Proceedings of the Globalex Workshop on Linked Lexicography, pages 92–97
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

NUIG at TIAD: Combining Unsupervised NLP and Graph Metrics for Translation Inference

John P. McCrae and Mihael Arcan
Data Science Institute, Insight Centre for Data Analytics
National University of Ireland Galway
john@mccr.ae, mihael.arcan@insight-centre.org

Abstract

In this paper, we present the NUIG system at the TIAD shard task. This system includes graph-based metrics calculated using novel algorithms, with an unsupervised document embedding tool called ONETA and an unsupervised multi-way neural machine translation method. The results are an improvement over our previous system and produce the highest precision among all systems in the task as well as very competitive F-Measure results. Incorporating features from other systems should be easy in the framework we describe in this paper, suggesting this could very easily be extended to an even stronger result.

Keywords: translation inference, machine translation, multiway translation, document embeddings

1. Introduction

Translation inference is the task of inferring new translations between a pair of languages, based on existing translations to one or more pivot language. In the particular context of the TIAD task (Gracia et al., 2019), there is a graph of translations shown in Figure 1 available from the Apertium project (Forcada et al., 2011) and the goal is to use this graph of translations to infer missing links (shown with dotted lines), in particular between English, French and Portuguese. This year, we combined two systems that had participated in a previous task (Arcan et al., 2019; McCrae, 2019) and show that this combination can improve the results. This combination consists of an unsupervised cross-lingual document embeddings system called Orthonormal Explicit Topic Analysis (McCrae et al., 2013, ONETA) and the results of unsupervised machine translation using the multi-way neural machine translation (NMT) approach (Ha et al., 2016). We also further extended this system by developing a new methodology of analysing the graph to find candidates and we show that most of the candidates (74.5%) that are likely to be correct are at a graph distance of 2, that is they are discoverable using only a single pivot translation, while quite a large amount of translations cannot be inferred using the graph (23.1%). This shows that the use of more sophisticated graph metrics is unlikely to gain more improvement in this task and that attention should instead be directed to unsupervised NLP techniques. We also analyzed the provided reference data and found that the data seems to diverge quite distinctly from the training data, suggesting that there may be a need to look for more robust methods of evaluation for future editions of this task.

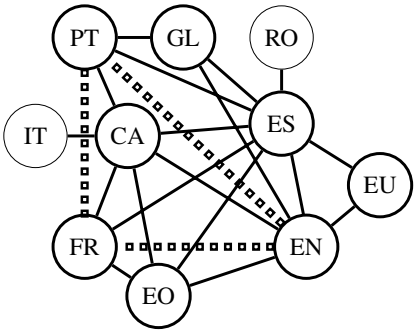

Figure 1: Languages available in the Apertium training data (solid lines) and language pairs to be inferred in the translation graph (dotted lines)

2. Methodology

2.1. Graph Extraction

One of the principal challenges of working with the TIAD data is that there are a very large number of entities and it is difficult to predict which ones are likely to be good candidates for translation inference. Following, the intuition that translations should be connected in the graph, we wish to find for a pair of languages l_1,l_2 all the entities that are connected. As the graph of all TIAD connections contains 1,053,203 nodes connected with 531,053 edges, calculating all the possible connections between the edges of the graph can be quite challenging when approached naively.

We developed the following approach to constructing the set of distances between all nodes in two languages, based on a set of translations T_{l_i,l_j} by language and a lexicon of words W_i for language l_i as shown in Algorithm 1.

The first step of this algorithm is to initialize two distance lists $dist_1$ and $dist_2$ that measure the distance between terms in l_1 or l_2 respectively and all terms in languages other than l_1 or l_2. The next step is then to iterate through all translations between languages other than l_1 and l_2 and connect the distance metrics $dist_1$ and $dist_2$. In this way, the first value of $dist_1$ contains only terms in l_1 and so they can easily be implemented as an array of associative arrays and hence kept quite sparse. Finally, we iterate through the words of l_1 and l_2 and calculate the distance between each word. This relies on the *keys* function which returns the list of terms in a third language, which have a non-infinite distance in $dist_1$ and $dist_2$. In practice, this is implemented by taking the smaller of the associative arrays associated with $dist_1$ or $dist_2$ and filtering the results according to the presence in the larger associative array. As such, while the worst-case performance of the algorithm is

Graph Distance	Correct	Total	Precision	Recall
2	30,988	40,321	0.7685	0.7452
3	838	19,820	0.0423	0.0202
4	102	24,113	0.0042	0.0025
5	38	36,848	0.0010	0.0001
6	4	37,178	0.0001	0.0000
7	5	47,686	0.0001	0.0000
8	1	42,378	0.0000	0.0000
9	0	47,739	0.0000	0.0000
10	0	39,261	0.0000	0.0000
11	1	39,246	0.0000	0.0000
12	0	29,902	0.0000	0.0000
13	0	26,441	0.0000	0.0000
14	0	19,531	0.0000	0.0000
15	0	15,484	0.0000	0.0000
16	0	10,799	0.0000	0.0000
17	0	7,549	0.0000	0.0000
18	0	4,792	0.0000	0.0000
19	0	3,163	0.0000	0.0000
20	0	2,201	0.0000	0.0000
21	0	1,134	0.0000	0.0000
22	0	528	0.0000	0.0000
23	0	258	0.0000	0.0000
24	0	52	0.0000	0.0000
25	0	3	0.0000	0.0000
Unconnected	9,606	1.3×10^9	0.0000	0.2310

Table 1: Evaluation of English-Spanish Apertium dataset based on graph distance

Algorithm 1: Distance calculation algorithm

Result: The distances between in l_1 and l_2: $dist$
for $l \in L$, $l \neq l_1, l \neq l_2$ **do**
  for $(s,t) \in T_{l_1,l}$ **do**
    $dist_1(s,t) \leftarrow 1$
  end
  for $(s,t) \in T_{l_2,l}$ **do**
    $dist_2(s,t) \leftarrow 1$
  end
end
for $l_i \in L, l_j \in L, l_i \neq l_1, l_i \neq l_2, l_i \neq l_1, l_j \neq l_2$ **do**
  for $(s,t) \in T_{l_i,l_j}$ **do**
    for $u \in W_1$ **do**
      $dist_1(u,t) \leftarrow$
      $\min(dist_1(u,t), dist_1(u,s)+1)$
    end
    for $u \in W_2$ **do**
      $dist_2(u,t) \leftarrow$
      $\min(dist_2(u,t), dist_2(u,s)+1)$
    end
  end
end
for $s \in W_1$ **do**
  for $t \in W_2$ **do**
    $dist(s,t) \leftarrow$
    $\min_{u \in \text{keys}(s,t)} dist_1(s,u) + dist_2(u,t)$
  end
end

still $\mathcal{O}(|W_1| \times |W_2| \times |W'_{1,2}|)$ where $W'_{1,2}$ is the words in all languages other than l_1 and l_2, in fact the calculation of *keys* is

$$\mathcal{O}[\min(|X_1(s)|, |X_2(t)|) \times \log\max(|X_1(s)|, |X_2(t)|)]$$

Where:

$$X_i(s) = \{u : dist_i(s,u) < \infty\}$$

In order to analyze the results of this analysis, we considered the provided Apertium training data holding out the translations for one language pair, namely English-Spanish, and the results are presented in Table 1. We see that there are 46,004 terms in the English data and 28,615 terms in the Spanish data meaning there are potentially 1.3 billion translations that can be inferred. Our algorithm found that only 496,427 of these term pairs are connected in the Apertium graph, which overlaps quite well with the correct translations in the Apertium data. In fact, 23.1% of translations from the gold standard are not connected whereas 76.9% are connected at graph distance 2, that is inferred by a single pivot translation. For this reason, we used this method as the basis for generating candidate translations, in particular, we only considered translations that were at graph distance 2 or 3, and in addition, we extracted the size of the *keys* set for each translation as it was a useful and readily available statistic.

2.2. ONETA

The OrthoNormal Explicit Topic Analysis (ONETA) methodology used in the system was not much changed

from how it was applied previously (McCrae, 2019), only this time instead of just using a single language for finding potential pivots, the results of the graph distance method were used to select all translations at distance 2 or 3. For the purpose of completeness, we will briefly recap the methodology here. ONETA aims to find a vector to represent a term satisfying

$$\phi_{\text{ONETA}}(d_i)^{\text{T}}\phi_{\text{TF-IDF}}(d_j) = \delta_{ij}$$

It does this by constructing the TF-IDF vectors for each of the words and organizing them in a matrix $\mathbf{X}$ and then the vector for ONETA can be obtained as[1]:

$$\phi_{\text{ONETA}}(d_i) = \mathbf{X}^{+}\phi_{\text{TF-IDF}}(d_j)$$

Where:

$$x_{ij} = \phi_{\text{TF-IDF}}(d_i)^{\text{T}}\phi_{\text{TF-IDF}}(d_j)$$

It was shown (McCrae et al., 2013) that this can be efficiently approximated by organizing the matrix $\mathbf{X}$ into the form

$$\mathbf{X} \simeq \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{C} \end{pmatrix}$$

And using the following form of the projection:

$$\phi_{\text{ONETA}}(d_i) = \begin{pmatrix} \mathbf{A}^{+} & -\mathbf{A}^{+}\mathbf{B}\mathbf{C}^{+} \\ \mathbf{0} & \mathbf{C}^{+} \end{pmatrix} \phi_{\text{TF-IDF}}(d_j).$$

2.3. Multi-way Neural Machine

To perform experiments on neural machine translation (NMT) models with a minimal set of parallel data, i.e. for less-resourced languages, we trained a multi-source and multi-target NMT model (Ha et al., 2016) with well-resourced language pairs. In our work, we have chosen parallel corpora in the Romance language family, i.e. Spanish, Italian, French, Portuguese, Romanian, as well as English. To train the multi-way NMT system, we used all possible language combinations within the targeted Romance language family, but excluded the English-Spanish, English-French, English-Portuguese and Portuguese-French language pair.

Neural Machine Translation Setup We used OpenNMT (Klein et al., 2017), a generic deep learning framework mainly specialised in sequence-to-sequence models covering a variety of tasks such as machine translation, summarisation, speech processing and question answering as NMT framework. Due to computational complexity, the vocabulary in NMT models had to be limited. To overcome this limitation, we used byte pair encoding (BPE) to generate subword units (Sennrich et al., 2016). BPE is a data compression technique that iteratively replaces the most frequent pair of bytes in a sequence with a single, unused byte. We used the following default neural network training parameters: two hidden layers, 500 hidden LSTM (long short term memory) units per layer, input feeding enabled, 13 epochs, batch size of 64, 0.3 dropout probability, dynamic learning rate decay, 500 dimension embeddings.

[1] $\mathbf{X}^{+}$ denotes the Moore-Penrose pseudo-inverse

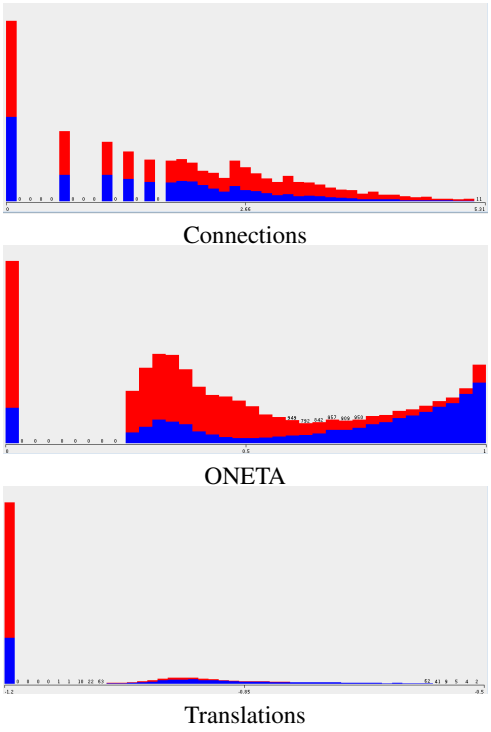

Figure 2: Distribution of the features relative to correct (blue) translations and incorrect (red) translations

Dataset for NMT training To train the multi-way model, we used the DGT (Directorate General for Translation) corpus (Steinberger et al., 2012), a publicly accessible resource provided by the European Commission to support multilingualism and the re-use of European Commission information available in 24 different European languages. The English, Spanish, French, Romanian, Italian and Portuguese languages were selected to train the multi-way NMT system, from which we extracted 200,000 translated sentences present in all six languages within the DGT corpus (Table 2).

3. Results

3.1. Results on Apertium

In order to develop and train our system, we used the available Apertium data as a gold standard. In this case, we held out the English-Spanish translation data and tried to predict the values in this dataset. From our methods, we had the following features

Distance The graph distance, either 2 or 3.

Connections The size of the *keys* set used in calculating the graph distance. To improve the result, we scaled this logarithmically.

ONETA The score coming out of ONETA. We scaled this geometrically to obtain a roughly even distribution of values.

Translation & Inverse Translation The perplexity of the translation. As the translation methodology is not

	Source		Target		
Multi-Way	# Subwords	# Uniq. Subwords	# Subwords	# Uniq. Subwords	# Lines
train	131,146,463	32,180	121,544,872	32,161	4,400,000
validation	656,154	29,380	608,006	29,408	22,000

Table 2: Dataset statistics for the DGT corpus the combined multi-way dataset used to train the translation system

symmetric we obtained two scores for English → Spanish and Spanish → English. As the perplexity naturally decreases for longer outputs, we divided it by the number of tokens in the output score.

An analysis of these features using 10-fold cross-validation compared is shown in Table 3. Note that due to the limitation of using only those translations that have a graph distance of 2 or 3, the highest recall we could achieve is 0.76 and the highest F-Measure is 0.870.

Method	Precision	Recall	F-Measure
ONETA	0.772	0.501	0.607
Connections	0.568	0.678	0.618
Translations	0.767	0.453	0.570
Random Tree	0.758	0.565	0.647
Random Forest	0.774	0.602	0.677
J48	0.822	0.599	0.693
Naïve Bayes	0.821	0.518	0.635
Logistic Regression	0.821	0.591	0.687
SVM	0.820	0.583	0.681

Table 3: Performance of our system on predicting English-Spanish Apertium data

3.2. Task Results

The official results from the organizers are reproduced in Table 4. We can see from this that in all evaluations, the system described in this paper (labelled 'NUIG'), produced the highest precision in its results. However, as we saw in the Apertium analysis we had a significant drop in recall compared to the baselines and these overall meant that the system was 2nd or 3rd in terms of F-Measure. We also note that the systems to beat ours were those based on one-time inverse consultation (Tanaka and Umemura, 1994), and it should be relatively easy to combine these results into our architecture, suggesting that we could easily obtain a much stronger result.

3.3. Discussion

The organizers of the TIAD task released a small part of the evaluation dataset, and it appears that this dataset has significant differences to the translations that form Apertium. For example, in Table 5, the translation for chestnuts are given [2], and we see that the gold standard gives 'châtaigne' as does our system but also gives two more terms 'châtaignier' and 'marronnier', which our system does not. These terms refer to chestnut as a tree and our system correctly predicts that this is a translation of 'chestnuttree' and fails to generate a translation for these terms, principally because they only occur in a single translation language pair (French-Esperanto) and so are not connected in any way to the English. More concerningly, the term 'marron' is missed in the gold standard, as well as by our system, even though this is the translation preferred by several online sources. In Figure 3, we see a relative plot of the correct terms in the released gold standard versus the graph distance calculated according to the training data. The distribution is quite different from the training data, with much less of the data being connected by a single pivot translation (that is at graph distance 1) and much more distant connections. It is especially surprising that some of the translations are at a distance of 4 or 5, which for English-Portuguese and French-Portuguese represents about 9% of the data but in the training set, while the precision of such distant links was less than 1% in the training set.

[2]This is the second example given by the organizer for this language pair

4. Conclusion

We have presented the results of our system for the TIAD task that combined unsupervised document embedding, unsupervised machine translation and graph analysis to produce a very high precision result. We have seen that the graph metrics are a good initial filtering, but that the main improvement can be achieved by incorporating metrics related to unsupervised multilingual NLP and the one-time inverse consultation method. This leads us to some obvious paths that can improve our results for future evaluations.

Acknowledgements

This publication has emanated from research supported in part by a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289_P2, co-funded by the European Regional Development Fund, as well as by the H2020 project Prêt-à-LLOD under Grant Agreement number 825182.

Bibliographical References

Arcan, M., Torregrosa, D., Ahmadi, S., and McCrae, J. P. (2019). Inferring translation candidates for multilingual dictionary generation. In *Proceedings of the 2nd Translation Inference Across Dictionaries (TIAD) Shared Task*.

Forcada, M. L., Ginestí-Rosell, M., Nordfalk, J., O'Regan, J., Ortiz-Rojas, S., Pérez-Ortiz, J. A., Sánchez-Martínez, F., Ramírez-Sánchez, G., and Tyers, F. M. (2011). Apertium: a free/open-source platform for rule-based machine translation. *Machine translation*, 25(2):127–144.

System	EN-FR			EN-PT			FR-PT		
	P	R	F	P	R	F	P	R	F
Baseline OTIC	0.67	**0.44**	**0.53**	0.64	0.38	0.48	0.74	0.54	**0.62**
Baseline word2vec	0.37	0.41	0.39	0.23	**0.39**	0.29	0.27	0.34	0.30
NUIG	**0.80**	0.35	0.49	**0.68**	0.31	0.43	**0.84**	0.40	0.54
ACOLI Baseline	0.57	0.30	0.39	0.48	0.24	0.32	0.63	0.27	0.38
ACOLI WordNet	0.59	0.18	0.28	0.54	0.13	0.21	0.62	0.15	0.24
CL - Embeddings	-	-	-	0.52	0.35	0.42	0.55	0.34	0.42
Ciclos - OTIC	-	-	-	0.57	0.44	**0.50**	0.67	**0.55**	0.60
Multi-Strategy	-	-	-	0.52	0.34	0.41	0.58	0.34	0.43

Table 4: The performance of systems in the TIAD-2020 benchmark from the organizers in terms of **P**recision, **R**ecall and **F**-Measure

English	French	Gold Standard	Our System	Graph Distance
chestnut	châtaigne	Yes	Yes	2
chestnut	châtaignier	Yes	No	∞
chestnut	marronnier	Yes	No	∞
chestnut	marron	No	No	2
chestnuttree	châtaignier	?	Yes	2

Table 5: Translations in the released gold standard and our system

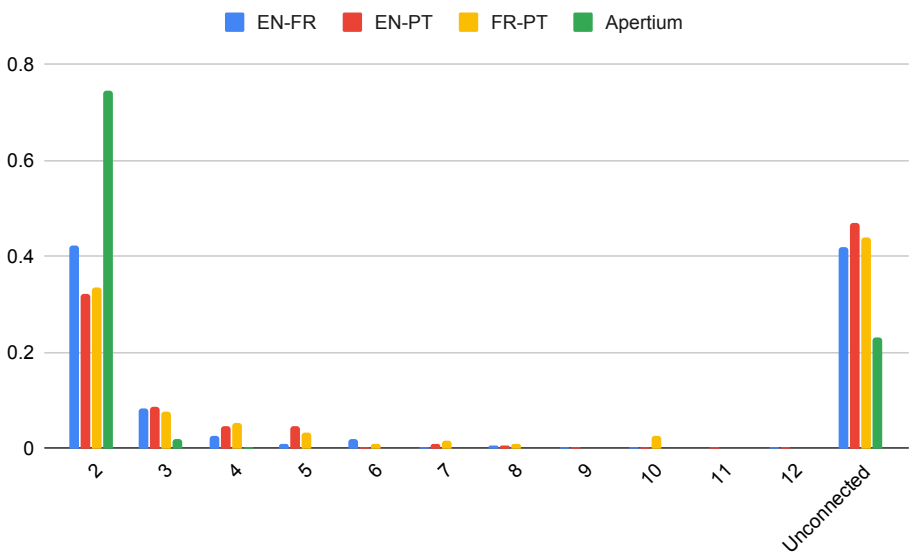

Figure 3: Distribution of translations relative to distance in training data

Gracia, J., Kabashi, B., Kernerman, I., Lanau-Coronas, M., and Lonke, D. (2019). Results of the translation inference across dictionaries 2019 shared task. pages 1–12.

Ha, T., Niehues, J., and Waibel, A. H. (2016). Toward multilingual neural machine translation with universal encoder and decoder. *CoRR*, abs/1611.04798.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 67–72.

McCrae, J., Cimiano, P., and Klinger, R. (2013). Orthonormal explicit topic analysis for cross-lingual document matching. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1732–1742.

McCrae, J. P. (2019). TIAD Shared Task 2019: Orthonormal Explicit Topic Analysis for Translation Inference across Dictionaries. In *Proceedings of the 2nd Translation Inference Across Dictionaries (TIAD) Shared Task*.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. *Proceedings of the 54th Annual Meeting of the Associa-*

tion for Computational Linguistics, abs/1508.07909.

Steinberger, R., Eisele, A., Klocek, S., Pilos, S., and Schlüter, P. (2012). DGT-TM: A freely available Translation Memory in 22 languages. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC12)*, pages 454–459, Istanbul, Turkey.

Tanaka, K. and Umemura, K. (1994). Construction of a bilingual dictionary intermediated by a third language. In *Proceedings of the 15th conference on Computational linguistics-Volume 1*, pages 297–303. Association for Computational Linguistics.