ISSN 2395-8618

Soft Cardinality in Semantic Text Processing: Experience of the SemEval International Competitions

Sergio Jimenez, Fabio A. Gonzalez, and Alexander Gelbukh

***Abstract*—Soft cardinality is a generalization of the classic set cardinality (i.e., the number of elements in a set), which exploits similarities between elements to provide a "soft" counting of the number of elements in a collection. This model is so general that can be used interchangeability as cardinality function in resemblance coefficients such as Jaccard's, Dice's, cosine and others. Beyond that, cardinality-based features can be extracted from pairs of objects being compared to learn adaptive similarity functions from training data. This approach can be used for comparing any object that can be represented as a set or bag. We and other international teams used soft cardinality to address a series of natural language processing (NLP) tasks in the recent SemEval (semantic evaluation) competitions from 2012 to 2014. The systems based on soft cardinality have always been among the best systems in all the tasks in which they participated. This paper describes our experience in that journey by presenting the generalities of the model and some practical techniques for using soft cardinality for NLP problems.**

***Index Terms*—Similarity measure, soft computing, set cardinality, semantics, natural language processing.**

I. Introduction

THE SemEval[1] (Semantic Evaluation) competition is a series of academic workshops which aims to bring together the scientific community in the field of natural language processing (NLP) around tasks involving automatic analysis of texts. Each year, a set of challenges is proposed dealing with different aspects of the area of computational semantics attracting the attention of research groups of institutions worldwide. Each challenge follows a peer reviewing screening process ensuring the relevance, correctness, quality, and fairness of each competition. Task organizers pose an interesting challenge by providing a new dataset and a methodology for evaluating systems that address that challenge. For instance, organizers of the semantic textual similarity task (STS) provide several training datasets containing pairs of short texts labeled with a gold standard built using human annotators. Next, participating teams build systems that predict annotations in unseen test data, and organizers evaluate the performance of each system. Finally, organizers and participants describe their experiences and used approaches in peer-reviewed articles, which become de facto state of the art methods.

The authors, researchers from the Universidad Nacional de Colombia and the Centro de Investigación en Computación of the IPN in Mexico, collaborated to participate in several SemEval tasks since 2012. The core component of our participating systems is soft cardinality [1], a recently proposed approach to make the classic cardinality of set theory sensitive to the similarities and differences between the elements in a collection. This approach is particularly appropriate for addressing NLP problems because it allows finding commonalities between texts that do not share words but have words that are similar in some degree. Somehow surprisingly, systems build with this simple approach obtained impressive results in several SemEval challenges defeating considerably more complex and costly approaches. In addition, our team was the one with the highest number of participations from 2012 to 2014 also using the same core approach for addressing all tasks and always obtaining very satisfactory results.

This paper describes our experience in that journey by reviewing our participations in SemEval. Section II presents a brief description of soft cardinality, some parameterized resemblance coefficients, and the method for extracting cardinality-based feature representations. Section III presents some of the techniques and resources used for addressing NLP tasks using soft cardinality. Section IV reviews the systems and particular tasks addressed in SemEval, and a summary of the obtained results is presented. Finally in Section V we provide some concluding remarks.

II. Soft cardinality approach

The cardinality of a collection of elements is the counting of non-repeated elements within. This definition is intrinsically associated with the notion of set, which is a collection of non-repeating elements. The notation of the cardinality of a collection or set A is $|A|$. Jimenez et al. [1] proposed

Manuscript received on February 17, 2015, accepted for publication on May 27, 2015, published on June 15, 2015.

Sergio Jimenez and Fabio A. Gonzalez are with the Departamento de Ingeniería de Sistemas e Industrial of the Universidad Nacional de Colombia, Bogota, Colombia (e-mail: fagonzalezo@unal.edu.co, sergio.jimenez.vargas@gmail.com).

Alexander Gelbukh is with the Centro de Investigación en Computación, Instituto Politécnico Nacional, Mexico City, Mexico (e-mail: gelbukh@gelbukh.com).

[1]http://en.wikipedia.org/wiki/SemEval

ISSN 2395-8618

soft cardinality, which uses a notion of similarity between elements for grouping not only identical elements but similar too. That notion of similarity between elements is provided by a similarity function that compares two elements a_i and a_j returning a score in [0,1] interval having $sim(x, x) = 1$. Although, it is not necessary that sim fulfills another mathematical property aside identity, symmetry is also desirable. Thus, the soft cardinality of a collection A, whose elements $a_1, a_2, \ldots, a_{|A|}$ are comparable with a similarity function $sim(a_i, a_j)$, is denoted as $|A|_{sim}$. This soft cardinality is given by the following expression:

$$|A|_{sim} = \sum_{i=1}^{|A|} \frac{w_{a_i}}{\sum_{j=1}^{|A|} sim(a_i, a_j)^p} \quad (1)$$

It is trivial to see that $|A| = |A|_{sim}$ if either $p \to \infty$ or when the function sim is a crisp comparator, i.e., one that returns 1 for identical elements and 0 otherwise. This property shows that soft cardinality generalizes classic cardinality and that the parameter p controls its degree of "softness", the default value is $p = 1$. The values w_{a_i} are optional "importance" weights associated with each element a_i, by default those weights can be assigned to 1.

A. Inferring intersection cardinality

The soft cardinality of the intersection of two collections cannot be calculated directly from $A \cap B$ because the intersection operator is inherently crisp. This means that, if there are no common elements between A and B, their intersection is empty, and so its soft cardinality is 0. The following definition allows inferring the soft cardinality of the intersection through soft cardinalities of each collection and their union.

Let A and B be two collections, the soft cardinality of their intersection is $|A \cap B|_{sim} = |A|_{sim} + |B|_{sim} - |A \cup B|_{sim}$. In this case, the operator $\cup$ means *bag union*, which takes the maximum number of occurrences of the elements in each bag. Example: $\{1, 1, 2, 3\} \cup \{1, 2, 2\} = \{1, 1, 2, 2, 3\}$ [2].

This infers non-empty intersections for pairs of collections that have not common elements, but have similar elements. Once $|A \cup B|_{sim}$, $|A \cap B|_{sim}$, $|A|_{sim}$ and $|B|_{sim}$ are known, it is possible to obtain all other areas in the Venn's diagram of two sets, i.e., $|A \triangle B|_{sim} = |A \cup B|_{sim} - |A \cap B|_{sim}$, $|A \setminus B|_{sim} = |A|_{sim} - |A \cap B|_{sim}$ and $|B \setminus A|_{sim} = |B|_{sim} - |A \cap B|_{sim}$. These are the building blocks of almost any cardinality-based resemblance coefficient.

B. Cardinality-based resemblance coefficients

Since more than a century when Jaccard [3] proposed his well-known index, the classic set cardinality has been used to build similarity functions for set comparison. Basically, any cardinality-based similarity function is an algebraic combination of $|A|$, $|B|$ and either $|A \cap B|$ or $|A \cup B|$ (e.g. Jaccard, Dice [4], Tversky [5], overlap and cosine [6] coefficients). Table I shows some of the most used resemblance coefficients.

TABLE I
NAMED RESEMBLANCE COEFFICIENTS

Resemblance coefficient	$SIM(A, B) =$
Jaccard [3]	$\frac{\lvert A \cap B \rvert}{\lvert A \cup B \rvert}$
Dice or Sørensen [4]	$\frac{\lvert A \cap B \rvert}{0.5(\lvert A \rvert + \lvert B \rvert)}$
Overlap	$\frac{\lvert A \cap B \rvert}{\min(\lvert A \rvert, \lvert B \rvert)}$
Cosine or Ochiai [6]	$\frac{\lvert A \cap B \rvert}{\sqrt{\lvert A \rvert \cdot \lvert B \rvert}}$
Hamming	$\frac{1}{1 + \lvert A \triangle B \rvert}$

The simplest way to build similarity functions with soft cardinality is to replace the classic cardinality $|*|$ by soft cardinality $|*|_{sim}$. These coefficients have mathematical properties (e.g. transitivity, metric properties) that make of them a good option for many applications. When cosine coefficient is used in combination with soft cardinality, the resulting approach is conceptually similar to the soft cosine measure proposed by Sidorov et al. [7].

C. Parameterized resemblance coefficients

Some resemblance coefficients contain in its formulation parameters that allow adaptation to particular tasks. One of them is the Tversky's index [5], which was proposed as a cognitive model of similarity:

$$\begin{aligned} SIM(A, B) &= \frac{|A \cap B|}{\alpha|A \setminus B| + \beta|B \setminus A| + |A \cap B|}; \\ \alpha, \beta &\geq 0 \end{aligned}$$

There, parameters α and β control the balance of the differences between A and B. In Tversky's model, one of the sets being compared is the *referent* and the other is the *variant*, making this similarity measure asymmetric when $\alpha \neq \beta$. This asymmetry makes of Tversky's model an inclusion measure rather than a similarity measure. Nevertheless, in its original form it is still useful in text applications where the texts being compared have an ordinal relation, e.g. question-answer in question answering, query-document in information retrieval, text-hypothesis in textual entailment, text-summary in summarization, and others. In applications such as textual similarity or paraphrase detection, symmetry plays an important role. Jimenez et al. [8] proposed a symmetrization of Tversky's index in the following way:

$$\begin{aligned} SIM(A, B) &= \frac{c}{\beta(\alpha a + (1 - \alpha)b) + c} \quad (2) \\ |c| &= |A \cap B| + bias, \\ a &= \min[|A \setminus B|, |B \setminus A|], \\ b &= \max[|A \setminus B|, |B \setminus A|]. \end{aligned}$$

This formulation also re-arranges parameters α and β in a way that α controls the balance between the differences of

 http://dx.doi.org/10.17562/PB-51-9

ISSN 2395-8618

TABLE II
THE BASIC AND DERIVED FEATURE SETS FOR THE COMPARISON TWO COLLECTIONS OF WORDS.

Basic	Derived set 1	Derived set 2
$\|A\|$	$\|A \cap B\| = \|A\| + \|B\| - \|A \cup B\|$	$\max(\|A\|, \|B\|)$
$\|B\|$	$\|A \triangle B\| = \|A \cup B\| - \|A \cap B\|$	$\min(\|A\|, \|B\|)$
$\|A \cup B\|$	$\|A \setminus B\| = \|A\| - \|A \cap B\|$	$\max(\|A \setminus B\|, \|B \setminus A\|)$
	$\|B \setminus A\| = \|B\| - \|A \cap B\|$	$\min(\|A \setminus B\|, \|B \setminus A\|)$

A and B, and β controls the importance in the denominator between differences and commonalities between A and B. The additional parameter $bias$ allows removing an implicit degree of similarity between *A* and *B*, so usually $bias \leq 0$. This parameter can also be associated with the average or minimum intersections in a dataset. This coefficient generalizes Jaccard ($\alpha = \beta = 1; bias = 0$), Dice ($\alpha = \beta = 0.5; bias = 0$), overlap ($\alpha = 1; \beta = 0; bias = 0$) and Hamming ($\alpha = 1; \beta = 1; bias = 1 - |A \cap B|$).

Another generalization can be made by the observation that Dice and cosine coefficients are the ratio of $|A \cap B|$ and the arithmetic and geometric means, respectively. Therefore, the denominator can be replaced the expression of the generalized mean between $|A|$ and $|B|$:

$$SIM_p(A, B) = \frac{|A \cap B|}{0.5(|A|^p + |B|^p)^{1/p}} \quad (3)$$

Different values of the parameter p produce different known coefficients, i.e., Dice ($p = 1$), cosine ($p \to 0$) and overlap ($p \to \infty$). Other interesting values of p correspond to known means: $p = -1$ is the harmonic mean, $p = 2$ is the quadratic mean and $p \to -\infty$ is the minimum.

De-Baets and De-Meyer [9] proposed another hexaparametric generalized resemblance coefficient (a and b as in Eq. 2:

$$SIM(A, B) = \frac{\alpha a + \beta b + \delta |A \cap B|}{\alpha' a + \beta' b + \delta' |A \cap B|}$$

The values selected for parameters in resemblance coefficients are usually obtained by optimizing some criterion using training data. For example, in a dataset that consist of triples (A, B, g_{AB}) where g_{AB} is a gold standard of similarity (e.g. agreement of human judgments), the optimal set of parameters can be obtained by maximizing the correlation (Pearson or Spearman) between $SIM(A, B)$ and g_{AB} or by minimizing the mean-absolute error (MAE) or root-mean-squared error (RMSE).

D. Cardinality-based features for machine learning models

The parameterized resemblance coefficients allow the exploration and adaptation of a relatively large set of similarity functions to a particular problem. However, the space of possible formulations of similarity functions is huge. Which is the most appropriate similarity function for a particular problem is a question that can be addressed by adjusting parameters in these coefficients, but this strategy is nothing more than an arbitrary bias in the search. In this case, "a problem" means a dataset that needs to be modeling or explained by the similarity function. An exhaustive exploration of candidate similarity function is out of question given the large number of possible formulations. Genetic programming [10] can be used for this, but still the considered functions might be unable to model local non-linearities in some datasets. Using machine learning methods may be an appropriate option to address these issues.

Most machine learning algorithms builds models using a fixed features set (i.e., a vector) to represent each sample (e.g. linear regression, support vector machines, naïve Bayes, decision trees, *K*-means, etc.) Training data is a set of samples wherein each sample is associated with a target variable, a similarity score in our scenario. These labeled samples are used to construct a black box model, which is able, to some extent, to predict the target variable, and it is also able of producing predictions for unlabeled data. There is a variety of methods for obtaining these black box models including approaches whether geometric, probabilistic, algorithmic, information theoretical, among many others. This approach allows learning a similarity function adapted to the problem at hand efficiently and generally with a good level of generalization.

The proposed approach consists in extracting a fixed set of features from each pair of sample objects A and B, building a training dataset using these features, and labeling each sample with a gold-standard of similarity. Next, this training dataset is used to learn a machine learning model for the target variable. Finally, the learned model is used to provide similarity scores for other pairs of objects by extracting from them the same features set.

The proposed features for each pair of objects are based on cardinality, using either classical or soft cardinality. Thus, for a pair of objects (A, B) represented as sets (or bags), the basic set of cardinality-based features consist of $|A|$, $|B|$ and $|A \cup B|$. All other possible cardinality-based features are mathematical combinations thereof these three features. The following obvious features are the other areas in the Venn's diagram of two sets, i.e., $|A \cap B|$, $|A \triangle B|$, $|A \setminus B|$ and $|B \setminus A|$, Table II shows the basic and derived set of features described. An additional set of features aimed to enable machine learning algorithms to identify symmetrical patterns in the objects being compared is built using $\min()$ and $\max()$ functions, see "Derived set 2" in Table II.

Although, many machine learning methods requires or includes previous pre-processing steps of normalization or standardization of the features. Therefore, it makes sense to produce some features whose values are limited in a range.

ISSN 2395-8618

TABLE III
SET OF TEN EXTENDED RATIONAL FEATURES.

	Feature expression		Feature expression
#1	$\frac{\|A\|}{\|A\cup B\|}$	#6	$\frac{\|B\|-\|A\cap B\|}{\|B\|}$
#2	$\frac{\|A\|-\|A\cap B\|}{\|A\|}$	#7	$\frac{\|B\|-\|A\cap B\|}{\|A\cup B\|}$
#3	$\frac{\|A\|-\|A\cap B\|}{\|A\cup B\|}$	#8	$\frac{\|A\cap B\|}{\|B\|}$
#4	$\frac{\|A\cap B\|}{\|A\|}$	#9	$\frac{\|A\cap B\|}{\|A\cup B\|}$
#5	$\frac{\|B\|}{\|A\cup B\|}$	#10	$\frac{\|A\cup B\|-\|A\cap B\|}{\|A\cup B\|}$

Table III shows an extended set of features limited to [0,1] interval. These features are aimed to allow machine learning algorithms for learning patterns from the relative proportions of cardinality magnitudes. In the context of text applications, these rational features allow identifying patterns that are independent of the length of texts.

E. Exploring larger sets of features

Feature sets presented in the previous section have shown effective to address many natural language processing challenges at SemEval competitions. Despite their effectiveness, they seem to be arbitrary. For example, features shown in Table III are rational combinations of some of the features in Table II. Why only select these ten combinations? In fact, if the Table II contains 11 features and number 1 is added to this set, then the number of possible combinations of rational features is $12 \times 11 = 131$. With this new set of 131 features, the ten features in Table III seems to be arbitrary indeed. The reason for including number 1 in the basic feature set is thereby allowing the basic features and their inverses be included in the combined feature set, e.g. $|A \triangle B|$ and $\frac{1}{|A \triangle B|}$. Note that Jaccard index (i.e., $\frac{|A\cap B|}{|A\cup B|}$) is also included in this combined set. Let us call the basic set of features F, formally:

$$\begin{aligned} F(A,B) = & \\ & \{1, |A|, |B|, |A \cup B|, |A \cap B|, |A \triangle B|, \\ & |A \setminus B|, |B \setminus A|, \min(|A|,|B|), \max(|A|,|B|), \\ & \min(|A \setminus B|, ||B \setminus A|), \max(|A \setminus B|, ||B \setminus A|)\} \end{aligned}$$

Before providing a formal definition of the combined set of features, an additional set of basic features from different means (averages) must be considered as well. These additional features allow include Dice, cosine, and other coefficients as features too. For that, the expression of the generalized mean (see denominator at Eq. 3) can be used considering only a representative subset of the possible values for parameter p:

$$\begin{aligned} P = & \{-50, -20, -10, -4, -3, -2, -1, \\ & 0.0001, 1, 2, 3, 4, 10, 20, 50\} \end{aligned}$$

Now, the basic feature set F can be extended to F' by including all the generalized means restricted by P, between $|A|$ and $|B|$, and between $|A \setminus B|$ and $|B \setminus A|$, formally:

$$\begin{aligned} F'(A,B) = & F(A,B) \cup \\ & \{0.5(|A|^p + |B|^p)^{1/p} \,|\forall p \in P\} \cup \\ & \{0.5(|A \setminus B|^p + |B \setminus A|^p)^{1/p} \,|\forall p \in P\} \end{aligned}$$

Now, the number of features in $F'(A,B)$ is $|F(A,B)| + 2|P| = 42$ features. The combined set of features can be defined as:

$$C(A,B) = \left\{ \frac{f_1}{f_2} \,|(f_1, f_2) \in F'(A,B) \times F'(A,B) \wedge f_1 \neq f_2 \right\}$$

This combination produce $42 \times 41 = 1,722$ features in $C(A,B)$. This is a very large number of features for comparing only two set. Clearly, only subsets of this set of features are useful for particular applications. Even different datasets for a same task could require different representations. The idea is to make a selection of features (see [11] for an introductory tutorial) before using any machine learning regressor or classifier for a particular task. This allows to learn an adequate representation for the task prior to learn and adequate black-box (or even an interpretable) model for addressing the task. The optimal feature set for a particular task is very difficult to find because it would require considering $2^{1,772}$ possible subsets. Generally, using known methods only a near-optimal subset can be found, whose size is usually not too small nor too large. Jimenez et al. [12] observed that as a general rule the number of near-optimal features is between 10% and 20% of the number of available training samples. However, the larger the number of possible features explored, the higher the chances of finding smaller feature subsets. For example, Dueñas et al. [13] considering a similar feature set but also including logarithmic functions, found that the most correlated feature to the difficulty of a short-answer question was $\frac{|A \setminus B|}{\log(0.5\sqrt{|A|^2+|B|^2})}$, where A corresponds to the text of the reference answer and B to the question.

Although, we did not use these cardinality-based feature representation learned from training data in SemEval competitions, in subsequent studies showed this approach effective for lexical similarity task and in the analysis of questions for student evaluation. Therefore, we believe this approach may also be useful for other applications of NLP.

III. USING SOFT CARDINALITY FOR NLP

A. Textual similarity

The way to build a text similarity function is *i)* to select a linguistic unit to be compared (e.g. sentences), *ii)* to use a representation of the texts based in bags (e.g. bags of words, *n*-grams, dependencies, etc.), *iii)* to choose a cardinality based similarity coefficient (e.g. Jaccard's, Tversky's, De Beat's coefficients), and *iv)* to provide a pairwise similarity function SIM_{word} for comparing the elements produced by the used text representation (e.g. normalized Levenshtein similarity, nPMI [14], normalized path length in WordNet [15], etc.). The simplest example of such similarity function for sentence pairs is:

$$SIM_{sentence}(A,B) = \frac{|A \cap B|_{SIM_{word}}}{|A \cup B|_{SIM_{word}}}. \quad (4)$$

ISSN 2395-8618

The only parameter to be adjusted in Eq. 4 is p, the softness controller parameter. Jimenez et al. [16] showed that the default $p = 1$ works well for short sentences in English. However, a suitable value for p depends primarily on the range and distribution of the values returned by SIM_{word}, on the length of the texts, and on the task at hand. Clearly, any resemblance coefficient presented in Section II-B and Section II-C can be used.

It is important to note that Eq. 4 is recursive, similar to the popular Monge-Elkan measure [17], [18]. That is, the similarity function $SIM_{sentence}$ is obtained from another similarity function, SIM_{word}. This idea can be recursively used to build a similarity function $SIM_{paragraph}$ based on $SIM_{sentence}$, and so on. Thus, it is possible to build similarity functions exploiting the hierarchical structure of the text and natural language.

B. Term weights

Term weighting is a common practice in NLP to promote informative words and ignore non-informative words. For instance, the so-called stopwords are function words that can be removed of texts preserving their meaning to some extent, examples of these stopwords are *the*, *of*, *for*, etc. Removing stopwords may be interpreted as a binary weighting for the words in a text, i.e., assigning 1 for non-stopwords and 0 otherwise. However, a graded notion of informativeness has proven more effective than the binary approach. Probably the most used term-weighting schemes are tf.idf [19] and BM25 [20].

The soft cardinality allows the use of term weights at w_{a_i} in Eq. 1. It is important to note, that elements with zero weights (or close to 0) should be removed from texts because, although their contribution is 0, their similarities still interacts with the other elements affecting soft cardinality. This issue reveals the fact that most of the properties of the soft cardinality get overwritten because of term weighting. However, that weighted approach still preserves the original motivations of soft cardinality and extends its modeling capability [21].

C. Features for text comparison

In Section II-D we presented a method for extracting basic sets of cardinality-based features from a pair of texts represented as sets or bags of words. When the soft cardinality is being used in short texts, its word-to-word similarity function SIM_{word} plays a central role in the meaning of the extracted features. For instance, if the SIM_{word} compares words morphologically, then features extracted using $|*|_{SIM_{word}}$ reflect morphological features in texts. Additionally, other types of features can be extracted by modifying the set representation of text. For instance, a text A can be enriched with words taken from the dictionary definitions of the words already in A. These and others methods for feature extraction are presented in the following sections.

1) Morphological features: For extracting morphological features of texts it is only necessary to provide a $SIM_{word}(w_1, w_2)$ function based on the characters of the words. Some options are edit distance [22] (converted to a similarity function) or Jaro-Winkler similarity [23] (see [24] for a survey). Our choice was to use the Tversky symmetrized index (Eq. 2) by representing each word by 3-grams of characters, e.g. *house* is represented as {*hou*, *ous*, *use*}. The values of the parameters of the Tversky symmetrized index were obtained by building a simple text similarity function $SIM_{sentence}(A, B)$ using Dice's coefficient and soft cardinality using that function as auxiliary similarity function, i.e., $|*|_{SIM_{word}}$ by Eq. 1. Then the space of parameters were explored by hill-climbing optimizing the Pearson's correlation between the similarity score obtained $SIM_{sentence}$ and the gold standard of the SICK dataset [25]. The optimal values of the parameters were $\alpha = 1.9$, $\beta = 2.36$, $bias = -0.97$. In fact, the size of *n*-grams, $n = 3$, was also optimal for that function. The softness-control parameter of soft cardinality was optimized too, obtaining $p = 0.39$, but it is irrelevant for SIM_{word}. Thus, the proposed similarity function for comparing words is:

$$\begin{aligned} SIM_{word}(w_1, w_2) = & \\ & \frac{|w_1 \cap w_2| - 0.97}{2.36(1.9a - 0.9b) + |w_1 \cap w_2| - 0.97} \\ a &= \min[|w_1 \setminus w_2|, |w_2 \setminus w_1|] \\ b &= \max[|w_1 \setminus w_2|, |w_2 \setminus w_1|] \end{aligned} \quad (5)$$

Finally, having soft cardinality $|*|_{SIM_{word}}$ for each pair of texts A and B the features described in Section II-D or Section II-E can be obtained straightforwardly.

2) Semantic features: The proposed $SIM_{word}(w_1, w_2)$ function in previous section only exploits the superficial information of the words, therefore the extracted features using soft cardinality $|*|_{SIM_{word}}$ convey the same kind of information but at textual level. The obvious next step is to use a function of similarity of words that exploits semantic relationships between the words instead of comparing letters. In that way, the soft cardinality-based features would convey semantic information. There are several choices for that. First, knowledge-based lexical measures based on WordNet can do the job (see background section in [26].) Alternatively, distributional representations that make use of frequencies of the words taken from large corpora (see [27] for some examples) can be used for semantic lexical similarity. Recently, neural word embedding [28], [29] has become the state-of-the-art for semantic lexical similarity. The approach consists in building a predictive model for each word in the vocabulary of a large corpus based in local contexts. For this, each vocabulary word is represented as a fixed dimensional vector (usually from 100 to 300 dimensions). These representations are those that maximize the probability of generating the entire corpus. Although, the process of

ISSN 2395-8618

obtaining these representations is computationally expensive, pre-trained vectors are freely-available for use.[2] To obtain similarity scores with this approach, the cosine similarity between their vectorial representations is used.

3) ESA Features: For this set of features, we used the idea proposed by Gabrilovich and Markovitch [30] of extending the representation of a text by representing each word by its textual definition in a knowledge base, i.e., explicit semantic analysis (ESA). For that, we used as knowledge base the synset's textual definitions provided by WordNet. First, in order to determine the textual definition associated to each word, the texts were tagged using the maximum entropy POS tagger included in the NLTK.[3] Next, the Adapted Lesk's algorithm [31] for word sense disambiguation was applied in the texts disambiguating one word at the time. The software package used for this disambiguation process was *pywsd*.[4] The argument parameters needed for the disambiguation of each word are the POS tag of the target word and the entire sentence as context. Once all the words are disambiguated with their corresponding WordNet synsets, each word is replaced by all the words in their textual definition jointly with the same word and its lemma. The final result of this stage is that each text in the dataset is replaced by a longer text including the original text and some related words. The motivation of this procedure is that the extended versions of each pair of texts have more chance of sharing common words that the original texts.

Once the extended versions of the texts were available, the same features described in Section III-C1 or Section III-C2 can be obtained.

4) Features for each part-of-speech category: This set of features is motivated by the idea proposed by Corley and Mihalcea [32] of grouping words by their POS category before being compared for semantic textual similarity. Our approach provides a version of each text pair in the dataset for each POS category including only the words belonging to that category. For instance, the pair of texts {"*A beautiful girl is playing tennis*", "*A nice and handsome boy is playing football*"} produces new pairs such as: {"*beautiful*", "*nice handsome*"} for the ADJ tag, {"*girl tennis*", "*boy football*"} for NOUN and {"*is playing*", "*is playing*"} for VERB.

Again, the POS tags were provided by the NLTK's maximum entropy tagger. The 28 POS categories were simplified to nine categories in order to avoid an excessive number of features and hence sparseness; used mapping is shown in Table IV. Next, for each one of the nine new POS categories a set of features is extracted reusing again the method proposed in section II-D. The only difference consideration is the stopwords should not be removed and stemming should not be performed. The motivation for generating this feature sets grouped by POS category is that the machine learning algorithms could weight differently each category. The intuition behind this is that it is reasonable that categories such as VERB and NOUN could play a more important role for the task at hand than others such as ADV or PREP. Using these categorized features, such discrimination among POS categories can be discovered from the training data.

TABLE IV
MAPPING REDUCTION OF THE POS TAG SET

Reduced tag set	NLTK's POS tag set
ADJ	JJ,JJR,JJS
NOUN	NN,NNP,NNPS,NNS
ADV	RB,RBR,RBS,WRB
VERB	VB,VBD,VBG,VBN,VBP,VBZ
PRO	WP,WP$,PRP,PRP$
PREP	RP,IN
DET	PDT,DT,WDT
EX	EX
CC	CC

5) Features from dependencies: Syntactic soft cardinality [33], [34] extends the soft cardinality approach by representing texts as bags of dependencies instead of bags of words. Each dependency is a 3-tuple composed of two syntactically related words and the type of their relationship. For instance, the sentence "*The boy plays football*" is be represented with 3 dependencies: [**det**,"*boy*","*The*"], [**subj**,"*plays*","*boy*"] and [**obj**,"*plays*","*football*"]. Clearly, this representation distinguishes pairs of texts such as {"*The dog bites a boy*","*The boy bites a* dog"}, which are indistinguishable when they are represented as bags of words. This representation can be obtained automatically using the Stanford Parser [35], which, in addition, provides a dependency identifying the root word in a sentence.

Once the texts are represented as bags of dependencies, it is necessary to provide a similarity function between two dependency tuples in order to use soft cardinality, and hence to obtain the cardinality-based features in Table II. Such function can be obtained using the SIM_{word} function (Eq. 5) for comparing the first and second words between the dependencies and even the labels of the dependency types. Let's consider two dependencies tuples $d = [d_{dep}, d_{w_1}, d_{w_2}]$ and $p = [p_{dep}, p_{w_1}, p_{w_2}]$ where d_{dep} and p_{dep} are the labels of the dependency type; d_{w_1} and p_{w_1} are the first words on each dependency tuple; and d_{w_2} and p_{w_2} are the second words. The similarity function for comparing two dependency tuples can be a linear combination of the sim scores between the corresponding elements of the dependency tuples by the following expression:

$$sim_{dep}(d,p) = \gamma\, sim(d_{dep}, p_{dep}) + \delta\, sim(d_{w_1}, p_{w_2}) + \lambda\, sim(d_{w_2}, p_{w_2}).$$

Although, it is unusual to compare the dependencies' type labels d_{dep} and p_{dep} with a similarity function designed for words, we observed experimentally that this approach yield better overall performance in the textual relatedness task in comparison with a simple exact comparison. The optimal

[2] http://code.google.com/p/word2vec/; http://nlp.stanford.edu/projects/glove/
[3] http://www.nltk.org/
[4] https://github.com/alvations/pywsd

ISSN 2395-8618

values for the parameters $\gamma = -3$, $\delta = 10$ and $\lambda = 3$ were determined with the same methodology used in Section II-C for determining α, β and $bias$. Clearly, the fact that $\delta > \lambda$ means that the first words in the dependency tuples plays a more important role than the second ones. However, the fact that $\gamma < 0$ is counter intuitive because it means that the lower the similarity between the dependency type labels is, the larger the similarity between the two dependencies. Up to date, we have been unable to find a plausible explanation for this phenomenon.

IV. Soft cardinality at SemEval

The soft cardinality approach has been used by several teams for participating in several tasks in the recent SemEval campaigns (2012 to 2014). In SemEval, the task organizers propose a NLP task, provide datasets, and an evaluation setup that is carried out by them. This methodology ensures a fair comparison of the performance of the methods used by competitors. The participating systems that incorporated soft cardinality among their used methods have obtained very satisfactory results, obtaining in most of the cases rankings among the top systems. In this section, a brief overview of these participations is presented.

A. Semantic textual similarity

The task of automatically comparing the similarity or relatedness between pairs of texts is fundamental in NLP, which attracted the attention of many researchers in the last decade [36], [32]. This task consists in building a system able to compare pairs of texts, using (or not) training data and return graded predictions of similarity or relatedness. The system performance is evaluated by correlating its predictions against a gold standard built using human judgments in a graded scale. Table V contains a summary of the results obtained by the systems that used soft cardinality.

In 2012, soft cardinality was used for the first time [16] in the pilot of the Semantic Textual Similarity (STS) task [37]. The approach consisted in building a cardinality-based similarity function $SIM_{sentence}$ combining soft cardinality with a coefficient similar to Tversky's (see Subsection III-A.) The function SIM_{word} used for comparing pairs of words was based on *n*-grams of characters combined with the same rational coefficient used at sentence level (see Eq. 5.) The parameters p, n and those of both coefficients were obtained by looking for an optimal combination in the provided training data. Finally, tf-idf weights were associated with the words (weights w_{a_i} in Eq. 1.) This simple approach obtained an unexpected third place in the official ranking among 89 participating systems. Besides, as Table V shows, this system was pretty close to the top system, which used considerably more resources [37]. Besides, comparing the rankings obtained for individual datasets and the overall ranking (3^{rd}), it can be seen that the soft cardinality system was more consistent across different data sets than most of the other systems.

In 2013, the STS task was proposed again but with increased difficulty because no additional data was provided for training. Our 2012 approach was extended by building an additional similarity function for sentences using nPMI [14] as the comparator of words. Moreover, the predictions were obtained training a regression SVM with the features described in Subsection II-D. This system ranked 19^{th} among 89 systems. However, in addition to the official results, we discovered that the same 2012 function averaged with the new nPMI function correlated much better (4^{th}) [8].

In addition, in 2013, a pilot for the Typed Similarity task was proposed. It consisted in comparing pairs of text records associated with objects from the Europeana[5] database. Croce et al. [34] built a system based on the previously proposed *syntactic soft cardinality* [33]. This consists in representing texts as sets of triples (*word1*, *word2*, *relation*) extracted from dependency graphs, and combine them using soft cardinality with a similarity function for those triplets. This system ranked first among 15 participants.

In 2014, the task 10 at SemEval was the third STS version [38], which included additional datasets in Spanish. Lynum et al. [39] proposed a system for the data sets in English using features (among others) extracted with soft cardinality ranking first in 4 out of 6 data sets among 37 participating systems. Similarly, Jimenez et al. [40] proposed a system based on the soft cardinality for the Spanish data sets, ranking first in one of the data sets and third overall among 22 systems. This system also participated in tasks 1 [25] and 3 [41], which addressed text relatedness and similarity between different lexical levels (e.g. paragraph to sentence) respectively. In these tasks, the systems based on the soft cardinality ranked 4^{th} out of 17, and 3^{rd} out of 38 systems. The used features were a combination of the feature sets presented in Sections III-C1, III-C2, III-C3, III-C3, III-C4, and III-C5.

These results show that soft cardinality is a very competitive tool for building text similarity functions with relatively few resources, namely: a similarity function for comparing pairs of words, soft cardinality, and a cardinality-based coefficient or a regression method to learn this coefficient.

B. Textual Entailment

Textual entailment (TE) is the task that consists in determining whether or not a text entails another one. It was proposed under the name *cross-lingual textual entailment* (CLTE) [42], [43] in SemEval with the additional difficulty of having the two texts in different languages. The results obtained by the systems based on the soft cardinality that participated in this task in 2012 and 2013 are shown in Table VI. The approach consisted in providing two versions of the pair of texts, each one in a single language, using machine translations from Google translate.[6] Once in a single language,

[5]http://www.europeana.eu/

[6]https://translate.google.com

ISSN 2395-8618

TABLE V
BEST RESULTS OBTAINED BY SYSTEMS THAT USED SOFT CARDINALITY AT SEMEVAL 2012–2014 FOR TEXTUAL SIMILARITY AND RELATEDNESS (PEARSON CORRELATION)

Year	Task	Dataset	Rank	Soft Card.†	Top Sys.‡	Ref.
2012	STS	MSRpar	7^{th}/89	0.6405	0.7343	[16]
		MSRvid	9^{th}/89	0.8562	0.8803	
		SMT-eur	9^{th}/89	0.5152	0.5666	
		OnWN	3^{rd}/89	0.7109	0.7273	
		SMT-news	11^{th}/89	0.4833	0.6085	
		All (w. mean)	**3^{rd}/89**	**0.6708**	**0.6773**	
2013	STS	Headlines	30^{th}/90	0.6713	0.7838	[8]
		OnWN	7^{th}/90	0.7412	0.8431	
		FNWM	22^{th}/90	0.3838	0.5818	
		SMT	54^{th}/90	0.3035	0.4035	
		All (w. mean)	**19^{th}/90**	**0.5402**	**0.6181**	
		All (unofficial)	**4^{th}/90**	**0.5747**	**0.6181**	
	Typed sim.	**Europeana**	**1^{st}/15**	**0.7620**	**0.7620**	[34]
2014	Task 1-STS	**SICK**	**4^{th}/17**	**0.8043**	**0.8280**	[40]
	Task 3	Para2Sent	1^{st}/38	0.8370	0.837	
		Sent2Phr	6^{th}/38	0.7390	0.7770	
		Phr2Word	3^{rd}/22	0.2740	0.4150	
		Word2Sense	5^{th}/20	0.2560	0.3890	
		All (w. mean)	**3^{rd}/38**	**0.5260**	**0.5810**	
	Task 10 (en)	deft-forum	1^{st}/38	0.5305	0.5305	[39]
		deft-news	2^{nd}/37	0.7813	0.7850	
		headlines	1^{st}/37	0.7837	0.7837	
		images	1^{st}/37	0.8343	0.8343	
		OnWN	4^{th}/37	0.8502	0.8745	
		tweet-news	1^{st}/37	0.7921	0.7921	
		All (w. mean)	**3^{rd}/38**	**0.7549**	**0.7610**	
	Task 10 (es)	Wikipedia	1^{st}/22	0.7804	0.7804	[40]
		news	7^{th}/22	0.8154	0.8454	
		All (w. mean)	**3^{rd}/22**	**0.8013**	**0.8072**	

† Results for the best system using the soft cardinality. ‡ Results for the best system in competition.

the soft cardinality features explained in Subsection II-D were extracted for each text pair using the same word-to-word similarity function SIM_{word} used for STS. Finally, these features were combined by a classifier to determine the type of entailment. In 2013, Jimenez et al. [44] showed that these features are also language independent, making possible to train a single classifier using data in different languages. This approach produced (not included in the official ranking) state-of-the-art results for all CLTE datasets [45].

In 2014, the textual entailment task was proposed for the SICK dataset (Sentences Involving Compositional Knowledge) [25]. Using the same approach as in CLTE, but combining additional features from soft cardinalities obtained with word similarity functions based on WordNet, ESA and dependency graphs, the soft-cardinality system [40] ranked 3^{rd} of 18. Table VII shows the results obtained by the soft cardinality system both in textual entailment and textual relatedness sub-tasks.

C. Automatic students' answer grading

The task consisted in grading the correctness of a student answer (*SA*) to a question (*Q*) given a reference answer (*RA*) [47]. The approach of the system that used soft cardinality [45] consisted in extracting features for pairs (*SA*,*Q*), (*Q*,*RA*), (*SA*,*RA*) (again using the simple SIM_{word} word similarity function) and training with them a J48-graft tree classifier. Table VIII shows the results obtained by the soft cardinality system predicting correctness in 5 categories. In all other numbers of categories and evaluation measures, the soft cardinality system also ranked 1^{st} overall datasets [47]. Recently, Leeman-Munk et al. [48] integrated the soft cardinality approach in an experimental automatic tutoring system.

V. CONCLUSION

We presented our experience participating in SemEval competitions using soft cardinality and cardinality-based feature representations. This article describes the basic methods and particular methods for addressing textual similarity, multilingual textual similarity, typed-textual similarity, textual entailment, cross-lingual textual entailment and automatic students' answer grading. A summary of the official results obtained in SemEval challenges provides the evidence of the effectiveness of the used methods in open competition. It can be come to the conclusion that soft cardinality is a practical and effective tool to address several NLP problems. Furthermore, the soft cardinality model is general enough to be used in other domains and applications.

ISSN 2395-8618

TABLE VI
BEST RESULTS OBTAINED BY THE SYSTEMS THAT USED THE SOFT CARDINALITY AT SEMEVAL 2012–2014 FOR THE TEXTUAL ENTAILMENT TASK (ACCURACY)

Year	Task	Dataset	Rank	Soft Card.	Top Sys.	Reference
2012	CLTE	Spanish-English	5^{th}/29	0.552	0.632	[46]
		Italian-English	1^{st}/21	0.566	0.566	
		French-English	1^{st}/21	0.570	0.570	
		German-English	3^{rd}/21	0.550	0.558	
2013	CLTE	Spanish-English	1^{st}/15	0.434	0.434	[44]
		Italian-English	1^{st}/15	0.454	0.454	
		French-English	6^{th}/15	0.426	0.458	
		German-English	6^{th}/16	0.414	0.452	

TABLE VII
RESULTS FOR SEMEVAL TASK 1 IN 2014

	Entailment		Relatedness			
system	accuracy	official rank	Pearson	Spearman	MSE	official rank
UNAL-NLP_run1 (primary)	83.05%	3rd/18	0.8043	0.7458	0.3593	4th/17
UNAL-NLP_run2	79.81%	–	0.7482	0.7033	0.4487	-
UNAL-NLP_run3	80.15%	–	0.7747	0.7286	0.4081	-
UNAL-NLP_run4	80.21%	–	0.7662	0.7142	0.4210	-
UNAL-NLP_run5	83.24%	–	0.8070	0.7489	0.3550	-
ECNU_run1	83.64%	2nd/18	**0.8280**	**0.7689**	0.3250	1st/17
Stanford_run5	74.49%	12th/18	0.8272	0.7559	**0.3230**	2nd/17
Illinois-LH_run1	**84.58%**	1st/18	0.7993	0.7538	0.3692	5th/17

TABLE VIII
BEST RESULTS OBTAINED BY THE SOFT-CARDINALITY SYSTEM ON THE STUDENT RESPONSE ANALYSIS TASK AT SEMEVAL 2013 (WEIGHTED-AVERAGE F_1 IN 5 CORRECTNESS LEVELS)

Dataset	Testing group	Size	Rank	Soft Cardinality	Top System
Beetle	unseen answers	439	4^{th}/9	0.558	0.705
	unseen questions	819	4^{th}/9	0.450	0.614
SciEntsBank	unseen answers	540	4^{th}/9	0.537	0.625
	unseen questions	733	1^{st}/9	0.492	0.492
	unseen domains	4,562	1^{st}/9	0.471	0.471
F_1 weighted average		**7,093**	$\mathbf{1^{th}}$**/9**	**0.502**	**0.502**

ACKNOWLEDGMENT

The second author acknowledges the support of LACCIR R1212LAC006 under the project "Multimodal image retrieval to support medical case-based scientific literature search." The third author acknowledges the support of the Mexican Government via SNI, CONACYT, and the Instituto Politécnico Nacional, SIP-IPN grants 20152100 and 20152095.

REFERENCES

[1] S. Jimenez, F. Gonzalez, and A. Gelbukh, "Text Comparison Using Soft Cardinality," in *String Processing and Information Retrieval*, ser. LNCS, E. Chavez and S. Lonardi, Eds. Berlin, Heidelberg: Springer, 2010, vol. 6393, pp. 297–302.

[2] S. P. Jena, S. K. Ghosh, and B. K. Tripathy, "On the theory of bags and lists," *Information Sciences*, vol. 132, no. 1-4, pp. 241–254, 2001.

[3] P. Jaccard, "Etude comparative de la distribution florare dans une portion des {A}lpes et des {J}ura," *Bulletin de la Société Vaudoise des Sciences Naturelles*, pp. 547–579, 1901.

[4] L. R. Dice, "Measures of the Amount of Ecologic Association Between Species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[5] A. Tversky, "Features of similarity," *Psychological Review*, vol. 84, no. 4, pp. 327–352, 1977.

[6] Ochiai, Akira, "Zoogeographical studies on the soleoid fishes found Japan and its neighboring regions," *Jap. Soc. Sci. Fish.*, vol. 22, no. 9, pp. 526–530, 1957.

[7] G. Sidorov, A. Gelbukh, H. Gomez-Adorno, and D. Pinto, "Soft Similarity and Soft Cosine Measure: Similarity of Features in Vector Space Model," *Computacion y Sistemas*, vol. 18, no. 3, pp. 491–504, 2014.

[8] S. Jimenez, C. Becerra, and A. Gelbukh, "SOFTCARDINALITY-CORE: Improving Text Overlap with Distributional Measures for Semantic Textual Similarity," in *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task*. Atlanta, Georgia, USA: ACL, Jun. 2013, pp. 194–201.

[9] B. D. Baets, H. D. Meyer, and H. Naessens, "A class of rational cardinality-based similarity measures," *Journal of Computational and Applied Mathematics*, vol. 132, no. 1, pp. 51–69, Jul. 2001.

[10] R. Poli, W. B. Langdon, N. F. McPhee, and J. R. Koza, *A field guide to genetic programming*. Lulu. com, 2008.

[11] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[12] Jimenez, Sergio, Gonzalez, Fabio A., and Gelbukh, Alexander, "Cardinality-based lexical similarity in WordNet: Bridging the gap to neural embedding," *to appear*, 2015.

[13] Dueñas, George, Jimenez, Sergio, and Julia, Baquero, "Automatic prediction of item difficulty for short-answer questions," in *to appear*, 2015.

[14] Bouma, Gerlof, "Normalized (pointwise) mutual information in collocation extraction," in *Proceedings of the Biennial GSCL Conference*, 2009, pp. 31–40.

ISSN 2395-8618

[15] T. Pedersen, S. Patwardhan, and J. Michelizzi, "WordNet::Similarity: measuring the relatedness of concepts," in *Proceedings HLT-NAACL–Demonstration Papers*. Stroudsburg, PA, USA: ACL, 2004.

[16] S. Jimenez, C. Becerra, and A. Gelbukh, "Soft Cardinality+ ML: Learning Adaptive Similarity Functions for Cross-lingual Textual Entailment," in *First Joint Conference on Lexical and Computational Semantics (*SEM)*. Montreal, Canada: ACL, 2012, pp. 684–688.

[17] A. E. Monge and C. Elkan, "The field matching problem: Algorithms and applications," in *Proceeding of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*, Portland, OR, 1996, pp. 267–270.

[18] S. Jimenez, C. Becerra, A. Gelbukh, and F. Gonzalez, "Generalized Mongue-Elkan Method for Approximate Text String Comparison," in *Computational Linguistics and Intelligent Text Processing*, ser. Lecture Notes in Computer Science, A. Gelbukh, Ed. Springer, Jan. 2009, no. 5449, pp. 559–570.

[19] G. Salton, *Introduction to modern information retrieval*. McGraw-Hill, 1983.

[20] S. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford, "Okapi at TREC-3," in *Proceedings of the Third Text REtrieval Conference (TREC 1994)*, Gaithersburg, USA, 1994, pp. 109–126.

[21] Jimenez, Sergio, Gonzalez, Fabio A., and Gelbukh, Alexander, "Mathematical properties of Soft Cardinality: Enhancing Jaccard, Dice and cosine similarity measures with element-wise distance," *to appear*, 2015.

[22] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707–710, 1966.

[23] W. E. Winkler, "The State of Record Linkage and Current Research Problems," *Statistical Research Division, US Census Bureau*, 1999.

[24] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, "Duplicate Record Detection: A Survey," *IEEE Trans. on Knowl. and Data Eng.*, vol. 19, no. 1, pp. 1–16, 2007.

[25] M. Marelli, L. Bentivogli, M. Baroni, R. Bernardi, S. Menini, and R. Zamparelli, "Semeval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. Dublin, Ireland: ACL, 2014, pp. 1–8.

[26] B. T. McInnes, T. Pedersen, Y. Liu, G. B. Melton, and S. V. Pakhomov, "U-path: An undirected path-based measure of semantic similarity," in *AMIA Annual Symposium Proceedings*, vol. 2014. American Medical Informatics Association, 2014, p. 882.

[27] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pasca, and A. Soroa, "A study on similarity and relatedness using distributional and WordNet-based approaches," in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, ser. NAACL'09. Stroudsburg, PA, USA: ACL, 2009, pp. 19–27.

[28] Mikolov, Tomas, Sutskever, Ilya, Chen, Kai, Corrado, Greg, and Dean, Jeff, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems (NIPS)*, 2013, pp. 3111–3119.

[29] Pennington, Jeffrey, Socher, Richard, and Manning, Christopher D., "Glove: Global vectors for word representation," in *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)*, vol. 12, Doha, Qatar, 2014, pp. 1532–1543.

[30] E. Gabrilovich and S. Markovitch, "Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, ser. IJCAI'07. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007, pp. 1606–1611.

[31] S. Banerjee and T. Pedersen, "An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet," in *Computational Linguistics and Intelligent Text Processing*, ser. Lecture Notes in Computer Science, A. Gelbukh, Ed. Springer, 2002, no. 2276, pp. 136–145.

[32] C. Corley and R. Mihalcea, "Measuring the semantic similarity of texts," in *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, ser. EMSEE'05. Stroudsburg, PA, USA: Association for Computational Linguistics, 2005, pp. 13–18.

[33] D. Croce, V. Storch, P. Annesi, and R. Basili, "Distributional Compositional Semantics and Text Similarity," in *Proceedings of the IEEE Sixth International Conference on Semantic Computing (ICSC)*, Sep. 2012, pp. 242–249.

[34] D. Croce, V. Storch, and R. Basili, "UNITOR-CORE TYPED: Combining Text Similarity and Semantic Filters through SV Regression," in *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: SemanticTextual Similarity*. Atlanta, Georgia, USA: ACL, 2013, pp. 59–65.

[35] M.-C. De Marneffe, B. MacCartney, C. D. Manning, and others, "Generating typed dependency parses from phrase structure parses," in *Proceedings of LREC*, vol. 6, 2006, pp. 449–454.

[36] M. D. Lee, B. Pincombe, and M. Welsh, "An empirical evaluation of models of text document similarity," in *In CogSci2005*. Erlbaum, 2005, pp. 1254–1259.

[37] E. Agirre, D. Cer, M. Diab, and G.-A. Aitor, "SemEval-2012 Task 6: A Pilot on Semantic Textual Similarity," in *First Joint Conference on Lexical and Computational Semantics (*SEM)*. Montreal,Canada: ACL, 2012, pp. 385–393.

[38] E. Agirre, C. Banea, C. Cardie, D. Cer, M. Diab, A. Gonzalez-Agirre, W. Guo, R. Mihalcea, G. Rigau, and J. Wiebe, "SemEval-2014 Task 10: Multilingual semantic textual similarity," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. Dublin, Ireland: ACL, 2014, pp. 81–91.

[39] A. Lynum, P. Pakray, B. Gambäck, and S. Jimenez, "NTNU: Measuring Semantic Similarity with Sublexical Feature Representations and Soft Cardinality," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. Dublin, Ireland: ACL, 2014, pp. 448–453.

[40] S. Jimenez, G. Duenas, J. Baquero, and A. Gelbukh, "UNAL-NLP: Combining soft cardinality features for semantic textual similarity, relatedness and entailment," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. Dublin, Ireland: ACL, 2014, pp. 732–742.

[41] D. Jurgens, M. T. Pilehvar, and R. Navigli, "SemEval-2014 Task 3: Cross-level semantic similarity," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. Dublin, Ireland: ACL, 2014, pp. 17–26.

[42] M. Negri, A. Marchetti, Y. Mehdad, L. Bentivogli, and D. Giampiccolo, "2012. Semeval-2012 Task 8: Cross-lingual Textual Entailment for Content Synchronization," in *First Joint Conference on Lexical and Computational Semantics (*SEM)*. Montreal, Canada: ACL, 2012, pp. 399–407.

[43] M. Negri, A. Marchetti, Y. Mehdad, and L. Bentivogli, "Semeval-2013 Task 8: Cross-lingual Textual Entailment for Content Synchronization," in *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013)*. Atlanta, Georgia, USA: ACL, 2013, pp. 25–33.

[44] S. Jimenez, C. Becerra, and A. Gelbukh, "SOFTCARDINALITY: Hierarchical Text Overlap for Student Response Analysis," in *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Atlanta, Georgia, USA: ACL, 2013, pp. 280–284.

[45] ——, "SOFTCARDINALITY: Learning to Identify Directional Cross-Lingual Entailment from Cardinalities and SMT," in *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Atlanta, Georgia, USA: ACL, Jun. 2013, pp. 34–38.

[46] ——, "Soft Cardinality: A Parameterized Similarity Function for Text Comparison," in *First Joint Conference on Lexical and Computational Semantics (*SEM)*. Montreal, Canada: ACL, 2012, pp. 449–453.

[47] M. O. Dzikovska, R. D. Nielsen, C. Brew, C. Leacock, D. Giampiccolo, L. Bentivogli, P. Clark, I. Dagan, and H. T. Dang, "SemEval-2013 Task 7: The Joint Student Response Analysis and 8th Recognizing Textual Entailment Challenge," in *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013), in conjunction with the Second Joint Conference on Lexical and Computational Semantcis (*SEM 2013)*. Atlanta, Georgia, USA: ACL, 2013, pp. 263–274.

[48] S. P. Leeman-Munk, E. N. Wiebe, and J. C. Lester, "Assessing elementary students' science competency with text analytics," in *Proceedins of the Fourth International Conference on Learning Analytics And Knowledge (LAK 14)*. Indianapolis, Indiana, USA: ACM, 2014, pp. 143–147.