UMCC_DLSI: Multidimensional Lexical-Semantic Textual Similarity

Antonio Fernández, Yoan Gutiérrez, Alexander Chávez, Héctor Dávila, Andy González, Rainel Estrada , Yenier Castañeda

DI, University of Matanzas
Autopista a Varadero km 3 ½
Matanzas, Cuba

**Sonia Vázquez
Andrés Montoyo, Rafael Muñoz,**

DLSI, University of Alicante
Carretera de San Vicente S/N
Alicante, Spain

Abstract

This paper describes the specifications and results of UMCC_DLSI system, which participated in the first Semantic Textual Similarity task (STS) of SemEval-2012. Our supervised system uses different kinds of semantic and lexical features to train classifiers and it uses a voting process to select the correct option. Related to the different features we can highlight the resource ISR-WN[1] used to extract semantic relations among words and the use of different algorithms to establish semantic and lexical similarities. In order to establish which features are the most appropriate to improve STS results we participated with three runs using different set of features. Our best approach reached the position 18 of 89 runs, obtaining a general correlation coefficient up to 0.72.

1. Introduction

SemEval 2012 competition for evaluating Natural Language Processing (NLP) systems presents a new task called Semantic Textual Similarity (STS) (Agirre *et al.*, 2012). In STS the participating systems must examine the degree of semantic equivalence between two sentences. The goal of this task is to create a unified framework for the evaluation of semantic textual similarity modules and to characterize their impact on NLP applications.

STS is related to Textual Entailment (TE) and Paraphrase tasks. The main difference is that STS assumes bidirectional graded equivalence between the pair of textual snippets. In the case of TE the equivalence is directional (e.g. a student is a person, but a person is not necessarily a student). In addition, STS differs from TE and Paraphrase in that, rather than being a binary yes/no decision, STS is a similarity-graded notion (e.g. a student and a person are more similar than a dog and a person). This bidirectional gradation is useful for NLP tasks such as Machine Translation, Information Extraction, Question Answering, and Summarization. Several semantic tasks could be added as modules in the STS framework, "such as Word Sense Disambiguation and Induction, Lexical Substitution, Semantic Role Labeling, Multiword Expression detection and handling, Anaphora and Co-reference resolution, Time and Date resolution and Named Entity Recognition, among others"[2]

1.1. Description of 2012 pilot task

In STS, all systems were provided with a set of sentence pairs obtained from a segmented corpus. For each sentence pair, s_1 and s_2, all participants had to quantify how similar s_1 and s_2 were, providing a similarity score. The output of different systems was compared to the manual scores provided by SemEval-2012 gold standard file, which range from 5 to 0 according to the next criterions[3]:

- (5) "The two sentences are equivalent, as they mean the same thing".

[1] Integration of Semantic Resource based on WordNet.

[2] http://www.cs.york.ac.uk/semeval-2012/task6/

[3] http://www.cs.york.ac.uk/semeval-2012/task6/data/uploads/datasets/train-readme.txt

*First Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 608–616,
Montréal, Canada, June 7-8, 2012.

- (4) “The two sentences are mostly equivalent, but some unimportant details differ”.
- (3) “The two sentences are roughly equivalent, but some important information differs/missing”.
- (2) “The two sentences are not equivalent, but share some details”.
- (1) “The two sentences are not equivalent, but are on the same topic”.
- (0) “The two sentences are on different topics”.

After this introduction, the rest of the paper is organized as follows. Section 2 shows the architecture of our system and a description of the different runs. In section 3 we describe the algorithms and methods used to obtain the features for our system, and Section 4 describe the training phase. The obtained results and a discussion are provided in Section 5, and finally the conclusions and future works in Section 6.

2. System architecture and description of the runs

As we can see in Figure 1 our three runs begin with the pre-processing of SemEval 2012’s training set. Every sentence pair is tokenized, lemmatized and POS tagged using Freeling tool (Atserias *et al.*, 2006). Afterwards, several methods and algorithms are applied in order to extract all features for our Machine Learning System (MLS). Each run uses a particular group of features.

The Run 1 (MultiSemLex) is our main run. This takes into account all extracted features and trains a model with a Voting classifier composed by the following techniques: Bagging (using M5P), Bagging (using REPTree), Random SubSpace (using REPTree) and MP5. The training corpus has been provided by SemEval-2012 competition, in concrete by the Semantic Textual Similarity task.

The Runs 2 and 3 use the same classifier, but including different features. Run 2 (MultiLex) uses (see Figure 1) features extracted from Lexical-Semantic Metrics (LS-M) described in section 3.1, Lexical-Semantic Alignment (LS-A) described in section 3.2 and Sentiment Polarity (SP) described in section 3.3.

On the other hand, the Run 3 (MultiSem) uses features extracted only from Semantic Alignment (SA) described in section 3.4 and the textual edit distances named QGram-Distances.

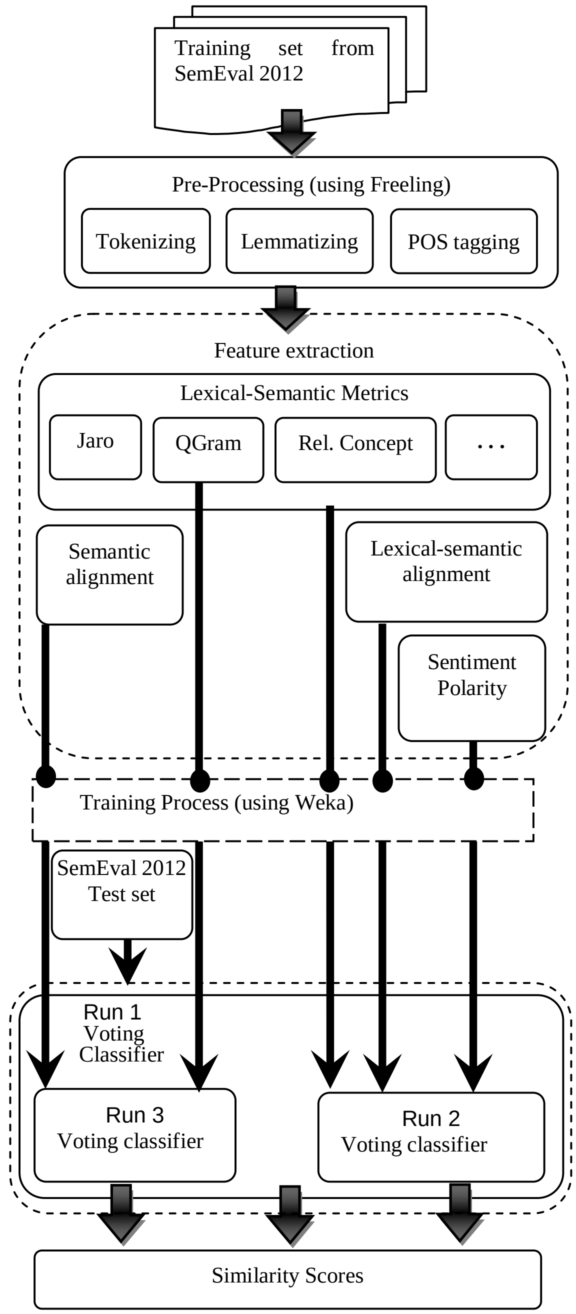

Figure 1. System Architecture.

As a result, we obtain three trained models capable to estimate the similarity value between two sentences.

Finally, we test our system with the SemEval 2012 test set (see Table 7 with the results of our three runs). The following section describes the features extraction process.

3. Description of the features used in the Machine Learning System

Sometimes, when two sentences are very similar, one sentence is in a high degree lexically overlapped by the other. Inspired by this fact we developed various algorithms, which measure the level of overlapping by computing a quantity of matching words (the quantity of lemmas that correspond exactly by its morphology) in a pair of sentences. In our system, we used lexical and semantic similarity measures as features for a MLS. Other features were extracted from a lexical-semantic sentences alignment and a variant using only a semantic alignment.

3.1. Similarity measures

We have used well-known string based similarity measures like: Needleman-Wunch (NW) (sequence alignment), Smith-Waterman (SW) (sequence alignment), Jaro, Jaro-Winkler (JaroW), Chapman-Mean-Length (CMLength), QGram-Distance (QGramD), Block-Distance (BD), Jaccard Similarity (JaccardS), Monge-Elkan (ME) and Overlap-Coefficient (OC). These algorithms have been obtained from an API (*Application Program Interface*) SimMetrics library v1.5[4] for .NET 2.0. Copyright (c) 2006 by Chris Parkinson. We obtained 10 features for our MLS from these similarity measures.

Using Levenshtein's edit distance (LED), we computed also two different algorithms in order to obtain the alignment of the phrases. In the first one, we considered a value of the alignment as the LED between two sentences and the normalized variant named NomLED. Contrary to (Tatu *et al.*, 2006), we do not remove the punctuation or stop words from the sentences, neither consider different cost for transformation operation, and we used all the operations (deletion, insertion and substitution). The second one is a variant that we named Double Levenshtein's Edit Distance (DLED). For this algorithm, we used LED to measure the distance between the sentences, but to compare the similarity between the words, we used LED again. Another feature is the normalized variant of DLED named NomDLED.

The unique difference between classic LED algorithm and DLED is the comparison of similitude between two words. With LED should be: $s[i] = t[i]$, whereas for our DLED we calculate words similarity also with LED (e.g. $DLED(s[i], t[i]) <= 2$). Values above a decision threshold (experimentally 2) mean unequal words. We obtain as result two new different features from these algorithms.

Another distance we used is an extension of LED named Extended Distance (EDx) (see (Fernández Orquín *et al.*, 2009) for details). This algorithm is an extension of the Levenshtein's algorithm, with which penalties are applied by considering what kind of operation or transformation is carried out (insertion, deletion, substitution, or non-operation) in what position, along with the character involved in the operation. In addition to the cost matrixes used by Levenshtein's algorithm, EDx also obtains the Longest Common Subsequence (LCS) (Hirschberg, 1977) and other helpful attributes for determining similarity between strings in a single iteration. It is worth noting that the inclusion of all these penalizations makes the EDx algorithm a good candidate for our approach. In our previous work (Fernández Orquín *et al.*, 2009), EDx demonstrated excellent results when it was compared with other distances as (Levenshtein, 1965), (Needleman and Wunsch, 1970), (Winkler, 1999). How to calculate EDx is briefly described as follows (we refer reader to (Fernández Orquín *et al.*, 2009) for a further description):

$$\mathrm{EDx} = \sqrt[8]{\frac{\sum_{i=0}^{l-1} V_{(O_i)} * \left(P_{\left(c1_j\right)}, P_{\left(c2_k\right)}\right)(2R_{max}+1)^{L-i}}{N}}; \quad (1)$$

Where:

V - Transformations accomplished on the words (O, I, D, S).

O - Not operations at all,

I - Insertion,

D - Deletion,

S - Substitution.

We formalize V as a vector:

$$V = \begin{Bmatrix} (0,0) : o \\ (1,0) : i \\ (0,1) : d \\ (1,1) : s \end{Bmatrix}$$

$c1$ and $c2$ - The examined words

$c1_j$ - The *j-th* character of the word $c1$

[4] http://sourceforge.net/projects/simmetrics/

$c2_k$ - The *k-th* character of the word $c2$
P - The weight of each character

We can vary all this weights in order to make a flexible penalization to the interchangeable characters.

$Pc1_j$ - The weight of characters at $c1_j$
$Pc2_k$ - The weight of characters at $c2_k$

$$j = \begin{Bmatrix} j+1 & si\ O_i \neq I \\ j & si\ O_i = I \end{Bmatrix};\ k = \begin{Bmatrix} k+1 & si\ O_i \neq D \\ k & si\ O_i = D \end{Bmatrix}$$

L - The biggest word length of the language
l - Edit operations length
O_i - Operation at (i) position
$Rmax$ - Greatest value of P ranking

$$N = \sum_{i=0}^{L-1} 2R_{max}(2R_{max}+1)^i \qquad (2)$$

As we can see in the equation (1), the term $V_{(O_i)} * \left(P_{(c1_j)}, P_{(c2_k)}\right)$ is the Cartesian product that analyzes the importance of doing *i-th* operation between the characters at *j-th* and *k-th* position

The term $(2\mathrm{R}_{\max}+1)^{\mathrm{L-i}}$ in equation (1) penalizes the position of the operations. The most to the left hand the operation is the highest the penalization is. The term N (see equation (2) normalizes the EDx into [0,1] interval. This measure is also used as a feature for the system.

We also used as a feature the Minimal Semantic Distances (Breadth First Search (BFS)) obtained between the most relevant concepts of both sentences. The relevant concepts pertain to semantic resources ISR-WN (Gutiérrez *et al.*, 2011a; 2010b), as WordNet (Miller *et al.*, 1990), WordNet Affect (Strapparava and Valitutti, 2004), SUMO (Niles and Pease, 2001) and Semantic Classes (Izquierdo *et al.*, 2007). Those concepts were obtained after having applied the Association Ratio (AR) measure between concepts and words over each sentence. The obtained distances for each resource SUMO, WordNet Affect, WordNet and Semantic Classes are named SDist, AffDist, WNDist and SCDist respectively.

ISR-WN, takes into account different kind of labels linked to WN: Level Upper Concepts (SUMO), Domains and Emotion labels. In this work, our purpose is to use a semantic network, which links different semantic resources aligned to WN. After several tests, we decided to apply ISR-WN. Although others resources provide different semantic relations, ISR-WN has the highest quantity of semantic dimensions aligned, so it is a suitable resource to run our algorithm.

Using ISR-WN we are able to extract important information from the interrelations of four ontological resources: WN, WND, WNA and SUMO. ISR-WN resource is based on WN1.6 or WN2.0 versions. In the last updated version, Semantic Classes and SentiWordNet were also included. Furthermore, ISR-WN provides a tool that allows the navigation across internal links. At this point, we can discover the multidimensionality of concepts that exists in each sentence. In order to establish the concepts associated to each sentence we apply Relevant Semantic Trees (Gutiérrez *et al.*, 2010a; Gutiérrez *et al.*, 2011b) approach using the provided links of ISR-WN. We refer reader to (Gutiérrez *et al.*, 2010a) for a further description.

3.2. Lexical-Semantic alignment

Another algorithm that we created is the Lexical-Semantic Alignment. In this algorithm, we tried to align the sentences by its lemmas. If the lemmas coincide we look for coincidences among parts of speech, and then the phrase is realigned using both. If the words do not share the same part of speech, they will not be aligned. Until here, we only have taken into account a lexical alignment. From now on, we are going to apply a semantic variant. After all the process, the non-aligned words will be analyzed taking into account its WorldNet's relations (synonymy, hyponymy, hyperonymy, derivationally – related – form, similar-to, verbal group, entailment and cause-to relation); and a set of equivalencies like abbreviations of months, countries, capitals, days and coins. In the case of the relation of hyperonymy and hyponymy, the words will be aligned if there is a word in the first sentence that is in the same relation (hyperonymy or hyponymy) of another one in the second sentence. For the relations of "cause-to" and "implication" the words will be aligned if there is a word in the first sentence that causes or implicates another one of the second sentence. All the other types of relations will be carried out in bidirectional way, that is, there is an alignment if a word of the first sentence is a synonymous of another one belonging to the second one or vice versa. Finally, we obtain a value we called alignment relation. This value is calculated as $FAV = NAW \,/\, NWSP$. Where FAV is the final alignment value, NAW is the number of aligned word and $NWSP$ is the number of words of the shorter phrase. This value is also another feature for our system.

3.3. Sentiment Polarity Feature

Another feature is obtained calculating SentiWordNet Polarities matches of the analyzed sentences (see (Gutiérrez *et al.*, 2011c) for detail). This analysis has been applied from several dimensions (WordNet, WordNet Domains, WordNet Affect, SUMO, and Semantic Classes) where the words with sentimental polarity offer to the relevant concepts (for each conceptual resource from ISR-WN (e.g. WordNet, WordNet Domains, WordNet Affect, SUMO, and Semantic Classes)) its polarity values. Other analysis were the integration of all results of polarity in a measure and further a voting process where all polarities output are involved (for more details see (Fernández *et al.*, 2012)).

The final measure corresponds to $PV = PolS_1 + PolS_2$, where $PolS_1$ is a polarity value of the sentence S_1 and $PolS_2$ is a polarity value of the sentence S_2. The negative, neutral, and positive values of polarities are represented as -1, 0 and 1 respectively.

3.4. Semantic Alignment

This alignment method depends on calculating the semantic similarity between sentences based on an analysis of the relations, in ISR-WN, of the words that fix them.

First, the two sentences are pre-processed with Freeling and the words are classified according to their parts of speech (noun, verb, adjective, and adverbs.).

We take 30% of the most probable senses of every word and we treat them as a group. The distance between two groups will be the minimal distance between senses of any pair of words belonging to the group. For example:

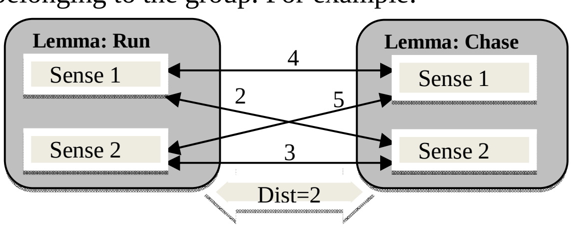

Figure 2. Minimal Distance between “Run” and “Chase”.

In the example of Figure 2 the $Dist = 2$ is selected for the pair “Run-Chase”, because this pair has the minimal cost=2.

For nouns and the words that are not found in WordNet like common nouns or Christian names, the distance is calculated in a different way. In this case, we used LED.

Let's see the following example:

We could take the pair 99 of corpus MSRvid (from training set) with a litter of transformation in order to a better explanation of our method.

Original pair

<u>A:</u> A polar bear is running towards a group of walruses.

<u>B:</u> A polar bear is chasing a group of walruses.

Transformed pair:

<u>A_1:</u> A polar bear runs towards a group of cats.

<u>B_1:</u> A wale chases a group of dogs.

Later on, using the algorithm showed in the example of Figure 2, a matrix with the distances between all groups of both sentences is created (see Table 1).

<u>GROUPS</u>	polar	bear	runs	towards	group	cats
wale	Dist:=3	Dist:=2	Dist:=3	Dist:=5		Dist:=2
chases	Dist:=4	Dist:=3	Dist:=2	Dist:=4		Dist:=3
group					Dist:=0	
dogs	Dist:=3	Dist:=1	Dist:=4	Dist:=4		Dist:=1

Table 1. Distances between the groups.

Using the Hungarian Algorithm (Kuhn, 1955) for Minimum Cost Assignment, each group of the smaller sentence is checked with an element of the biggest sentence and the rest is marked as words that were not aligned.

In the previous example the words “toward” and “polar” are the words that were not aligned, so the number of non-aligned words is 2. There is only one perfect match: “group-group” (match with $cost = 0$). The length of the shortest sentence is 4. The Table 2 shows the results of this analysis.

Number of exact coincidences (Same)	**Total Distances of optimal Matching (Cost)**	**Number of non-aligned Words (Dif)**	**Number of lemmas of shorter sentence (Min)**
1	5	2	4

Table 2. Features extracted from the analyzed sentences.

This process has to be repeated for the verbs, nouns (see Table 3), adjectives, and adverbs. On the contrary, the tables have to be created only with the similar groups of the sentences. Table 3 shows features extracted from the analysis of nouns.

GROUPS	bear	group	cats
wale	Dist := 2		Dist := 2
group		Dist := 0	
dogs	Dist := 1		Dist := 1

Table 3. Distances between the groups of nouns.

Number of exact coincidences (SameN)	**Total Distances of optimal Matching (CostN)**	**Number of non-aligned Words (DifN)**	**Number of lemmas of shorter sentence (MinN)**
1	3	0	3

Table 4. Feature extracted the analysis of nouns.

Several attributes are extracted from the pair of sentences. Four attributes from the entire sentences, four attributes considering only verbs, only nouns, only adjectives, and only adverbs. These attributes are:

- Number of exact coincidences (Same)
- Total distance of optimal matching (Cost).
- Number of words that do not match (Dif).
- Number of lemmas of the shortest sentence (Min).

As a result, we finally obtain 20 attributes from this alignment method. For each part-of-speech, the attributes are represented adding to its names the characters N, V, A and R to represent features for nouns, verbs, adjectives, and adverbs respectively.

It is important to remark that this alignment process searches to solve, for each word from the rows (see Table 3) its respectively word from the columns.

4. Description of the training phase

For the training process, we used a supervised learning framework, including all the training set (MSRpar, MSRvid and SMTeuroparl) as a training corpus. Using 10 fold cross validation with the classifier mentioned in section 2 (experimentally selected).

As we can see in Table 5, the features: FAV, EDx, CMLength, QGramD, BD, Same, SameN, obtain values over 0.50 of correlation. The more relevant are EDx and QGramD, which were selected as a lexical base for the experiment in Run 3. It is important to remark that feature SameN and Same only using number of exact coincidences obtain an encourage value of correlation.

Feature	**Correlation**	**Feature**	**Correlation**	**Feature**	**Correlation**	**Correlation using all features (correspond to Run 1)**
FAV	0.5064	ME	0.4971	CostV	0.1517	0.8519
LED	0.4572	OC	0.4983	SameN	0.5307	
DLED	0.4782	SDist	0.4037	MinN	0.4149	
NormLED	0.4349	AffDist	0.4043	DifN	0.1132	
NormDLED	0.4457	WNDist	0.2098	CostN	0.1984	
EDx	0.596	SCDist	0.1532	SameA	0.4182	
NW	0.2431	PV	0.0342	MinA	0.4261	
SW	0.2803	Same	0.5753	DifA	0.3818	
Jaro	0.3611	Min	0.5398	CostA	0.3794	
JaroW	0.2366	Dif	0.2588	SameR	0.3586	
CMLength	0.5588	Cost	0.2568	MinR	0.362	
QGramD	0.5749	SameV	0.3004	DifR	0.3678	
BD	0.5259	MinV	0.4227	CostR	0.3461	
JaccardS	0.4849	DifV	0.2634			

Table 5. Correlation of individual features over all training sets.

We decide to include the Sentiment Polarity as a feature, because our previous results on Textual Entailment task in (Fernández *et al.*, 2012). But, contrary to what we obtain in this paper, the influence of the polarity (PV) for this task is very low, its contribution working together with other features is not remarkable, but neither negative (Table 6), So we decide remaining in our system.

In oder to select the lexical base for Run 3 (MultiSem, features marked in bold) we compared the individual influences of the best lexical features (EDx, QGramD, CMLength), obtaining the 0.82, 0.83, 0.81 respectively (Table 6). Finally, we decided to use QGramD.

The conceptual features SDist, AffDist, WNDist, SCDist do not increase the similarity score, this is due to the generality of the obtained concept, losing the essential characteristic between both sentences. Just like with PV we decide to keep them in our system.

As we can see in Table 5, when all features are taking into account the system obtain the best score.

<table>
<tr><th>Feature</th><th colspan="9">Pearson (MSRpar, MSRvid and SMTeuroparl)</th></tr>
<tr><td>SDist</td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td rowspan="28">0.8509</td></tr>
<tr><td>AffDist</td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td></tr>
<tr><td>WNDist</td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td></tr>
<tr><td>SCDist</td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td></tr>
<tr><td>EDx</td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td></td><td></td><td rowspan="24">0.8507</td></tr>
<tr><td>PV</td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td rowspan="2"></td><td style="background:#bbb"></td><td style="background:#bbb"></td></tr>
<tr><td>QGramD</td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td><td></td><td style="background:#bbb"></td><td rowspan="22">0.8491</td></tr>
<tr><td>CMLength</td><td style="background:#bbb"></td><td style="background:#bbb"></td><td rowspan="21">0.8075</td><td style="background:#bbb"></td><td style="background:#bbb"></td><td style="background:#bbb"></td></tr>
<tr><td>Same</td><td rowspan="4">0.7043</td><td rowspan="20">0.795</td><td rowspan="20">0.829</td><td rowspan="20">0.8302</td><td rowspan="20">0.8228</td></tr>
<tr><td>Min</td></tr>
<tr><td>Dif</td></tr>
<tr><td>Cost</td></tr>
<tr><td>SameV</td><td rowspan="4">0.576</td></tr>
<tr><td>MinV</td></tr>
<tr><td>DifV</td></tr>
<tr><td>CostV</td></tr>
<tr><td>SameN</td><td rowspan="4">0.5975</td></tr>
<tr><td>MinN</td></tr>
<tr><td>DifN</td></tr>
<tr><td>CostN</td></tr>
<tr><td>SameA</td><td rowspan="4">0.4285</td></tr>
<tr><td>MinA</td></tr>
<tr><td>DifA</td></tr>
<tr><td>CostA</td></tr>
<tr><td>SameR</td><td rowspan="4">0.3778</td></tr>
<tr><td>MinR</td></tr>
<tr><td>DifR</td></tr>
<tr><td>CostR</td></tr>
</table>

Table 6. Features influence.

Note: Gray cells mean features that are not taking into account.

5. Result and discussion

Semantic Textual Similarity task of SemEval-2012 offered three official measures to rank the systems[5]:

1. ALL: Pearson correlation with the gold standard for the five datasets, and corresponding rank.
2. ALLnrm: Pearson correlation after the system outputs for each dataset are fitted to the gold standard using least squares, and corresponding rank.
3. Mean: Weighted mean across the five datasets, where the weight depends on the number of pairs in the dataset.
4. Pearson for individual datasets.

Using these measures, our main run (Run 1) obtained the best results (see Table 7). This demonstrates the importance of tackling this problem from a multidimensional lexical-semantic point of view.

Run	MSRpar	MSRvid	SMT-eur	On-WN	SMT-news
1	0.6205	0.8104	0.4325	0.6256	0.4340
2	0.6022	0.7709	0.4435	0.4327	0.4264
3	0.5269	0.7756	0.4688	0.6539	0.5470

Table 7. Official SemEval 2012 results.

Run	ALL	Rank	ALLnrm	RankNrm	Mean	RankMean
1	0.7213	18	0.8239	14	0.6158	15
2	0.6630	26	0.7922	46	0.5560	49
3	0.6529	29	0.8115	23	0.6116	16

Table 8. Ranking position of our runs in SemEval 2012.

The Run 2 uses a lot of lexical analysis and not much of semantic analysis. For this reason, the results for Run 2 is poorer (in comparison to the Run 3) (see Table 7) for the test sets: SMT-eur, On-WN and SMT-news. Of course, these tests have more complex semantic structures than the others. However, for test MSRpar it function better and for test MSRvid it functions very similar to Run 3.

Otherwise, the Run 3 uses more semantic analysis that Run 2 (it uses all features mentioned except feature marked in bold on Table 6) and only one lexical similarity measure (QGram-Distance). This makes it to work better for test sets SMT-eur, On-WN and SMT-news (see Table 7). It is important to remark that this run obtains important results for the test SMT-news, positioning this variant in the fifth place of 89 runs. Moreover, it is interesting to notice (Table 7) that when mixing the semantic features with the lexical one (creating Run 1) it makes the system to improve its general results, except for the test: SMT-eur, On-WN and SMT-news in comparison with Run 3. For these test sets seem to be necessary more semantic analysis than lexical in order to improve similarity estimation. We assume that Run 1 is non-balance according to the quantity of lexical and semantic features, because this run has a high quantity of

[5] http://www.cs.york.ac.uk/semeval-2012/task6/index.php?id=results-update

lexical and a few of semantic analysis. For that reason, Run 3 has a better performance than Run 1 for these test sets.

Even when the semantic measures demonstrate significant results, we do not discard the lexical help on Run 3. After doing experimental evaluations on the training phase, when lexical feature from QGram-Distance is not taken into account, the Run 3 scores decrease. This demonstrates that at least a lexical base is necessary for the Semantic Textual Similarity systems.

6. Conclusion and future works

This paper introduced a new framework for recognizing Semantic Textual Similarity, which depends on the extraction of several features that can be inferred from a conventional interpretation of a text.

As mentioned in section 2 we have conducted three different runs, these runs only differ in the type of attributes used. We can see in Table 7 that all runs obtained encouraging results. Our best run was placed between the first 18th positions of the ranking of Semeval 2012 (from 89 Runs) in all cases. Table 8 shows the reached positions for the three different runs and the ranking according to the rest of the teams.

In our participation, we used a MLS that works with features extracted from five different strategies: String Based Similarity Measures, Semantic Similarity Measures, Lexical-Semantic Alignment, Semantic Alignment, and Sentiment Polarity Cross-checking.

We have conducted the semantic features extraction in a multidimensional context using the resource ISR-WN, the one that allowed us to navigate across several semantic resources (WordNet, WordNet Domains, WordNet Affect, SUMO, SentiWorNet and Semantic Classes).

Finally, we can conclude that our system performs quite well. In our current work, we show that this approach can be used to correctly classify several examples from the STS task of SemEval-2012. Comparing with the best run (UKP_Run2 (see Table 9)) of the ranking our main run has very closed results. In two times we increased the best UKP's run (UKP_Run 2), for MSRvid test set in 0.2824 points and for On-WN test set in 0.1319 points (see Table 10).

Run	ALL	Rank	ALLnrm	RankNrm	Mean	RankMean
(UKP) Run 2	0.8239	1	0.8579	2	0.6773	1

Table 9. The best run of SemEval 2012.

It is important to remark that we do not expand any corpus to train the classifier of our system. This fact locates us at disadvantage according to other teams that do it.

Run	ALL	MSRpar	MSRvid	SMT-eur	On-WN	SMT-news
(UKP) Run 2	0.8239	0.8739	0.528	0.6641	0.4937	0.4937
(Our) Run 1	0.721	0.6205	0.8104	0.4325	0.6256	0.434

Table 10. Comparison of our distance with the best.

As future work we are planning to enrich our semantic alignment method with Extended WordNet (Moldovan and Rus, 2001), we think that with this improvement we can increase the results obtained with texts like those in On-WN test set.

Acknowledgments

This paper has been supported partially by Ministerio de Ciencia e Innovación - Spanish Government (grant no. TIN2009-13391-C04-01), and Conselleria d'Educación - Generalitat Valenciana (grant no. PROMETEO/2009/119 and ACOMP/2010/288).

Reference

Antonio Fernández, Yoan Gutiérrez, Rafael Muñoz and Andrés Montoyo. 2012. *Approaching Textual Entailment with Sentiment Polarity*. In ICAI'12 - The 2012 International Conference on Artificial Intelligence, Las Vegas, Nevada, USA.

Antonio Celso Fernández Orquín, Díaz Blanco Josval, Alfredo Fundora Rolo and Rafael Muñoz Guillena. 2009. *Un algoritmo para la extracción de características lexicográficas en la comparación de palabras*. In IV Convención Científica Internacional CIUM, Matanzas, Cuba.

Carlo Strapparava and Alessandro Valitutti. 2004. *WordNet-Affect: an affective extension of WordNet*. In Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004), Lisbon, 1083-1086.

Daniel S. Hirschberg. 1977. Algorithms for the longest common subsequence problem *Journal of the ACM*, 24: 664–675.

Dan I. Moldovan and Vasile Rus. 2001. Explaining Answers with Extended WordNet *ACL*.

Eneko Agirre, Daniel Cer, Mona Diab and Aitor Gonzalez-Agirre. 2012. *SemEval-2012 Task 6: A Pilot on Semantic Textual Similarity.* In Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012), in conjunction with the First Joint Conference on Lexical and Computational Semantics (*SEM 2012), Montreal, Canada, ACL.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross and Katherine Miller. 1990. Introduction to WordNet: An On-line Lexical Database *International Journal of Lexicography, 3(4):235-244.*

Harold W. Kuhn. 1955. The Hungarian Method for the assignment problem *Naval Research Logistics Quarterly*, 2: 83–97.

Ian Niles and Adam Pease. 2001. *Origins of the IEEE Standard Upper Ontology*. In Working Notes of the IJCAI-2001 Workshop on the IEEE Standard Upper Ontology, Seattle, Washington, USA.

Jordi Atserias, Bernardino Casas, Elisabet Comelles, Meritxell González, Lluís Padró and Muntsa Padró. 2006. *FreeLing 1.3: Syntactic and semantic services in an open source NLP library*. In Proceedings of the fifth international conference on Language Resources and Evaluation (LREC 2006), Genoa, Italy.

Marta Tatu, Brandon Iles, John Slavick, Novischi Adrian and Dan Moldovan. 2006. *COGEX at the Second Recognizing Textual Entailment Challenge*. In Proceedings of the Second PASCAL Recognising Textual Entailment Challenge Workshop, Venice, Italy, 104-109.

Rubén Izquierdo, Armando Suárez and German Rigau. 2007. A Proposal of Automatic Selection of Coarse-grained Semantic Classes for WSD *Procesamiento del Lenguaje Natural*, 39: 189-196.

Saul B. Needleman and Christian D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins *Journal of Molecular Biology*, 48(3): 443-453.

Vladimir Losifovich Levenshtein. 1965. *Binary codes capable of correcting spurious insertions and deletions of ones. Problems of information Transmission.* pp. 8-17.

William E. Winkler. 1999. *The state of record linkage and current research problems*. *Technical Report*. U.S. Census Bureau, Statistical Research Division.

Yoan Gutiérrez, Antonio Fernández, Andés Montoyo and Sonia Vázquez. 2010a. *UMCC-DLSI: Integrative resource for disambiguation task*. In Proceedings of the 5th International Workshop on Semantic Evaluation, Uppsala, Sweden, Association for Computational Linguistics, 427-432.

Yoan Gutiérrez, Antonio Fernández, Andrés Montoyo and Sonia Vázquez. 2010b. Integration of semantic resources based on WordNet *XXVI Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural*, 45: 161-168.

Yoan Gutiérrez, Antonio Fernández, Andrés Montoyo and Sonia Vázquez. 2011a. Enriching the Integration of Semantic Resources based on WordNet *Procesamiento del Lenguaje Natural*, 47: 249-257.

Yoan Gutiérrez, Sonia Vázquez and Andrés Montoyo. 2011b. *Improving WSD using ISR-WN with Relevant Semantic Trees and SemCor Senses Frequency*. In Proceedings of the International Conference Recent Advances in Natural Language Processing 2011, Hissar, Bulgaria, RANLP 2011 Organising Committee, 233--239.

Yoan Gutiérrez, Sonia Vázquez and Andrés Montoyo. 2011c. *Sentiment Classification Using Semantic Features Extracted from WordNet-based Resources*. In Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA 2.011), Portland, Oregon., Association for Computational Linguistics, 139--145.