labels by exploiting a common representation for them. In this scenario, these labels correspond to target language words and their common representation is provided by their corresponding embeddings. This is prototypical zero shot learning problem, and similar mapping techniques have also been used in other zero-shot tasks like image labelling etc.

CLEAR CLEAR expands as Cross-lingual Lexical Entailment Attract Repel. It was proposed by Ivan et. al. [7]. It is an enhancement of the classic LEAR algorithm to allow for cross-lingual considerations.

The input to the method is as follows. Two independently trained monolingual word vector spaces in two languages L1 and L2, sets of external lexical constraints in the resource-rich language L1 (say English), and a bilingual L1-L2 dictionary D. The goal is to fine-tune input word vectors in both languages using the L1 lexical constraints and the dictionary D, and obtain a shared cross-lingual space specialised for lexical entailment. Additionally, CLEAR adds a regularization term to the cost function of the LEAR algorithm due to the addition of the dictionary D to the constraints.

Similar to LEAR and the Attract-Repel model for symmetric similarity specialisation, CLEAR defines two types of symmetric objectives for the L1 pairs.

1. The ATTRACT (*Att*) objective aims to bring closer together in the vector space words that are semantically similar (i.e. synonyms and hyponym hypernym pairs)
2. The REPEL (*Rep*) objective pushes apart vectors of dissimilar word (i.e. antonyms).

We denote $\beta = (x_l^{(k)}, s_r^{(k)})_{k=1}^{K}$ as the set of k word vector pairs for which the *Att* or *Rep* score is to be computed. These are referred to as *positive examples* (B). The set of corresponding *negative examples* T is created by coupling each positive ATTRACT example (x_l, x_r) with a negative example pair (t_l, t_r) where t_l is the vector closest (within the current batch in terms of cosine similarity) to x_l and t_r the vector closest to x_r. This design is very similar to the design of the *attract* term in LEAR and a similar design is followed for the *repel* term as well.

Crucially, similar to LEAR, this method forces specialised vectors to reflect the asymmetry of the LE relation with an asymmetric distance-based objective. Starting from the *Le* (hyponym-hypernym) pairs, the goal is to rearrange vectors of words in these pairs, that is, to preserve the cosine distances in the specialised space while steering vectors of more general concepts to take larger norms. This term is represented by $LE(B_{Le})$.

Now, as we have covered all the terms occurring in LEAR, we need to consider the cross-lingual aspect of this problem. This is handled by the translation pairs from the dictionary D that are also "attracted" to each other, but using a different objective. This term is represented as $Att_D(B_D)$ objective on a batch of translation pairs B_D as the simple l_2 distance between the two word in each pair:

$$Att_D(B_D) = \lambda_D \sum_{k=1}^{K} ||x_l^{(k)} - x_r^{(k)}|| \tag{1}$$

Let us look at this term in detail. We can see that $x_l^{(k)}$ is the vector of an L1 word from the source language vector space and $x_r^{(k)}$ the vector of its L2 translation from the target language space. λ_D is the cross-lingual regularisation factor. So, the final Cost Function formulation is as follows:

$$\begin{aligned} J = Att(B_S, T_S) + Rep(B_A, T_A) + \\ Att(B_Le, T_Le) + LE(B_Le) + \\ Att_D(B_D) + Reg(B_s, B_A, B_Le, B_D) \end{aligned}$$

This joint objective rearranges vectors from both input monolingual vector spaces and enables the transfer of LE signal from the resource rich language L1 to the target language.

5.3 LE Determining Criteria

Monolingual and cross-lingual LE strength can be inferred directly from the CLEAR-specialised cross-lingual space. It is done through a distance function that reflects both the cosine distance between the vectors (semantic similarity)

as well as the asymmetric difference between the vector's norms.

$$I_{LE}(x, y) = dcos(x, y) + \frac{||x|| - ||y||}{||x|| + ||y||} \tag{2}$$

where x and y are vectors of any two words x and y in the cross-lingual space. Now, for binary LE, we just ensure that $I_{LE}(x, y) < t$ where t is some binarization threshold. CLEAR specialised vectors of general concepts obtain larger norms than vectors of specific concepts. Strong LE pairs should display both small cosine distances and negative norm differences.

6 Solution Ideas

Given below are the list of solution ideas that were used to try to solve Cross-lingual Lexical Entailment:

CLEAR CLEAR itself is a solution to cross-lingual lexical entailment.

Enhancing/Modifying CLEAR CLEAR adds an L2 distance calculation term for cross-lingual synonym constraints ($Att_D B_D$). However, we believe that this could be experimented upon to see if other distance calculations can change/improve results.

Using a hyperbolic word space While we haven't explored this aspect deeply, the authors of CLEAR believe that there is some significant scope of research available in using a hyperbolic space word embedding on CLEAR for cross-lingual lexical entailment.

Enhacing the input bi-lingual shared space The bi-lingual word vector space used in CLEAR was a simply superposition of word vectors for each language into the same space and greatly depended on the dictionary (cross-lingual word translation) to help identify Lexical Entailment Relationship. We think that fine-tuning this initial word space (through VecMap) before passing it to CLEAR could be a viable solution idea.

Method	Mono/Cross	Word Vectors	Testing	L2	Epochs	Accuracy
LEAR	Mono	VecMap(Word2vec)	HyperLex	-	5	0.682
LEAR	Mono	VecMap(Word2vec)	HyperLex-Nouns	-	5	0.702
CLEAR	Multi (EN-DE-IT-RU)	VecMap(Word2vec)	Custom Data	German	5	0.26
CLEAR	Bi (EN-DE)	VecMap(Word2vec)	Custom Data	German	5	0.46

Table 2. Results

7 Results

Given in Table 2 are the results of some of the experiments we were able to run.

8 Conclusion

In this report, we have explored Lexical Entailment as a task in great detail. We have defined the problem of Cross-Lingual lexical Entailment and documented various methods/solutions that exist for solving cross-lingual lexical entailment. We have even attempted to perform our own enhancements/experiments to existing algorithms as well. The code is available here.

References

[1] Peter D. Turney and Saif M. Mohammad. "Experiments with Three Approaches to Recognizing Lexical Entailment". In: *CoRR* abs/1401.8269 (2014). arXiv: 1401.8269. URL: http://arxiv.org/abs/1401.8269.

[2] Stephen Creig Roller. "Identifying lexical relationships and entailments with distributional semantics". PhD thesis. 2017. URL: http://hdl.handle.net/2152/61528.

[3] Ivan Vulić and Nikola Mrkšić. "Specialising Word Vectors for Lexical Entailment". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, June 2018, pp. 1134–1145. DOI: 10.18653/v1/N18-1103. URL: https://www.aclweb.org/anthology/N18-1103.

[4] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. "An Effective Approach to Unsupervised Machine Translation". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 194–203. DOI: 10.18653/v1/P19-1019. URL: https://www.aclweb.org/anthology/P19-1019.

[5] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. "Generalizing and Improving Bilingual Word Embedding Mappings with a Multi-Step Framework of Linear Transformations". In: (2018). URL: https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16935/16781.

[6] Yogarshi Vyas and Marine Carpuat. "Sparse Bilingual Word Representations for Cross-lingual Lexical Entailment". In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 1187–1197. DOI: 10.18653/v1/N16-1142. URL: https://www.aclweb.org/anthology/N16-1142.

[7] Ivan Vulić, Simone Paolo Ponzetto, and Goran Glavaš. "Multilingual and Cross-Lingual Graded Lexical Entailment". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 4963–4974. DOI: 10.18653/v1/P19-1490. URL: https://www.aclweb.org/anthology/P19-1490.