LG at WiC-ITA: Exploring the relation between semantic distance and equivalence in translation.

Lorenzo Gregori

University of Florence

Abstract

The Word in Context task has been addressed here on the basis of a simple intuition: if the same lemma has different senses in two different contexts, it tends to be translated (in other languages) with two different lemmas; conversely, in the case of the same sense, translation lemmas tend to be the same. The proposed methodology is based on the translation of sentence pairs in 21 languages, and the use of a SVM classifier/regressor. Obtained results are excellent in binary classification and average in regression tasks.

Keywords

EVALITA, Word In Context, Machine Translation, Semantics

1. Introduction

The Word-in-Context (WiC) task is a new task [1, 2], that aims to identify if the same word used in two contexts has the same sense in both contexts or if it's used with different senses. Interestingly, in the current EVALITA task [3, 4] sentences are manually judged by several annotators.

The method chosen to solve this task is completely different from the previously used methods, mostly based on the use of pre-trained language models: see recent works by Alan Ansell and colleagues, and Qianchu Liu and colleagues, among others [5, 6]. Actually, in the proposed approach language models are used not to directly accomplish the given task, but to provide accurate sentence translations in several languages.

The intuition behind the proposed methodology is that a word with the same sense in two contexts is probably translated with the same lemma in a target language; otherwise, a word that is used in two different senses tends to be translated with two different lemmas. This idea has been first explored by Gale, Church, and Yarowsky in the early 1990s [7, 8], and is a basic concept behind the work on semantic spaces produced by Melissa Bowerman and colleagues in the early 2000s [9, 10], and the IMAGACT Ontology of Action [11, 12].

Gale and colleagues showed that having a text translated into another language can be useful for word sense disambiguation, given that a word with two different senses is frequently translated with two different words in the target language. Bowerman and colleagues analyzed the variation of event categorization in different languages, by studying the semantic variation of general verbs "cut" and "break". Their work showed how languages use their own verbs to partition differently the semantic space related to these two events. IMAGACT[1] highlights the relation between different semantic types of an action verb, and different groups of verbs usable to predicate them. From this resource, we can clearly observe that, in general, the verb set allowed for one type is not the same set allowed for another type, even if the two types belong to the same verb. This is a shared property across languages: it occurs in Italian, English, and in many other languages.

The proposed approach is based on high-quality translation of the provided sentences in several languages; translated sentences are then word-aligned to the original ones, and lemmatized. With this data, it's easy to verify if the lemma used to translate the target word in the two original sentences is the same or not. This binary information repeated for each language is used to compile a feature vector. Then, a Support Vector Machines (SVM) classifier and an SVM regressor are trained on these vectors to decide if (or how much) the two word senses are different.

2. Word semantics in translation

Translating a word to another language is not a trivial task, because it's pretty rare to find a word in the target language with exactly the same meaning as the original word. More likely, there are several possible translators, each one suitable for some contexts and not for others. Moreover, even at the lower level of word senses, it is hard to find perfect matches between two languages, given that languages partition semantic spaces in their own way [13, 14].

EVALITA'23: Evaluation of NLP and Speech Tools for Italian, September 7-8, 2023, Parma, Italy

lorenzo.gregori@unifi.it (L. Gregori)
0000-0001-9208-2311 (L. Gregori)

[1] http://www.imagact.it/

	$s(w_a) = s(w_b)$	$s(w_a) \neq s(w_b)$
$s(t_1) = s(t_2)$	$p =?$	$p =?$
$s(t_1) \neq s(t_2)$	$p =?$	$p =?$

Table 1
The set of 21 translation languages used to solve the WiC task.

These premises, plus the fact that the same word can have different meanings, and different words can be synonyms, make the picture of semantics in translation very complex.

Consider two occurrences of the same word (in different contexts), w_a and w_b, translated to another language with two words, t_1 and t_2. Then, w_a and w_b can have the same sense or different senses; t_1 and t_2 can be the same word or different words, and in both cases, they can have the same sense or two different senses (they can be synonyms if they are different word with the same sense, or polysemous if they are the same word with different senses).

So, the 4 cases represented in Table 1 ($s(x)$ means the sense of the word x) are all possible, both if t_1 and t_2 are a single word, or if they are different words. It is the probability of these cases that is undetermined ($p =?$).

The assumption behind this experiment is that the following two cases are more frequent than others:

- $(s(w_a) = s(w_b)) \land (s(t_1) = s(t_2))$ if t_1 and t_2 are the same word;
- $(s(w_a) \neq s(w_b)) \land (s(t_1) \neq s(t_2))$ if t_1 and t_2 are two different words.

If this is true for most sentences in most languages, then the use of many translation languages provides reliable information to identify semantic distance between w_a and w_b.

The two examples reported below are derived from the test set and aim to clarify the idea behind the proposed methods, and the practical issues.

Example 1. The target word has two different senses. The following two sentences have been labeled as 0 (i.e. target word belonging to different senses):

1. *Chi ha intenzioni meno serie, troverà godibile il tour fotografico del* ***complesso*** *, completato dalla visita virtuale del museo...*
2. *...la Camera dei deputati aveva approvato un* ***complesso*** *di disposizioni leggermente diverse da quelle recepite dalla n. 180.*

In fact, in most languages two different lemmas have been used in translation:

- French uses **complexe** in (1) and **ensemble** in (2);
- Finnish uses **kompleksi** in (1) and **joukko** in (2);
- English uses **complex** in (1) and **set** in (2).

The adoption of two different lemmas to translate these occurrences of "complesso" is spread over several languages, but, of course, there are some exceptions. In Bulgarian, for example, in both of the sentences, "complesso" is translated with a unique word "комплекс", similar to Italian.

Dealing with translations, it's not possible to rely on a perfect word-to-word alignment: it can occur that multiple words are translated with only one word in the target language, or, vice-versa, a unique word translated with more words, or some words are just omitted in translation. Moreover, some alignment errors must be expected by the word aligner. In this example, the word in (1) is translated into Lithuanian with "kompleksas", while in (2) there is not any word aligned to "complesso". It could be due to the linguistic properties of Lithuanian or to an error in the word alignment.

Example 2. The target word has the same sense. The following two sentences have been labeled as 1 (i.e. target word belonging to the same sense):

1. *inserendo in questa* ***maschera*** *la parola greca "mache" (battaglia) si otterranno tutti i termini collegati...*
2. *Questa* ***maschera*** *consente di visualizzare alcune informazioni in forma sintetica: il titolo del documento,...*

As in the previous example, most languages used the same lemma to translate "maschera" in these two contexts:

- Lithuanian used **kaukė** for both (1) and (2);
- Malay used **topeng** for both (1) and (2);
- Spanish used **máscara** for both (1) and (2).

Icelandic used two different words: **sjónarhóll** for (1), and **gríma** for (2).

3. Description of the system

The system proposed to solve the WiC-ITA task is divided into two parts: (a) the creation of a feature vector related to each sentence pair, and (b) the training of a machine learning algorithm on the feature vectors.

3.1. Feature vectors

Given two sentences $s1$ and $s2$, containing both an occurrence of the same lemma; these occurrences are the target words $w1$ and $w2$. For each of the 21 languages (l) considered (see Table 2 for the full list), the algorithm performs the following steps:

Albanian	French	Lithuanian
Armenian	German	Macedonian
Bulgarian	Greek	Malay
Czech	Hindi	Manx
Danish	Hungarian	Russian
English	Icelandic	Spanish
Finnish	German	Macedonian

Table 2
The set of 21 translation languages used to solve the WiC task.

1. Translate $s1$ in language $l \rightarrow t1$;
2. Translate $s2$ in language $l \rightarrow t2$;
3. Align at word-level $t1$ to $s1 \rightarrow t1a$;
4. Align at word-level $t2$ to $s2 \rightarrow t2a$;
5. Lemmatize $t1a$ and find the lemma of $t1a$ aligned to $w1 \rightarrow w1a$;
6. Lemmatize $t2a$ and find the lemma of $t2a$ aligned to $w2 \rightarrow w2a$;
7. Assign 0.5 if $w1a$ or $w2a$ are non-words;
   Assign 0 if $w1a = w2a$;
   Assign 1 if $w1a \neq w2a$;

The result of this algorithm is a feature vector v with a number {0, 0.5, 1} for each language. The size of v is the number of languages used (21 in the proposed experiment).

As an example, consider the following sentences pair, belonging to the test set:

- *... che vi sia alla base un accordo tra i coniugi, soprattutto in relazione all'educazione del* ***minore****.*
- *Infatti ogni religione istituzionalizzata annette importanza maggiore o* ***minore*** *alla propagazione ... dei suoi riti.*

After the algorithm execution, the v vector appears as below (only the first 10 elements are reported).

$$\begin{array}{ccccccccccc} en & sv & el & id & ru & hi & es & de & hy & bg & \\ (\,1 & 1 & 1 & 0.5 & 1 & 1 & 0 & 1 & 1 & 1 & \ldots) \end{array}$$

Most of the values are 1, meaning that many languages translated the word *minore* with different words; there is a 0.5 in 4th position, meaning that there is a missing alignment of the word *minore* with a corresponding Indonesian (id) word (in at least one of the two sentences); the value of 0 (7th position) means that in Spanish (es) *minore* has been translated with the same word in both the contexts.

3.2. External tools

The key point of this algorithm is the sentence translation engine, which needs to have a high accuracy, and ensure a really contextualized translation; moreover, the translation system must be available in several languages. To this aim, Opus-MT system [15, 16] has been selected: it is a state-of-the-art system of neural machine translation, for which more than 1,400 pretrained models are freely available[2]: each model is specifically trained on a language pair.

label	inst.
0	806
1	1,999

Table 3
Number of instances (sentence pairs) for each label in the training set of the binary classification task.

To align sentences and translations at the word level, the multilingual word aligner created by Dou & Neubig has been selected [17]: the pretrained model aligned on multilingual BERT is freely available online[3].

The lemmatizer used for this work is Simplemma[4], a tool available as a Python library that performs sentence lemmatization in over 50 languages.

Both the aligner and the lemmatizer have been chosen to be multilingual ready-to-use tools, that can be easily included in the implemented pipeline.

3.3. Training the machine

Support Vector Machines algorithms have been used on the current dataset both for regression and classification.

Algorithms have been trained on the Italian dataset and tested on the Italian dataset only.

Binary classification task After the conversion of each pair of input sentences in a vector, a classifier was trained on the training set and tested on the development set. An SVM classifier with a linear kernel was chosen for its good performance on this task. The complexity hyperparameter (C) has been tuned on the dev set, obtaining the best results with $C = 0.1$.

The dataset has been balanced with a random undersampling technique, to train the algorithm with an equal number of examples of positive and negative classes (i.e. the sentences where the target word has the same sense, and the ones where the target word has different senses). As reported in Table 3, the training set for the classification task is highly unbalanced, with a proportion between the two classes of 29% - 71%.

Ranking task

The ranking task has been solved with an SVM regressor, using a Gaussian kernel, that led to better performance

[2] https://huggingface.co/Helsinki-NLP
[3] https://github.com/neulab/awesome-align
[4] https://pypi.org/project/simplemma/

score	inst.	score	inst.
1	387	3	303
1.5	300	3.5	723
2	119	4	973

Table 4
Number of instances (sentence pairs) for each score in the training set of the ranking task.

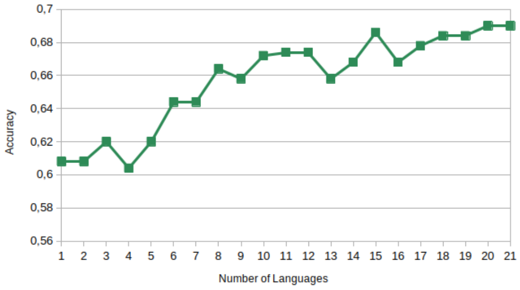

Figure 1: Accuracy growth of the binary classification, when increasing the number of languages.

than the linear kernel[5].

A preliminary analysis of the training set for the ranking task highlighted that data are highly unbalanced; moreover, the score values assigned to each pair of sentences are exactly 6: {1, 1.5, 2, 3, 3.5, 4 } (see Table 4).

It is possible to identify two sources of biases: (a) 71% of instances have a high score (3 to 4), while only 29% has a low score (1 to 2); there is an increasing number of scores, moving form central scores (2 and 3) to the extreme scores (1 and 4). This suggested performing a random undersampling to balance the sentences assigned to each score and train the algorithm in a reduced unbiased space.

4. Results

The first issue that emerged in the training was about the number of languages that should be used to obtain higher accuracy. So, at first, the algorithm has been tested using an increasing number of languages (1 to 21).

The graph reported in Fig. 1 shows the change in accuracy of the binary classification algorithm with respect to the number of languages used for training. This graph is based on the development set.

In general, we can see that with just one language the algorithm accuracy is about 0.60, moving up towards 0.70 as the number of languages increases. The choice of 21 languages seems reasonable, considering the trend of the curve, which starts flattening with more than 10 languages.

Table 5
Best results of the Italian WiC task.

Team	Binary	Ranking
BERT4EVER	0.56	0.34
LG	**0.73**	0.49
The Time-Emb. Travelers	0.67	0.55
extremITA	0.61	-
baseline	0.59	**0.57**

Results of the Italian task are resumed in Table 5, where the highest accuracy for each team is reported. The results obtained in the classification task (on the test set) is an accuracy of 0.73, which is very high for a WiC competition [2]. Conversely, this algorithm didn't emerge in the ranking task, obtaining an average score of 0.49, which is below the baseline.

5. Discussion

The high accuracy obtained in the classification task with only features related to lemma equivalences in translation is, first of all, a piece of strong evidence to support some linguistic theory about semantic spaces.

The implemented model is easy to interpret, and quick to train. This makes the proposed system, a very flexible tool to perform other experiments, like changing the number and the type of languages used, finding the minimal set of languages with the maximum discriminative power, and so on. It would be also interesting to try to apply on another language the model trained in one language, as suggested by the task organizers.

The proposed algorithm can be improved:

- The number of languages is probably enough to reach the maximum accuracy with feature vectors; otherwise the set of used languages could be changed, by introducing, for example, some Asian languages, like Chinese, Japanese, or Korean, that probably would bring a big contribution to this task;
- Alignment and lemmatization could be improved, by using, for each language, a state-of-the-art tool that is specifically tuned for that language; this would probably lead to results that are better than the ones obtained with multi-language tools.

About the computational cost of the proposed approach, it is completely moved from the training stage (which has almost no cost) to the feature extraction: in fact, the computationally intense stage is the neural machine translation in 21 languages. This task is also very slow, but easy to parallelize, using simultaneous translation engines. Interestingly enough, once the vectors have been compiled for the dataset in use, all the experiments

[5] Complexity hyper-parameter C is tuned to 0.1, as in the classification problem

with changes in the types or the number of languages are not costly.

References

[1] M. T. Pilehvar, J. Camacho-Collados, Wic: the word-in-context dataset for evaluating context-sensitive meaning representations, arXiv preprint arXiv:1808.09121 (2018).

[2] A. Raganato, T. Pasini, J. Camacho-Collados, M. T. Pilehvar, Xl-wic: A multilingual benchmark for evaluating semantic contextualization, 2020. arXiv:2010.06478.

[3] P. Cassotti, L. Siciliani, L. Passaro, M. Gatto, P. Basile, Wic-ita at evalita2023: Overview of the evalita2023 word-in-context for italian task, in: Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), CEUR.org, Parma, Italy, 2023.

[4] M. Lai, S. Menini, M. Polignano, V. Russo, R. Sprugnoli, G. Venturi, Evalita 2023: Overview of the 8th evaluation campaign of natural language processing and speech tools for italian, in: Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), CEUR.org, Parma, Italy, 2023.

[5] A. Ansell, F. Bravo-Marquez, B. Pfahringer, An elmo-inspired approach to semdeep-5's word-in-context task, in: Proceedings of the 5th Workshop on Semantic Deep Learning (SemDeep-5), 2019, pp. 21–25.

[6] Q. Liu, F. Liu, N. Collier, A. Korhonen, I. Vulić, Mirrorwic: On eliciting word-in-context representations from pretrained language models, arXiv preprint arXiv:2109.09237 (2021).

[7] W. A. Gale, K. Church, D. Yarowsky, Using bilingual materials to develop word sense disambiguation methods, in: Proceedings of the Fourth Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages, 1992.

[8] W. A. Gale, K. W. Church, D. Yarowsky, A method for disambiguating word senses in a large corpus, Computers and the Humanities 26 (1992) 415–439.

[9] M. Bowerman, Why can't you "open" a nut or "break" a cooked noodle? learning covert object categories in action word meanings, in: Building object categories in developmental time, Psychology Press, 2005, pp. 227–262.

[10] A. Majid, J. S. Boster, M. Bowerman, The cross-linguistic categorization of everyday events: A study of cutting and breaking, Cognition 109 (2008) 235–250. URL: https://www.sciencedirect.com/science/article/pii/S0010027708001911. doi:https://doi.org/10.1016/j.cognition.2008.08.009.

[11] M. Moneglia, G. Gagliardi, L. Gregori, A. Panunzi, S. Paladini, A. Williams, La variazione dei verbi generali nei corpora di parlato spontaneo. l'ontologia imagact, in: Proceedings of the VIIth GSCP International Conference: Speech and Corpora, 2012, pp. 406–411.

[12] M. Moneglia, M. Monachini, O. Calabrese, A. Panunzi, F. Frontini, G. Gagliardi, I. Russo, The imagact cross-linguistic ontology of action. a new infrastructure for natural language disambiguation., in: LREC, 2012, pp. 2606–2613.

[13] M. Bowerman, S. Choi, et al., Shaping meanings for language: Universal and language-specific in the acquisition of spatial semantic categories, Language acquisition and conceptual development 3 (2001) 475–511.

[14] Y. Wilks, N. Ide, Making sense about sense. word sense disambiguation, algorithms and applications (pp. 47–73), 2007.

[15] J. Tiedemann, S. Thottingal, Opus-mt–building open translation services for the world, in: Proceedings of the 22nd Annual Conference of the European Association for Machine Translation, European Association for Machine Translation, 2020.

[16] J. Tiedemann, M. Aulamo, D. Bakshandaeva, M. Boggia, S.-A. Grönroos, T. Nieminen, A. Raganato, Y. Scherrer, R. Vazquez, S. Virpioja, Democratizing machine translation with opus-mt, arXiv preprint arXiv:2212.01936 (2022).

[17] Z.-Y. Dou, G. Neubig, Word alignment by fine-tuning embeddings on parallel corpora, in: Conference of the European Chapter of the Association for Computational Linguistics (EACL), 2021.