Proceedings of the Globalex Workshop on Linked Lexicography, pages 76–83
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

UNIOR NLP at MWSA Task - GlobaLex 2020: Siamese LSTM with Attention for Word Sense Alignment

Raffaele Manna, Giulia Speranza, Maria Pia di Buono, Johanna Monti
UNIOR NLP Research Group
University of Naples "L'Orientale" - Naples, Italy
{rmanna, gsperanza, mpdibuono, jmonti}@unior.it

Abstract

In this paper we describe the system submitted to the ELEXIS Monolingual Word Sense Alignment Task. We test different systems, which are two types of LSTMs and a system based on a pretrained Bidirectional Encoder Representations from Transformers (BERT) model, to solve the task. LSTM models use fastText pre-trained word vectors features with different settings. For training the models, we did not combine external data with the dataset provided for the task. We select a sub-set of languages among the proposed ones, namely a set of Romance languages, i.e., Italian, Spanish, Portuguese, together with English and Dutch. The Siamese LSTM with attention and PoS tagging (LSTM-A) performed better than the other two systems, achieving a 5-Class Accuracy score of 0.844 in the Overall Results, ranking the first position among five teams.

Keywords: Word Sense Alignment, Siamese LSTM, BERT, Text Similarity, Semantic Text Similarity, Semantic Classification

1. Introduction

As the number of lexical resources has been increased widely in the last decade, the need of integrating complementary information from several sources and knowledge bases is growing. The integration of such different information requires a process capable of aligning both monolingual and multilingual lexical resources preserving the granularity of semantic relations among senses.

The alignment of sense descriptions of lexical resources represents a crucial task for many Natural Language Processing (NLP) and Machine Translation (MT) applications. Indeed, it has been shown that aligned lexical-semantic resources can lead to better performance in NLP and MT applications than using the resources individually (Matuschek and Gurevych, 2013).

To the aim of improving large-scale and interlinked lexical-semantic resources, covering different information types and languages, many research efforts in Word Sense Alignment (WSA) area have been carried out. According to Matuschek and Gurevych (2013), WSA is the identification of pairs of senses from two lexical-semantic resources which denote the same meaning. WSA improves semantic interoperability among resources in that it allows sense matching and disambiguation, supporting an enhanced semantic processing and contributing to the creation and development of lexical-semantic resources. WSA can be performed both on multilingual data (Carpuat et al., 2006), in order to align senses among languages, and monolingual data for merging different resources (Caselli et al., 2013).

Some WSA-related shared tasks have been organized as different application scenarios. Among those, the one proposed within HLT/NAACL 2003 Workshop on Building and Using Parallel Text (Mihalcea and Pedersen, 2003) which focused on word alignment to find correspondences between words and phrases in parallel texts. Starting from a sentence aligned in a bilingual corpus in languages L1 and L2, this task aims at indicating which word token in the corpus of language L1 corresponds to which word token in the corpus of language L2.

The 1st "Monolingual Word Sense Alignment" Shared Task has been organised by the ELEXIS Project[1], as part of the GLOBALEX (Global Alliance for Lexicography)[2] - Linked Lexicography workshop at the 12th Language Resources and Evaluation Conference (LREC 2020).

The task consists of developing a system capable of predicting the semantic relation between two monolingual senses extracted from two different sources. Five types of relations among the two senses are considered: *exact* if the two entries express the same sense, *broader* if the sense of the first entry is more generic and includes the second entry's sense, *narrower* if the first entry conveys a more specific sense than the second one, *related* if the two senses are somehow connected to one-another for some aspects and *none* if the two entries express two totally different senses, so that no match is to be found.

We test different systems, namely a system based on a pretrained Bidirectional Encoder Representations from Transformers (BERT) model and two types of LSTMs, to solve the task. LSTM models use pretrained fastText features with different settings. For training the models, we did not combine external data and adjust the class distribution of the provided data set neither. We select a sub-set of languages among the proposed ones, namely Romance languages, i.e., Italian, Spanish, Portuguese, together with English and Dutch. The Siamese LSTM with attention and PoS tagging (LSTM-A) performed better than the other two systems, achieving a 5-Class Accuracy score of 0.844 in the Overall Results. The system ranked the first position among five teams in the Overall Results and ranked different positions for each language we selected.

The remainder of the paper is organized as follows: first, we introduce some related work (Section 2), then, in Section 3, we describe the dataset provided by the task organisers and subsequently discuss the implemented systems (Section 4).

[1] https://elex.is/

[2] https://globalex.link/

Finally, we comment system results and present our conclusion and future work, respectively in Section 5 and 6.

2. Related Works

Previous works related to WSA mainly adopt two approaches: similarity-based and graph-based or a combination of both.

Niemann and Gurevych (2011) use a two-step approach to align WordNet noun synsets and Wikipedia articles using the Personalized PageRank (PPR) algorithm (Agirre and Soroa, 2009) and a word overlap measure, reporting a performance of 0.78 F1-Measure and 94.5% accuracy.

Meyer and Gurevych (2011) align Wiktionary and WordNet using similarity of glosses, cosine (COS) or personalized page rank (PPR) similarity, reaching a F1 of 0.661 with the COS & PPR method.

In order to semi-automatically align GermaNet with sense definitions from Wiktionary, Henrich et al. (2011) use an approach based on bag of words and word overlap.

Laparra et al. (2010) make use of a shortest path algorithm (SSI-Dijkstra+) to align FrameNet lexical units with WordNet synsets.

The graph-based approach is applied in Matuschek and Gurevych (2014) who use Dijkstra-WSA algorithm (Matuschek and Gurevych, 2013) to calculate a distance-based similarity measure between word senses for aligning WordNet and OmegaWiki, WordNet and Wiktionary, Wiktionary and Wikipedia (English) and Wiktionary-Wikipedia (German), modelling different aspects of sense similarity by applying machine learning, outperforming the state of the art.

Recently, Ahmadi et al. (2019) proposes a textual and semantic similarity method with a weighted bipartite b-matching algorithm (WBbM) to align WordNet and Wiktionary.

In a way, the task of the word sense alignment can be compared to the task of defining and computing the similarity between two texts and, in particular, between two sentences. Among different construction methods and selection of the learning features and algorithms used, one of the best performing state of the art models is a Siamese adaptation of the Long Short-Term Memory (LSTM) network.

The Siamese network (Bromley et al., 1994) is an architecture for non-linear metric learning with similarity information. The Siamese network learns representations that incorporate the invariance and selectivity purposes through explicit information about similarity and dissimilarity between pairs of objects. One of the first research to adopt a Siamese LSTM architecture for labeled textual pairs of variable-length sequences is presented by Mueller and Thyagarajan (2016). In this work, a LSTM model with Siamese architecture is applied to assess semantic similarity between sentences. They provide word-vectors supplemented with synonymic information to the LSTMs, which use a fixed size vector to encode the underlying meaning expressed in a sentence.

Neculoiu et al. (2016) show that the bidirectional LSTM with a Siamese architecture achieves good results in learning a similarity metric on variable length character sequences in the task of job title normalization. The model projects variable length strings into a fixed-dimensional embedding space by using only information about the similarity between pairs of strings.

3. Dataset

For the ELEXIS monolingual WSA task, training data from different dictionaries and linguistic resources are available in several languages: Basque, Bulgarian, Danish, Dutch, English[3], Estonian, German, Hungarian, Irish, Italian, Portuguese, Russian, Serbian, Slovene and Spanish.

For each language, the organisers have provided a definitive training set containing the lemma shared between the two entries of the dictionaries, the PoS of the entries, the definition (gloss) of the sense of the first entry, the definition (gloss) of the sense of the second entry and the label indicating the relation between the two senses (*exact*, *broader*, *narrower*, *related* or *none*). A test dataset without the labels of the relation upon which to test the model is also provided (Ahmadi et al., 2020).

The following examples, extracted from the English_nuig training dataset, show data pairs for some of the relation types[4] between the glosses for the lemma *follow*, PoS-tagged as *verb*.

SOURCE: Princeton English WordNet (a) - Webster's 1913 Dictionary (b).

1. Type of relation: *exact*
   (a) *to be the product or result*
   (b) *to result from, as an effect from a cause, or an inference from a premise*

2. Type of relation: *narrower*
   (a) *choose and follow; as of theories, ideas, policies, strategies or plans*
   (b) *to copy after; to take as an example*

3. Type of relation: *related*
   (a) *travel along a certain course*
   (b) *to walk in, as a road or course; to attend upon closely, as a profession or calling*

4. Type of relation: *none*
   (a) *imitate in behavior; take as a model*
   (b) *to succeed in order of time, rank, or office*

In Table 1 we report the information about the training data composition provided for the languages (Dutch, English[5], Italian, Portuguese and Spanish) we chose to train our system on (section 4). The datasets in the different languages are not homogeneous in their respective sizes nor in the lemmas' PoS coverage.

[3] For the English language two datasets have been provided: the English_nuig containing glosses taken from the Princeton English WordNet and the Webster's 1913 dictionary, and the English_kd, which contains glosses from the Password and Global dictionary series provided by K Dictionaries through Lexicala.

[4] For the verb *follow*, taken as example, no *broader* relation is found in the dataset.

[5] We chose to use the English_nuig dataset.

For each language we provide the number of Aligned and Different senses according to the PoS of the lemma (e.g., (V), (N)). The Aligned Sense refers to the several combinations derived from the alignment between the first gloss (sense) coming from the first source and the second gloss (sense) from the second source; whereas the Different Sense is the total number of unaligned glosses coming from both dictionaries.

Some languages do not present some of the possible PoS, e.g., Italian which includes only verbs and nouns and no lemmas belonging to other categories.

Indeed, the dataset analysis reveals that some PoS are much more frequent in some languages than in others. The most frequent PoS attributed to lemmas in the English, Italian and Spanish datasets is *verb*, whereas in Portuguese and Dutch there is a conspicuous number of lemmas Pos-tagged as *noun*. Furthermore, with the exception of the Italian dataset, where no *adjective* or *adverb* occurrences are to be found (N/A), in the other languages' datasets adjectives are more present than adverbs. Other types of PoS (e.g., adposition, affix, conjunction) are only found in the Spanish and Portuguese datasets. As far as the size of training data is concerned, the Dutch language dataset appears to be larger compared to other languages, followed by English, Spanish, Italian and Portuguese, as it is shown in the Total column in Table 1. In addition, it is worth stressing that even though the training data are imbalanced, as reported in Table 2, we did not apply any technique to adjust the class distribution of a data set. For all the languages investigated, the datasets show a predominance of *none* and *exact* relations if compared to the other semantic relations types selected as possible candidates in the shared task.

With reference to the combination of relation and PoS, we notice that the number of aligned *exact* senses whose lemma was PoS-tagged as *noun* is higher in all the languages, whereas the label *none* is more frequently associated to the PoS *verb* in the English dataset and to the PoS *noun* in the Italian, Spanish, Portuguese and Dutch datasets. The total number of each relation type as well as the total number of relations in each training set are also reported.

4. System Description

To address the problem of WSA, we build three different models. We first investigate the capabilities of BERT, one of the most recent language representation models, released by Google in 2018. Then, we build two models based on Siamese LSTM (LSTM and LSTM-A), which has been recently applied to solve short text similarity tasks for multiple domains and languages (de Souza et al., 2020). Those two systems use two different types of lexical-semantic information as features and different settings.

The first LSTM takes gloss pairs as input with only few preprocessing steps. Gloss pairs are represented as word vectors trained on WSA datasets and intersected with pretrained word vectors. We use this vector addition or intersection to find a set *A* containing *n* words closer to the words vectors set trained on the gloss pairs in the training data. This was useful for possibly incorporating similar or related words not present in gloss pairs (Gagliano et al., 2016). The attention mechanism is not included in the parameters of this model.

The second LSTM (LSTM-A) includes more lexical-semantic information about the glosses with respect to the one described above. Indeed, such an LSTM model gives attention only to the words in sense descriptions which present the same PoS category assigned to the lemma they refer to. In other words, given a lemma labelled as noun, e.g., *dealer* and the following two glosses which refer to the target lemma:

1. *a seller of illicit goods*
2. *one who deals; one who has to do, or has concern, with others; esp., a trader, a trafficker, a shopkeeper, a broker, or a merchant;*

The model only process the words underlined in the pair of senses, which present the same lemma PoS. Then, in this model, the attention mechanism is used.

BERT Given the novelty and popularity of BERT model in the NLP field, we decide to use and implement with no fine-tuning efforts a semantic relations classification system based on this model. For this, we have used English-BERT[6] (Eng-BERT) to predict the relations of English senses and Multilingual BERT[7] (M-BERT) to predict the relations in the other languages involved in the experiments (i.e., Dutch, Italian, Portuguese and Spanish).

English BERT (Devlin et al., 2018) is a bi-directional model based on the transformer architecture. The transformer architecture is an architecture based solely on attention mechanism.

In the context of WSA shared task, we use the uncased large version of Eng-BERT to deal with the alignment of the English senses. This version has 24 layers and 16 attention heads and generates 1024 dimension vector for each word. We use 1024 dimension vector of the Extract layer as the representation of the glosses. Our classification layer consists of a single Dense layer. The dense layer consists of 3 units and the *softmax* activation function was used. The loss function used is *binary crossentropy*. The Adam optimizer is used for training the model for 15 epochs.

Whereas, for Romance languages and Dutch, Multilingual BERT is used, it is trained on monolingual Wikipedia articles of 104 different languages. It is intended to enable Multilingual BERT fine-tuned in one language to make predictions for another language. In our research, we use the M-BERT model having 12 layers and 12 heads. This model generates 768 dimension vector for each word. We used the 768 dimension vector of the Extract layer as the representation of the glosses and a single Dense layer is used as a classification relations model. The hyperparameters used for training the model is the same as mentioned above.

LSTM Since word sense alignment is viewed as a supervised learning problem in this shared task, the model takes as input two gloss pairs having different sequence length and a label for the pair which describes the underlying similarity or semantic relation between gloss pairs.

[6] Available at: `https://github.com/google-research/bert`

[7] Available at: `https://huggingface.co/models?filter=multilingual`

Languages	Senses	V	N	ADJ	ADV	Other PoS	Total
Dutch	Aligned	4766	8958	4118	1378	N/A	19220
	Different	**514**	**1730**	**602**	**119**	**N/A**	**2965**
English_nuig	Aligned	4755	2694	810	78	N/A	8337
	Different	**1109**	**1690**	**571**	**63**	**N/A**	**3433**
Italian	Aligned	946	1022	N/A	N/A	N/A	1968
	Different	**514**	**605**	**N/A**	**N/A**	**N/A**	**1119**
Spanish	Aligned	1051	2228	991	72	112	2342
	Different	**406**	**1127**	**504**	**47**	**31**	**2084**
Portuguese	Aligned	405	807	189	9	1	1411
	Different	**111**	**361**	**144**	**12**	**1**	**629**

Table 1: Number of Different and Aligned Senses in the Training Data

Languages	Relations	V	N	ADJ	ADV	Other PoS	Total
Dutch	Exact	77	264	93	10	N/A	444
	Broader	7	40	N/A	4	N/A	51
	Narrower	9	14	5	1	N/A	29
	Related	9	24	3	4	N/A	40
	None	4664	8616	4013	1363	N/A	18656
							19220
English_nuig	Exact	230	409	149	12	N/A	800
	Broader	19	11	7	2	N/A	39
	Narrower	100	143	58	9	N/A	310
	Related	25	16	8	2	N/A	51
	None	4381	2115	588	53	N/A	7137
							8337
Italian	Exact	120	161	N/A	N/A	N/A	281
	Broader	11	22	N/A	N/A	N/A	33
	Narrower	66	43	N/A	N/A	N/A	109
	Related	54	23	N/A	N/A	N/A	77
	None	695	773	N/A	N/A	N/A	1468
							1968
Portuguese	Exact	29	103	43	2	1	178
	Broader	N/A	2	1	N/A	N/A	3
	Narrower	3	18	10	1	N/A	32
	Related	5	7	10	N/A	N/A	22
	None	368	677	125	6	N/A	1176
							1411
Spanish	Exact	129	350	160	20	12	671
	Broader	23	50	19	N/A	N/A	92
	Narrower	24	72	29	N/A	2	127
	Related	10	38	16	1	5	70
	None	865	1718	797	50	93	3523
							4483

Table 2: Type of Relations and PoS in the Training Data

In our approach, we adopt a Siamese LSTM architecture for two of our models, namely LSTM and LSTM-A. Such an architecture is based on two identical sub-networks for each LSTM model. Indeed, it has been shown that Siamese LSTM produces a mapping from a general space f variable length sequences into an interpretable representation with fixed dimensionality vector space (Mueller and Thyagarajan, 2016). Thus, each sub-network reads a gloss and generates a fixed representation. In addition, as we previously stated, for one of the LSTM models (LSTM-A) we build a model based on word vectors which represent each preprocessed input gloss, keeping only words that belong to the same PoS of the lemma whose senses must be aligned. Then, this model employs its final hidden state as a vector representation for each gloss. Afterwards, the similarity and the semantic relation brought by the labels between these representations are used as a predictor of words senses similarity.

4.1. Preprocessing

For preprocessing the glosses we perform the following steps: tokenization, gloss lowercasing, gloss cleaning and word tagging with PoS tags using tools provided by spaCy package[8].

Tokenizer First we tokenize the glosses to identify all the expressions such as dates, time, currencies, acronyms. We use a Tokenizer[9] with the default settings for the languages involved. To this, we add some custom rules (regular expressions) to match all the expressions mentioned above. In this way, we keep all these expressions as one token, so later we can normalize them reducing the vocabulary size.

Gloss Cleaning As second step, we remove the punctuation and some particular elements that appear in the glosses. In fact, in several glosses, some markers are frequent, and are used to denote the different uses of a given sense (e.g., the domain) *(Anat.)*, figurative use *(Fig.)* and more.In addition to these, several specific notations related to the lexical resources associated with glosses such as numbered lists of the word sense and any residual HTML tags have been found and removed.

PoS Tagging As a final step, for the Romance languages considered in the experiments, we tag each word/token in the glosses with PoS information. Also Dutch and English glosses are involved in this PoS tagging step.To perform this step, we use the core model packages provided by the *spaCy*. For each language involved in this task, a gloss tagging was performed.

To accomplish this and build the linguistic features to be passed to the model, the PoS category belonging to each of the lemma items present in the data is taken into consideration. Then, only tokens tagged with the same PoS information as the target lemma have been kept in the glosses.

This procedure aims at isolating, keeping and processing only semantically related words, such as synonyms, hyperonyms and more.

4.2. Siamese LSTM

Word embeddings are dense vector representations of words (Mikolov et al., 2013), capturing their semantic and syntactic information. Like many top performing semantic similarity systems, our LSTMs take as input word-vectors which have been pre-trained on an external corpus intersecting these with our own word embeddings, using fastText. Thus, the word embeddings are used for initializing the weights of the first layer (embedding layer) of our network. We use the 300-dimensional fastText word embeddings (Bojanowski et al., 2017) trained on Common Crawl and Wikipedia[10].

In the model, there are two identical LSTM networks, LSTM$_a$ and LSTM$_b$ each of which process one of the preprocessed glosses in a given pair. Both subnetworks share the same weights, in order to project both glosses to the same vector space and thus be able to make a meaningful comparison between them. So, we just focus on siamese architectures with tied weights such that LSTM$_a$ = LSTM$_b$. The LSTM model learns a mapping from the space of variable length sequences of d_{in}-dimensional vectors into $R^{d_{rep}}$ (d_{in} = 300, d_{rep} = 50). Sense similarities in the representation space are subsequently used to infer the glosses underlying semantic similarity. More concretely, each gloss (represented as a sequence of word vectors belonging to the same PoS as the lemma) $x_1,...,x_T$, is passed to the LSTM, which updates its hidden state at each sequence-index.

In some cases, especially in long sequences, RNN architectures, such as LSTM, might not be able to hold all the important information in its final hidden state. In order to intensify the important elements (e.g., words) in the final representation, we use an attention mechanism (Chi and Zhang, 2018), that combines all the intermediate hidden states using their relative importance.

The final representation of each gloss is encoded by $h_T \in R^{d_{rep}}$, the last hidden state of the model. For a given pair of glosses, our approach applies a pre-defined similarity function $g : R^{d_{rep}} \times R^{d_{rep}} \rightarrow R$ to their LSTM-representations. Then, given the LSTM gloss representations, these are use to infer the glosses' underlying semantic similarity applying a simple Manhattan similarity function.

4.3. Regularization

The parameters of the model are optimized using the Nadam method (Ruder, 2016). We use the simple but effective technique of dropout (Srivastava et al., 2014) on the recurrent units (with probability 0.15) and between layers (with probability 0.25) to prevent overfitting. Dropout prevents co-adaptation of neurons and can also be thought as a form of ensemble learning, in that for each training item a subpart of the whole network is trained. Moreover, we apply dropout to the recurrent connections of both LSTMs (Gal and Ghahramani, 2016) to avoid overfitting. Finally, we stop the training of the network, after the validation loss stops decreasing (i.e., early-stopping).

5. Results and Evaluation

The official evaluation was performed using the CodaLab platform[11]. The official evaluation metrics for the ELEXIS Monolingual Word Sense Alignment shared task are: Accuracy, Precision, Recall and F-Measure.

The organizers provided the script for evaluation, which is performed for each chosen language. Besides this language-based evaluation, an average of the scores achieved for each language is added and ranked.

In the context of the MWSA shared task, the accuracy is calculated on the basis of the matches between predicted label and the reference label on the five classes. Instead, Precision, Recall and F-Measure are considered as the accuracy in predicting the type of relations according to a binary classification. In other words, predicting a sense pair as *related*, *narrower* or *broader* when the gold standard is *exact* is considered correct. On the contrary, it is considered incorrect to predict a "positive" relation when *none* is present in the gold standard or vice versa.

[8] https://spacy.io/

[9] https://keras.io/preprocessing/text/

[10] Publicly available at: https://fasttext.cc/docs/en/pretrained-vectors.html

[11] https://competitions.codalab.org/competitions/22163

Our team UNIOR NLP ranked 1st out of five teams in the Overall Results of the scores among the selected languages[12]. These results were achieved submitting the results obtained from LSTM with attention mechanism and with augmented lexical-semantic information related to the lemma PoS category (LSTM-A).

As previously stated, the LSTM-A system is the only one we submitted officially, nevertheless we evaluated all the models. Table 3 shows the results obtained for each language by each one of our three systems and the overall results. Our best performing system in predicting the type of semantic relations between the senses is the Siamese LSTM with attention and PoS information (LSTM-A).

In fact, as shown in table 3, our model reaches a 5-Class Accuracy score of 0.844 and a 2-Class F-Measure score of 0.594 in the overall results.

Our system performs quite well for Italian and Spanish sense pairs. In both languages, we ranked 1st among four teams with a 5-Class Accuracy score of 0.766 and a 2-Class F-Measure score of 0.741 for Italian, while correspondingly 0.829 and 0.810 for Spanish.

Whereas for Portuguese our model ranked as 2nd among four teams with a 5-Class Accuracy score of 0.933 and a a 2-Class F-Measure score of 0.641.

We chose to train our system mainly on Romance languages such as Italian, Spanish and Portuguese due to their common linguistic root which makes their lexico-grammar features very similar and comparable.

In addition, we chose to include the English and Dutch languages in order to compare the system also on totally, morpho-syntatically different languages to test and compare the results. In these two languages, our system performs and predicts slightly less well than the predictions related to the group of Romance languages, at least comparing them with the predictions made by the other teams.

In fact, on the English data set, our system ranked as 4th among six teams with a 5-Class Accuracy score of 0.759 and a 2-Class F-Measure score of 0.634. In Dutch, instead, our system ranked last among six teams with a 5-Class Accuracy score of 0.931 and a 2-Class F-Measure score of 0.145.

Hence, as the results in Table 3 show, LSTM-A which also holds PoS information outperforms the semantic relations classifier based on BERT and the LSTM system fed only with word vectors.

As for the other two systems, as shown in the overall results in Table 3, LSTM predicts better than the BERT based classifier. In some cases, however, the two systems almost achieve the same promising results at least for the 5-Class Accuracy. It means that the two systems are able to predict one of the five correct relations in large datasets such as the English, Portuguese and Dutch ones.

In addition to this last explanation, we propose some ideas to clarify the BERT based results in relation to the training and testing data imbalances. Looking at the performances of the BERT based model in the table 3, we can surprisingly observe a divergence of results between English and Portuguese according to the 5-Class Accuracy. In fact, one could expect higher results for a language with more resources available as in the English case.

Instead, for Portuguese language, the BERT model is able to achieve a high accuracy performance compared to the results for English, despite English language benefiting from more data.

A possible reason could lie in the imbalance of the labels distribution between the train and test data. Considering the train data statistics shown in table 2 above and bearing in mind the different sizes of the data, we can see that the most represented label is *none*, followed by *exact* in both English and Portuguese as well as other languages. Whereas in the test sets, the *none* proportion is equal to 75.6% for English and 93.6% (almost the whole test set) for the Portuguese language.

Therefore, the BERT model is capable to manage and learn better the predominant class-label *none* in the train data and predicts more often that class-label. Thus, given the aforementioned predominance of *none* relations in the test data, the model seems to achieve higher performance for Portuguese than English. Also, if we consider the 2-class Precision for these two languages in Table 3, we can notice that the BERT model tries to generalize and predict the label *exact* and those related to that. In this, the BERT model appears to be less effective given the greater attention paid to the *none* label. Despite this, it manages to get a higher score for English than for Portuguese.

As mentioned earlier, it is worth stressing that we use a Bert based classifier without fine-tuning efforts for the context of the WSA task. This means that, tuning different parameters to tackle a word sense-alignment task, a BERT based model could achieve different results. Here, we note that our BERT based semantic relations classifier does not perform very well compared to the two LSTM models with a Siamese architecture.

6. Conclusion

We use BERT based classifier and two Siamese LSTM systems to predict semantic relations between pairs of glosses in English, Dutch, Italian, Portuguese and Spanish. Our LSTM-A[13] enriched with PoS information performs remarkably well in predicting semantic relations on the test set and ranked 1st in the official overall results. Equally, it ranked 1st in the Italian and Spanish languages. Therefore, the information provided by the PoS category of the target lemma was incisive in correctly predicting the relations for each combination of monolingual senses coming from two different lexical resources.

The results obtained in this MWSA shared task have been achieved by a system with a very widespread architecture in the state of the art related to the lexicon-semantic similarity of sentences. In the future, we plan to investigate the possibilities of applying and test BERT based systems in word sense alignment tasks.

For future work, we also intend to test our model for bilingual or multilingual word sense alignment on different re-

[12] https://competitions.codalab.org/competitions/22163#results

[13] https://github.com/unior-nlp-research-group/MWSA20

Languages	Models	5-Class A	2-Class P	2-Class R	2-Class F1
Dutch	M-BERT	0.827	0.131	0.293	0.181
	LSTM	0.847	0.181	0.344	0.238
	LSTM-A	0.931	0.455	0.086	0.145
English	Eng-BERT	0.593	0.314	0.375	0.342
	LSTM	0.658	0.473	0.593	0.526
	LSTM-A	0.759	0.586	0.692	0.634
Italian	M-BERT	0.575	0.285	0.245	0.264
	LSTM	0.726	0.633	0.789	0.703
	LSTM-A	0.766	0.729	0.754	0.741
Portuguese	M-BERT	0.803	0.122	0.309	0.175
	LSTM	0.812	0.180	0.523	0.268
	LSTM-A	0.933	0.541	0.786	0.641
Spanish	M-BERT	0.457	0.262	0.481	0.339
	LSTM	0.722	0.545	0.709	0.616
	LSTM-A	0.829	0.742	0.891	0.810
Overall Results	BERT	0.651	0.223	0.341	0.260
	LSTM	0.753	0.402	0.591	0.470
	LSTM-A	**0.844**	**0.611**	**0.642**	**0.594**

Table 3: Model Results

sources. In addition, we would also integrate other Romance languages such as Catalan, French and Romanian.

7. Acknowledgements

This work has been partially supported by Programma Operativo Nazionale Ricerca e Innovazione 2014-2020 - Fondo Sociale Europeo, Azione I.2 “Attrazione e Mobilità Internazionale dei Ricercatori” Avviso D.D. n 407 del 27/02/2018 and by POR Campania FSE 2014-2020 “Dottorati di Ricerca a Caratterizzazione Industriale”.
Authorship contribution is as follows: Raffaele Manna is author of Section 4, Giulia Speranza is author of Section 2 and 3, Maria Pia di Buono is author of Section 1 and 6, Johanna Monti is author of Section 5.

8. Bibliographical References

Agirre, E. and Soroa, A. (2009). Personalizing pagerank for word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 33–41.

Ahmadi, S., Arcan, M., and McCrae, J. (2019). Lexical sense alignment using weighted bipartite b-matching. In *2nd Conference on Language, Data and Knowledge (LDK 2019)*. NUI Galway.

Ahmadi, S., McCrae, J. P., Nimb, S., Khan, F., Monachini, M., Pedersen, B. S., Declerck, T., Wissik, T., Bellandi, A., Pisani, I., Troelsgård, T., Olsen, S., Krek, S., Lipp, V., Váradi, T., Simon, L., Győrffy, A., Tiberius, C., Schoonheim, T., Ben Moshe, Y., Rudich, M., Abu Ahmad, R., Lonke, D., Kovalenko, K., Langemets, M., Kallas, J., Dereza, O., Fransen, T., Cillessen, D., Lindemann, D., Alonso, M., Salgado, A., Sancho, J. L., Ureña-Ruiz, R.-J., Simov, K., Osenova, P., Kancheva, Z., Radev, I., Stanković, R., Perdih, A., and Gabrovšek, D. (2020). A Multilingual Evaluation Dataset for Monolingual Word Sense Alignment. In *Proceedings of the 12th Language Resource and Evaluation Conference (LREC 2020)*, Marseille, France.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., and Shah, R. (1994). Signature verification using a” siamese” time delay neural network. In *Advances in neural information processing systems*, pages 737–744.

Carpuat, M., Fung, P., and Ngai, G. (2006). Aligning word senses using bilingual corpora. *ACM Transactions on Asian Language Information Processing (TALIP)*, 5(2):89–120.

Caselli, T., Vieu, L., Strapparava, C., and Vetere, G. (2013). Aligning verb senses in two italian lexical semantic resources.

Chi, Z. and Zhang, B. (2018). A sentence similarity estimation method based on improved siamese network. *Journal of Intelligent Learning Systems and Applications*, 10(4):121–134.

de Souza, J. V. A., Oliveira, L. E. S. E., Gumiel, Y. B., Carvalho, D. R., and Moro, C. M. C. (2020). Exploiting siamese neural networks on short text similarity tasks for multiple domains and languages. In *International Conference on Computational Processing of the Portuguese Language*, pages 357–367. Springer.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Gagliano, A., Paul, E., Booten, K., and Hearst, M. A. (2016). Intersecting word vectors to take figurative language to new heights. In *Proceedings of the Fifth Workshop on Computational Linguistics for Literature*, pages 20–31.

Gal, Y. and Ghahramani, Z. (2016). A theoretically grounded application of dropout in recurrent neural networks. In *Advances in neural information processing systems*, pages 1019–1027.

Gurevych, I. and Niemann, E. (2011). The people's web meets linguistic knowledge: automatic sense alignment of wikipedia and wordnet.

Henrich, V., Hinrichs, E., and Vodolazova, T. (2011). Semi-automatic extension of germanet with sense definitions from wiktionary. In *Proceedings of the 5th Language and Technology Conference (LTC 2011)*, pages 126–130.

Laparra, E., Rigau, G., and Cuadros, M. (2010). Exploring the integration of wordnet and framenet. In *Proceedings of the 5th Global WordNet Conference (GWC 2010), Mumbai, India*.

Matuschek, M. and Gurevych, I. (2013). Dijkstra-wsa: A graph-based approach to word sense alignment. *Transactions of the Association for Computational Linguistics*, 1:151–164.

Matuschek, M. and Gurevych, I. (2014). High performance word sense alignment by joint modeling of sense distance and gloss similarity. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 245–256.

Meyer, C. M. and Gurevych, I. (2011). What psycholinguists know about chemistry: Aligning wiktionary and wordnet for increased domain coverage. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 883–892.

Mihalcea, R. and Pedersen, T. (2003). An evaluation exercise for word alignment. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and using parallel texts: data driven machine translation and beyond*, pages 1–10.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mueller, J. and Thyagarajan, A. (2016). Siamese recurrent architectures for learning sentence similarity. In *thirtieth AAAI conference on artificial intelligence*.

Neculoiu, P., Versteegh, M., and Rotaru, M. (2016). Learning text similarity with siamese recurrent networks. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 148–157.

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.