Appl. Comput. Math., V.11, N.2, 2012, pp.180-199

BASELINES FOR NATURAL LANGUAGE PROCESSING TASKS BASED ON SOFT CARDINALITY SPECTRA

SERGIO JIMENEZ[1], ALEXANDER GELBUKH[2]

Abstract. Soft-cardinality spectra (SC spectra) is a new method of approximation for text strings in linear time, which divides text strings into character q-grams of different sizes. The method allows simultaneous use of weighting at term and q-gram levels. SC spectra in combination with resemblance coefficients allows the construction of a family of text similarity functions that only use the surface information of the texts and weights obtained in the same text collection. These similarity measures can be used in various tasks of natural language processing as baseline for other methods that exploit the hidden syntactic and/or semantic structure using resources based on knowledge, inference of large corpora. The proposed method was evaluated on 22 data sets to address the tasks of information retrieval, entity matching, paraphrase and textual entailment recognition. The results raised the bar near to the best published results in the used data sets. We claim that any method that uses any resource or information external to a particular data set should outperform our method. We found that our method is an effective and challenging baseline for the evaluated tasks.

Keywords: text similarity, soft cardinality, SC spectra, q-grams, NLP baselines, entity resolution, information retrieval, paraphrase recognition, textual entailment recognition

AMS Subject Classification: 68T50, 68P20

1. Introduction

The assessment of similarity is the ability to balance commonalities and differences between two objects to produce a similarity judgment. People and most animals have this intrinsic capacity, which makes this an important requirement for artificial intelligence systems. Although, the computational exact comparison of any two object representations is trivial, approximate comparison has to deal with issues such as noise, nuance and ambiguity. Therefore, the agreement of computer-generated and human similarity judgments is a challenge for artificial intelligence systems.

In natural language processing, text similarity functions are basic components in many particular tasks [26] namely, textual entailment, question answering, summarization, paraphrasing, semantic text similarity assessment, entity resolution, information retrieval, text classification, text clustering, etc. For instance, the entity resolution task consists of finding co-referential names in a couple of lists of names, dealing with misspellings, homonyms, initialisms, aliases, typos, and other issues. A text similarity function can be used to obtain a ranking of the most similar pairs as candidates to be the same entity. The results above a threshold are evaluated against a gold standard built with human judgments.

Similarly, many information retrieval approaches aim to reproduce the human relevance judgments building similarity functions that compare queries and documents using similarity scores as evidence of relevance. In paraphrase and textual entailment recognition, text similarity

[1]Universidad Nacional de Colombia, Ciudad Universitaria, ed. 453, of. 220, Bogota, Colombia, e-mail: sgjimenez@unal.edu.co.

[2]Centro de Investigación en Computación (CIC), Instituto Politécnico Nacional (IPN), Av. Juan Dios Bátiz s/n, casi esq. Av. Mendizábal, Col. Nueva Industrial Vallejo, CP 07738, DF, Mexico, e-mail: gelbukh@gelbukh.com

Manuscript received 07/06/2012.

functions have been used to compare pairs of texts to determine whether a particular pair is semantically similar enough to be a valid paraphrase or entailment pair. In this paper, we explore the usefulness of a new text similarity function in the following tasks: entity resolution (or name matching), information retrieval, paraphrase and textual entailment recognition.

The current text comparison methods can be classified by the level of granularity in which the texts are divided for comparison. For example, characters are used as comparison unit in the family of methods derived from the edit distance [28]. The granularity gradually decreases in the methods based on q-grams of characters [25]. Q-grams, also known as kmers or n-grams, are consecutive substrings of length q that overlap $q-1$ characters. Even coarser, methods such as the vector space model (VSM) [38] and the resemblance coefficients applied to text [39] make use of the terms (i.e., words or symbols) as subdivision unit. The methods that have achieved the best performance on the task of entity resolution are those that combine term-level comparisons with comparisons of character or q-gram level. Some examples of these hybrid approaches are the Monge-Elkan measure [32], SoftTFIDF [8], fuzzy match similarity [5], meta-Levenshtein [33] and the soft cardinality (SC) [22]. The method proposed in this paper combines information from several text subdivisions ranging from characters, q-grams to terms levels in a new approach based on the idea of soft cardinality.

Text similarity functions can also be classified by the information used to calculate their similarity scores. The simplest approach is to use *static similarity functions*, which only use the information contained in the surface of the pair of texts that are being compared, e.g. *edit_distance(A,B)* [28]. The *adaptive similarity functions* are the next category, because in addition to the information used by the static ones they use the entire collection of text being compared. For instance, the function *cosine_tf_idf(A,B,collection)* [38] requires the entire text collection (as third parameter) to obtain the *tf-idf* weights of the terms in texts A and B. In general, adaptive similarity functions recombine information of the text collection to produce the similarity score for any pair of texts. The third category is the *semantic similarity functions*, which make use of any additional resources based on large corpora, knowledge or combinations of them, e.g. POS-taggers, parsers, dictionaries, thesaurus, semantic networks (e.g. WordNet, ontologies), structured corpus (e.g. Wikipedia), annotated corpus (e.g. BNC[1], NY Times corpus), unlabeled corpus, parallel corpus (e.g. English-French Canadian Parliament hansards), the Web, etc. For instance, the weighted bilingual dictionary proposed in [40] can be used to leverage a similarity function that compares texts in two different languages with statistics gathered in large corpora. Other approaches that reveal from scratch the latent semantic structure of a text collection (e.g. LSA [16]) are a specialization of the adaptive similarity functions. However, these approaches are not being considered in this paper.

Given the amount of information and knowledge used by each one of the previous categories it is expected that each stage could be used as a baseline for the next. For example, if a static function outperforms an adaptive function in a particular task, it makes no sense to use the latter. Similarly, the semantic similarity functions should outperform static and adaptive ones to justify the use of their resources, which are generally complex in time and storage space. The method proposed in this paper can be whether static or adaptive depending on the used weighting scheme. We used a static version for the entity resolution (ER) and information retrieval (IR) tasks, and we compared its results against other adaptive approaches. The proposed static approach reached performances close to that of the adaptive approaches, and in some cases better results. For paraphrase and textual entailment recognition we used an adaptive version and its results were compared against other semantic methods. In this scenario, our method achieved performances close to the best results already published, and in some cases better than many semantic approaches.

The proposed method is based on the soft cardinality [22]. The soft cardinality is a set-based method for comparing objects that softens the rigid count of elements that makes the

[1]British National Corpus

classical set cardinality by considering the similarities among elements. The definition of the soft cardinality requires the calculation of 2^m intersections for a set with m elements. An approach to approximate the soft cardinality using only m^2 computations of an auxiliary similarity measure that compares pairs of elements is proposed in [22]. The soft cardinality can be used to compare texts considering texts as sets of terms.

In this paper, we propose a new method for approximating the soft cardinality that, unlike the current approach, does not require any auxiliary similarity measure. Furthermore, the new method allows the simultaneous comparison of unigrams (i.e., characters), bigrams or trigrams by combining any range of them. We call these combinations soft-cardinality spectra (or SC spectra for shorter). SC spectra can be computed in linear time allowing the use of soft cardinality with relatively large texts that are used in applications such as information retrieval.

We tested SC-spectra with 11 entity resolution data sets [8], 9 classical information retrieval collections [2], the MSR paraphrase corpus [15] and the RTE-3 textual entailment data set [18]. The proposed approach overcame all baselines and provided quite good results compared to other baselines and methods.

The remainder of this paper is organized as follows: Section 2 describes the cosine TF-IDF and softTFIDF measures. Section 3 briefly summarizes the soft cardinality method for text comparison. The proposed soft-cardinality spectra (SC spectra) method is presented in Section 4. Section 5 describes how to build similarity functions using the proposed cardinality. In Section 6, the proposed method is experimentally evaluated and a brief discussion is provided for each tested task. Related work is presented in Section 7. Finally, conclusions are drawn in Section 8.

2. Cosine TF-IDF and softTFIDF

The cosine TF-IDF measure [38] was proposed almost three decades ago and today is still considered an effective method for text comparison. For using this measure a pair of texts A and B are represented as vectors in a space indexed by the vocabulary of $A \cap B$. The values on each dimension of the vectors are the weights that determine the relative importance of the terms. These weights are obtained by combining evidence from the text (i.e. *tf* term frequency) and from the entire collection (i.e. *idf* inverse document frequency). *Tf-idf* weights are calculated using the following expressions:

$$idf\,(a_i) = \log\left(\frac{M}{m_{a_i}}\right) \tag{2.1}$$

$$tfidf(D, a_i) = tf(D, a_i) \times idf(a_i), \tag{2.2}$$

where M is the number of texts in the data set, m_{a_i} is the number of texts where the term a_i occurs and $tf(D, a_i)$ is the count of the term a_i in the document D. The similarity score between two texts is obtained by the cosine of the angle between both vectors:

$$TFIDF(A, B) = \sum_{t \in A \cap B} \left(\frac{tfidf\,(A, t)}{K(A)} \times \frac{tfidf\,(B, t)}{K(B)}\right), \tag{2.3}$$

where $K(A)$ is a normalization factor $K(A) = \sqrt{\sum_{t' \in A} tfidf\,(A, t')^2}$. This notation allows a better comparison with the softTFIDF measure [8]. SoftTFIDF addresses the problem of term interdependence by extending the set of terms that contributes to the commonalities from the terms in $A \cap B$ to those pairs of terms that surpassed a threshold θ of similarity provided by an auxiliary similarity function $sim\,(t_a, t_b)$:

$$softTFIDF(A,B,\theta) = \sum_{\substack{t_a \in A \\ t_b \in B \\ sim(t_a,t_b) > \theta}} \left(\frac{tfidf\,(A,t_a)}{K(A)} \times \frac{tfidf\,(B,t)}{K(B)} \times sim\,(t_a,t_b) \right) \tag{2.4}$$

The auxiliary similarity function $sim\,(t_a,t_b)$ can be any measure fulfilling the postulate of identity (i.e. $sim\,(t,t) = 1$) and alternatively other metric-space postulates such as symmetry, triangle inequality, and positiveness. In [8] the Jaro-Winkler measure [44] was used as auxiliary similarity function with a threshold $\theta = 0.9$.

3. Soft cardinality for text comparison

The classical set cardinality is a function of a set that counts the number of different elements in that set. When a text is represented as a bag of words, the cardinality of the bag is the size of the vocabulary of terms, i.e. the number of different terms used. The cardinality can be used with resemblance coefficients to provide similarity measures that compare pairs of sets. Examples of these measures are *Jaccard* ($|A \cap B|/|A \cup B|$), *Dice* ($2|A \cap B|/(|A| + |B|)$) and *cosine* ($|A \cap B|/\sqrt{|A||B|}$) coefficients. The effect of the cardinality function in these measures is to count the number of common elements and to compress the repeated elements in a single instance. Based on an information-theoretical definition of similarity proposed in [29], a compression distance [7], which explicitly takes advantage of this feature, was shown to be useful in text applications.

However, the compression provided by classical set cardinality is rigid. That is, while only identical elements in a set are counted once, two nearly identical elements are counted twice. The soft cardinality addresses this issue taking into account the similarities between the elements of the set. The intuition of the soft cardinality is as follows: the elements that have similarities with other elements contribute less to the total cardinality than unique elements. Therefore, the soft cardinality takes into account not only the elements that are identical but also the elements that are similar.

3.1. **Definition of soft cardinality.** The soft cardinality of a set is the cardinality of the union of its elements treated (themselves) as sets. Thus, for a set $A = \{a_1, a_2, \ldots, a_{|A|}\}$, the soft cardinality of A is:

$$|A|' = \left| \bigcup_{i=1}^{|A|} a_i \right| \tag{3.1}$$

This set-based definition allows to provide the following expressions:

$$|A \cup B|' = \left| \left(\bigcup_{i=1}^{|A|} a_i \right) \cup \left(\bigcup_{i=1}^{|B|} b_i \right) \right|$$

$$|A \cap B|' = |A|' + |B|' - |A \cup B|'$$

Given that the elements a_i of the set A are also sets, they have their own "sub"-elements (i.e. the elements of the elements) and cardinalities. These cardinalities are the number of different "sub"-elements in any a_i element, making trivial the computation of $|A|'$. However, the cardinality of each element $|a_i|$ can also be associated with the relative weight (or importance) of a_i in A. Besides, the cardinality of the intersections among the elements a_i of the set A can also be provided by information sources different from the common "sub"-elements of the elements a_i.

Let us consider the example depicted in Figure 3.1 in which two Spanish proper names are represented as sets A and B. The term elements a_i, b_j, and the intersections are represented as Venn diagrams. In this example, all elements are equally weighted and the represented intersections could be derivated from any information source such an edit-distance-based similarity

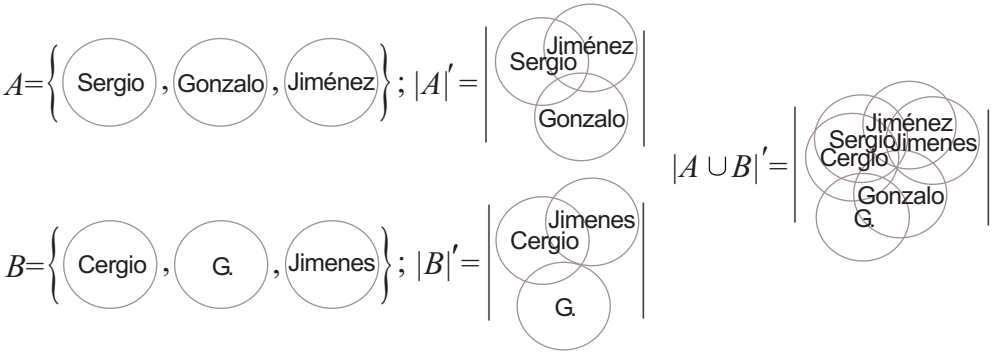

FIGURE 3.1. Example of soft cardinality with equally weighted terms.

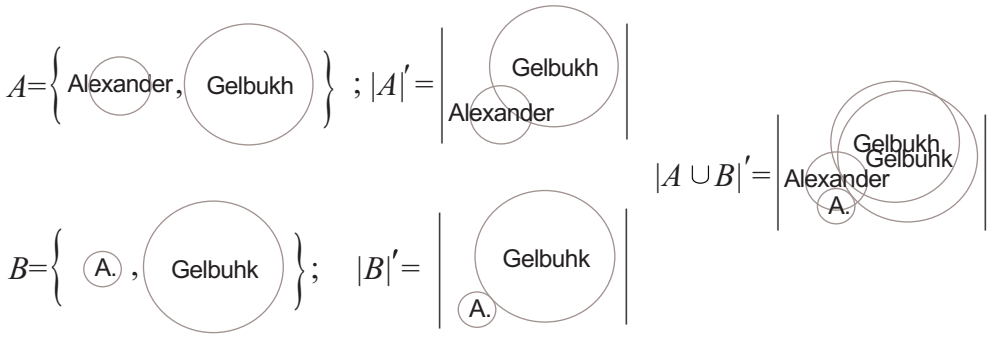

FIGURE 3.2. Example of soft cardinality using weighted terms.

function [28]. The Figure 3.2 shows a similar scenario but with differently weighted terms depicted as circles with different radii. These weights could be derived from *tf-idf* weights [38] that reflect the fact that "Alexander" is a commoner term than "Gelbukh" in a name database. In this graphic methaphor, the soft cardinality of sets A, B and $A \cap B$ are represented as the interior of the resulting cloud shaped border.

Obviously, in the scenarios described in the previous examples the expression 3.1 cannot be computed directly. That is, when the cardinalities $|a_i|$ and the cardinalities of the interseccions among elements a_i are provided by an external information source. Alternatively, it is possible to use properties of the classical set cardinality to compute 3.1 using only the cardinalities provided by the external information source. For instance, in Figure 3.1 a_1 = "Sergio", a_2 = "Gonzalo" and a_3 = "Vargas", thus the soft cardinality of A is $|A|^{'} = |a_1 \cup a_2 \cup a_3|$,

$$|A|^{'} = |a_1| + |a_2| + |a_3| - |a_1 \cap a_2| - |a_2 \cap a_3| - |a_1 \cap a_3| + |a_1 \cap a_2 \cap a_3| \tag{3.2}$$

It is important to note that, the cardinalities of the right side in 3.2 can be provided by term weighting approaches and auxiliarly similarity functions.

3.2. **SC approximation with similarity functions.** The approach proposed in the previous example using 3.2 is not practical because the number of terms in 3.2 increases exponentially with the number of elements in A. Alternatively, 3.1 can be approximated by using only pair wise intersections of the elements of A using the following expression proposed in [22]:

$$|A|^{'}_{\alpha} \simeq \sum_{i}^{n} \left(w_{a_i} \times \left(\sum_{j}^{n} \alpha(a_i, a_j)^p \right)^{-1} \right), \tag{3.3}$$

where $w_{a_i} = |a_i|$ and the function α is an auxiliary similarity function that can compare any pair of elements in A. This function α must return scores in the range $[0, 1]$ satisfying at least identity $\forall x : \alpha(x, x) = 1$ and symmetry $\forall x, y : \alpha(x, y) = \alpha(y, x)$ postulates. In fact, when α is a rigid comparator (i.e., returns 1 when the elements are identical and 0 otherwise) and weights w_{a_i} are equal to 1, $|A|'_\alpha$ becomes the classical cardinality $|A|$. Finally, the exponent p is a tuning parameter investigated in [22], obtaining good results using $p = 2$ in an entity-resolution task. The parameter p controls the "softness" of the cardinality, so that when $p \to \infty$, then $|A|'_\alpha \to \sum_{i=1}^{|A|} w_{a_i}$. Similarly, when $0 \leftarrow p$, then $|A|'_\alpha \to \frac{1}{|A|} \sum_{i=1}^{|A|} w_{a_i}$.

Note that the computational order of the approximation proposed in 3.3 is quadratic $O(n^2)$. Although, the complexity of this approach is far better than the exponential approach used for the example in 3.2, the usage of 3.3 is constrained to relatively short texts.

4. Computing soft cardinality using sub-strings

The soft cardinality approximation shown in 3.3 is quite general since the function of similarity between pairs of terms α can be any measure that may or may not use the surface representation of both strings. For example, the edit distance [28] is based on a surface representation of characters, in contrast to the semantic relatedness functions [34] that may be based on contexts in a large corpus or on a semantic network. However, the proposed approach is entirely static since the idea is to approximate the soft cardinality of a text represented as a set of terms by subdividing the terms into q-grams of characters.

Several comparative studies have shown the benefits of hybrid approaches that first tokenize (split into terms) a text string and then make comparisons between the terms at character or q-gram level [8, 4, 6, 35, 22]. Similarly, the soft cardinality approximation in 3.3 is based on an initial tokenization and an implicit further subdivision made by the function α for assessing similarities and differences between pairs of terms. The intuition of the new proposed soft cardinality approximation is to conduct an initial tokenization of the text, then to divide each term into q-grams, to make a list of all the different q-grams, and finally, calculate a weighted sum of the sub-strings with weights that depends on the number of substrings in each term. Besides, the proposed method also considers importance weights for each term and q-gram occurrence.

4.1. Soft cardinality based on q-grams.

Q-grams are consecutive overlapped subsequences of length q in a string [41]. Q-grams provides the ability to maintain a partial order in text representations based on unordered structures such as bags or sets. Although text can be represented as sets of q-grams at term or character levels, in this paper we only considered q-grams at character level.

The q-gram character subdivision of a word can be enriched with padding characters [24]. These padding characters are especial characters added at the beginning and end of each term before being divided into q-grams. These characters distinguish the heading and trailing q-grams from those that are in the middle of the term. The number of padding characters added can be 1 (*single padding*) or $q-1$ (*full padding*). For instance, the term "sunday" divided into trigrams using *not padding*, *single padding* and *full padding* are respectively {*sun*, *und*, *nda*, *day*}, {◁*su*, *sun*, *und*, *nda*, *day*, *ay*▷} and {◁◁*s*, ◁*su*, *sun*, *und*, *nda*, *day*, *ay*▷, *y*▷▷}. It is also possible to consider the smallest 1-grams (or unigrams) subdivision in which no padding characters are allowed, i.e. {*s*, *u*, *n*, *d*, *a*, *y*}. When q is greater than the length in characters of the term the q-gram subdivision is the term itself. For example, the 4-grams (or quadgrams) subdivision of the term "sun" is {*sun*}.

The soft cardinality of a text represented as a set of terms can be approximated representing each term as a set of q-grams in order to apply the definition 3.1. Consider the following example with the Spanish name "Gonzalo Gonzalez", A ={"Gonzalo","Gonzalez"}, a_1 ="Gonzalo" and a_2 ="Gonzalez". Using bigrams with padding characters as subdivision unit, the pair of terms can be represented as: $a_1^{[2]}$ ={◁*G*, *Go*, *on*, *nz*, *za*, *al*, *lo*, *o*▷} and $a_2^{[2]}$ ={◁*G*, *Go*, *on*, *nz*, *za*, *al*, *le*,

ez, *z⊳*}. The exponent in square brackets means the size q of the q-gram subdivision. Let $A^{[2]}$ be the set with all different bigrams $A^{[2]} = a_1^{[2]} \cup a_2^{[2]} =$ {◁*G*, *Go*, *on*, *nz*, *za*, *al*, *lo*, *o⊳*, *le*, *ez*, *z⊳*}, and its classical cardinality is $\left|A^{[2]}\right| = \left|a_1^{[2]} \cup a_2^{[2]}\right| = 11$. Similarly, $\left|a_1^{[2]} \setminus a_2^{[2]}\right| = 2$, $\left|a_2^{[2]} \setminus a_1^{[2]}\right| = 3$ and $\left|a_1^{[2]} \cap a_2^{[2]}\right| = 6$.

Similar to $|A|$, which provides an integer count of the number of terms in A, $|A|'$ provides a real number that represent the "soft" count of the number of terms in A. In our running example $|A|'$ must be a number in between 1.0 and 2.0. Thus, each one of the elements of $A^{[2]}$ adds a contribution to the total soft cardinality of A. The cardinality of $A^{[2]}$ represents the number of bigrams in A, in order to make it represent the "soft" number of terms in A the cardinalities of the bigrams in $A^{[2]}$ need to be adjusted. Now, let us denote $A_i^{[2]}$ as a bigram such that $A_j^{[2]} \in A^{[2]}$. The cardinalities of the non-common bigrams between $a_1^{[2]}$ and $a_2^{[2]}$ are adjusted by $\left|A_j^{[2]}\right| = \frac{1}{\left|a_1^{[2]}\right|}; \forall A_j^{[2]} \in \left(a_1^{[2]} \setminus a_2^{[2]}\right)$ and $\left|A_j^{[2]}\right| = \frac{1}{\left|a_2^{[2]}\right|}; \forall A_j^{[2]} \in \left(a_2^{[2]} \setminus a_1^{[2]}\right)$, that is the contribution of each bigram is inverse to the number of bigrams in the term. Similarly, the cardinality of the common bigrams is adjusted by the average $\left|A_j^{[2]}\right| = 0.5 \times \frac{1}{\left|a_1^{[2]}\right|} + 0.5 \times \frac{1}{\left|a_2^{[2]}\right|}; \forall A_j^{[2]} \in \left(a_1^{[2]} \cup a_2^{[2]}\right)$. In our example, $\frac{1}{\left|a_1^{[2]}\right|} = 0.125$, $\frac{1}{\left|a_2^{[2]}\right|} = 0.11\bar{1}$ and $0.5 \times \frac{1}{\left|a_1^{[2]}\right|} + 0.5 \times \frac{1}{\left|a_1^{[2]}\right|} = 0.118$. Finally, given that there are 6 common bigrams between $a_1^{[2]}$ and $a_2^{[2]}$, 2 bigrams exclusively in $a_1^{[2]}$ and 3 bigrams exclusively in $a_2^{[2]}$, the final soft cardinality for this example is $|A|' \simeq 0.118 \times 6 + 0.125 \times 2 + 0.11\bar{1} \times 3 = 1.292$. The soft cardinality of A reflects the fact that a_1 and a_2 are very similar in contrast to the classical cardinality that obtains $|A| = 2$

4.2. **Soft cardinality *q*-spectrum.** In the previous example we obtained an approximation of the soft cardinality using a partition of bigrams. The soft cardinality of any text string represented as a set of terms A can be approximated by the partition $A^{[q]} = \bigcup_{i=1}^{|A|} a_i^{[q]}$ of A in q-grams, where $a_i^{[q]}$ is the partition of i-th term in A into q-grams. Clearly, each one of the q-grams $A_j^{[q]}$ in $A^{[q]}$ can occur in various terms a_i in A. The indices i satisfying $A_j^{[q]} \in a_i^{[q]}$ indexes all the terms a_i in A where the q-gram $A_j^{[q]}$ occurs. The number of terms in A in which the q-gram $A_j^{[q]}$ occurs is denoted by K_{A_j}. The contribution of a particular q-gram $A_j^{[q]}$ to the total soft cardinality is the arithmetic average of the weights $\frac{1}{\left|a_i^{[q]}\right|}$ for each one of its occurrences in the text. The expression for the q-spectrum soft cardinality is:

$$|A|'_{[q]} \simeq \sum_{j=1}^{|A^{[q]}|} \left(\frac{1}{K_{A_j}} \times \sum_{i:A_j^{[q]} \in a_i^{[q]}} \left(\frac{1}{\left|a_i^{[q]}\right|} \right) \right) \tag{4.1}$$

The approximation $|A|'_{[q]}$ obtained from the 4.1 using q-grams is the soft cardinality (SC) q-spectrum of A. Note that this cardinality expression depends only on the information in the set A, so any similarity measure derived from 4.1 is a static measure.

The soft cardinality of a text provides a "soft" count of the number or terms in the text equally weighting all terms. However, it is already known that in a particular text some of its terms convey more information than others (see Section 2). The term weights obtained with 2.2 can also be integrated to the SC q-spectrum expression as follows:

$$|A|'_{[q]} \simeq \sum_{j=1}^{|A^{[q]}|} \left(\frac{1}{K_{A_j}} \times \sum_{i:A_j^{[q]} \in a_i^{[q]}} \left(\frac{tfidf(A, a_i)}{\left|a_i^{[q]}\right|} \right) \right) \tag{4.2}$$

Note that *tf-idf* weights used in 4.2 are obtained from statistics gathered from the entire collection of texts being compared. Thus, any similarity measure derived of the use of 4.2 is an adaptive measure.

Furthermore, the idea of weighting can also be also applied at q-gram level to discriminate substrings according with the amount of information that each q-gram conveys. This amount of information can be associated with the frequency of the q-gram in a large corpus or in a particular text collection. For instance, in the English language the character bigram "th" is considerably more frequent than the bigram "xy". Thus, the former conveys less information than the latter. We adopt a weighting scheme similar to that used in 2.1 for *idf* to weight q-grams based on the frequency of occurrence. Thus, the weight for a particular q-gram $A_j^{[q]}$ is:

$$qidf(A_j^{[q]}) = \log \frac{N}{n_{A_j^{[q]}}}, \tag{4.3}$$

where N is the total number of terms (words) in the collection of texts and $n_{A_j^{[q]}}$ is the number of terms in which the q-gram $A_j^{[q]}$ occurs. Considering that the number of q-grams per term may be significatively less than the number of terms per document, the effect of repeated q-grams in a term is low, so we do not consider *tf*-like weights at q-gram level. As *idf*, *qidf* weights can also be integrated to the SC q-spectrum expression as follows:

$$|A|'_{[q]} \simeq \sum_{j=1}^{|A^{[q]}|} \left(\frac{1}{K_{A_j}} \times \sum_{i:A_j^{[q]} \in a_i^{[q]}} \left(\frac{tfidf(A, a_i) \times qidf(A_j^{[q]})}{\left|a_i^{[q]}\right|} \right) \right) \tag{4.4}$$

The similarity measures derived from the use of 4.4 are adaptive measures, but their adaptiveness is made simultaneously at term and q-gram levels.

The inner expression $\frac{tfidf(A,a_i)\times qidf(A_j^{[q]})}{\left|a_i^{[q]}\right|}$ in 4.4 is the weight associated with each q-gram in each term in the text A. This weighting expression depends on the weight of the term given the text collection (i.e. term weights $tfidf(A, a_i)$), the weight of the q-gram given the term collection (i.e. q-gram weights $qidf(A_j^{[q]})$) and on the number of q-grams in the term (i.e. context q-gram weights $1/\left|a_i^{[q]}\right|$). The expression in 4.4 first averages the q-gram weights for repeated occurrences of all q-grams in the text and next it makes the sum of those averages. Clearly, the term and q-gram weighting functions $tfidf()$ and $qidf()$ can be replaced by any weighting mechanism.

In the experimental evaluation provided in Section 6 different combinations of term and q-gram weighting mechanisms will be tested for different natural language processing tasks.

4.3. **Soft-cardinality spectra.** A partition of q-grams allows the construction of similarity measures with its associated soft cardinality q-spectrum. The most fine-grained substring partition corresponds to $q = 1$ (i.e. characters or unigrams) and the coarsest is the partition into terms. While partitions such as unigrams, bigrams and trigrams are used in tasks such as entity resolution, the partition into terms is preferred for information retrieval, text classification and other tasks. Intuitively, the finer partitions seem to be suitable for short texts and term partitions seem to be more suitable for long texts.

However, as it was shown in [43, 23] the more convenient partitions for text comparison ranged from trigrams to heptagrams depending on the used similarity function and on the data set. It

was also shown that in general the performance of the similarity function decreases considerably when $q < 3$ or $q > 7$. Although their experimental results suggested that there is a single value of q for maximum performance for any pair $\{$*similarity function*, *data set*$\}$, there is not a method to unsupervisedly determine such optimal value of q.

The proposed method aims to combine a series of partitions with different q in order to obtain an aggregate score close to the optimum selection of q. For instance, in Figure 6.1 b) the performance of a particular similarity measure in the IR collection ADI is shown for different q-gram partition sizes. It is difficult to unsupervisedly determine that pentagrams (i.e. $q = 5$) is the best partition size to use. Nevertheless, it is possible to guess a range such as $q_1 = 2$, $q_2 = 8$ just by considering the features of the documents and the similarity to other reference collections. Moreover, the aggregation of different text representations derived from various q-gram sizes can provide a better final performance in the task taking into account the fundamental reasons why ensemble models may work better than the single ones, that is statistical, computational and representational reasons (see [13]).

As we mentioned, the combination of several contiguous partition granularities can be useful for comparing texts in a particular dataset. Since each SC q-spectrum provides a measure of the compressed amount of terms in a text, several SC q-spectrum can be averaged or added to get a combined measure. SC $[q_s : q_e]$-spectra is defined as the aggregation of a series of several q-spectrum from q_s to q_e, having $q_s \leq q_e$. For example, the SC $[2 : 4]$-spectra uses simultaneously bigrams, trigrams and quadgrams to get an approximation the soft cardinality of a bag of words. Thus, the SC spectra expression is:

$$|A|'_{[q_s:q_e]} = \sum_{i=s}^{e} |A|'_{[q_i]}. \tag{4.5}$$

5. Building text similarity functions

Once the SC spectra function is provided, text similarity functions can be constructed replacing the classical set cardinality by SC spectra in resemblance coefficients. Resemblance coefficients are binary similarity measures that compare two sets A and B by the ratio between the cardinality of the commonalities (i.e. $|A \cap B|$) and the aggregation of the cardinalities of the two sets, e.g. Jaccard [20] and Dice [12] coefficients. The aggregation of $|A|$ and $|B|$ can be made using the generalized mean, which control the aggregation by a parameter p. We call this resemblance coefficient as the *generalized mean coefficient*:

$$SIM(A, B) = \frac{|A \cap B|}{(0.5 \times |A|^p + 0.5 \times |B|^p)^{\frac{1}{p}}} \tag{5.1}$$

This coefficient can also be considered as a derivation for similarity of the Minkowski distance $D(x, y) = (|x_1 - y_1|^p + |x_2 - y_2|^p)^{\frac{1}{p}}$; note that vertical lines in this expression denotes absolute values rather than cardinalities. The generalized mean coefficient is similar to the coefficient proposed in [21] because it models the asymmetric selection of the referent for comparison. That is, when $-\infty \leftarrow p$ and $p \rightarrow \infty$ the denominator in 1 becomes $\min(|A|, |B|)$ and $\max(|A|, |B|)$ respectively. Different values of p in 5.1 produce a family of resemblance coefficients some values of p correspond to some already known coefficients (see Table 1).

Several text similarity functions can be proposed using the generalized mean coefficient and the SC spectra function proposed in Section 4. For instance, using $p = -1$ in 5.1 (i.e. harmonic coefficient) and SC spectra the following expression can be obtained:

$$sim(A, B) = 1 + \frac{1}{2} \times \left(\frac{|A|'_{[q_s:q_e]}}{|B|'_{[q_s:q_e]}} + \frac{|B|'_{[q_s:q_e]}}{|A|'_{[q_s:q_e]}} - \frac{|A \cup B|'_{[q_s:q_e]}}{|A|'_{[q_s:q_e]}} - \frac{|A \cup B|'_{[q_s:q_e]}}{|B|'_{[q_s:q_e]}} \right). \tag{5.2}$$

In the following evaluation section several text similarity functions obtained using different p values were tested.

Table 1. Different instances of the generalized mean coefficient.

p	Name	Expression
1	Dice coefficient	$\frac{2\times\lvert A\cap B\rvert}{\lvert A\rvert+\lvert B\rvert}$
$p\to 0$	cosine coefficient	$\frac{\lvert A\cap B\rvert}{\sqrt{\lvert A\rvert\times\lvert B\rvert}}$
2	quadratic coefficient	$\frac{\lvert A\cap B\rvert}{\sqrt{0.5\times\left(\lvert A\rvert^2+\lvert B\rvert^2\right)}}$
$-\infty\leftarrow p$	overlap coefficient	$\frac{\lvert A\cap B\rvert}{\min(\lvert A\rvert,\lvert B\rvert)}$
-1	harmonic coefficient	$\frac{\lvert A\cap B\rvert\times(\lvert A\rvert+\lvert B\rvert)}{2\times\lvert A\rvert\times\lvert B\rvert}$

Table 2. Naming convention for the weighting schemes used in experiments.

Convention name	Expression	Type	Level
none	1	static	n/a
c	$\frac{1}{\lvert a_i^{[q]}\rvert}$	static	character q-grams
idf	$idf(a_i)$	adaptive	terms
qidf	$qidf(A_j^{[q]})$	adaptive	character q-grams
c.idf	$\frac{idf(a_i)}{\lvert a_i^{[q]}\rvert}$	adaptive	character q-grams & terms
c.qidf	$\frac{qidf(A_j^{[q]})}{\lvert a_i^{[q]}\rvert}$	adaptive	character q-grams
idf.qidf	$idf(a_i)\times qidf(A_j^{[q]})$	adaptive	character q-grams & terms
c.idf.qidf	$\frac{idf(a_i)\times qidf(A_j^{[q]})}{\lvert a_i^{[q]}\rvert}$	adaptive	character q-grams & terms

6. Experimental evaluation

The proposed experiments aim to evaluate the text similarity functions based on SC spectra as baselines for several natural processing tasks, namely: information retrieval (IR) and entity resolution (ER) in subsection 6.1, paraphrase and textual entailment recognition in subsection 6.2. In addition, the experiments also intend to address the following issues: (i) to determine which of the different q-gram padding approaches are better suited for different tasks, (ii) to determine the suitability of the different weighting schemes at term and q-gram level, (iv) to determine whether the SC spectra aggregation is more convenient than individual SC q-spectrum, and (v) to compare the soft-cardinality spectra approach versus other approaches.

The different weighting schemes in the inner expression in 4.4 used in all experiments are listed using the naming convention given in Table 2.

6.1. **Information retrieval and entity resolution.** The classical information retrieval task is to find a ranked list of relevant documents for a set of queries (or information needs). The entity resolution task consists of given a pair of relations containing entities, finding all entity pairs that refer the same object. The entities are commonly represented as names or names extended with addresses and other information.

Information retrieval and entity resolution tasks usually involve large collections of documents and databases. Furthermore, using a naive approach these tasks involve the evaluation of a text similarity measure on the Cartesian product on the sets of queries and documents, or on the pair entity relations to be reconciled. Therefore, the use of semantic measures is restricted due to the considerable amount of resources that these measures require. The common practice is to use adaptive measures such as cosine TF-IDF. Besides, a static similarity measures such as the cosine similarity using Boolean weights is considered a baseline for these tasks.

6.1.1. *Data sets* . For the experimental evaluation, two groups of data sets were used for entity resolution and information retrieval tasks. The first group, called ER, consists of 11 data sets collected from different sources by the *secondstring framework*[2] creators. The second group, called IR, is composed of 9 "classical" collections described in [2][3]. Each data set consists of two series of texts and a gold-standard relation that associates pairs from both sets. The gold standard in all data sets was obtained from human judgments excluding *census* data set, which was built making random edit operations into a list of people names. In the ER data sets, the gold-standard relationship means identity equivalence and in IR data sets, it means relevance between a query and a document.

Texts in all data sets were divided into terms (i.e., tokenized) with a simple approach using as separator the space bar, punctuation characters, parenthesis and others special characters such as slash, hyphen, currency, tab, etc. Characters in all data sets were converted to their lowercase equivalents. Besides, no stop words removal or stemming was used either at ER or IR data sets.

6.1.2. *Performance measure.* The quality of the similarity function proposed in 5.2 can be measured quantitatively using various existing performance metrics for ER and IR tasks. We preferred to use interpolated average precision (IAP) because it is a performance measure that has been used at both tasks (see [2] for a detailed description). IAP allows to measure the two-ways classification performance (match vs. not-match and relevant vs. not relevant) of a ranked list of text pairs. While in IR, IAP reports the average measured in a different rank for each query, in ER a single rank for each data set is used. The reason for this is that the entity pairs in ER are texts of the same type, differently to the IR task where texts are whether queries or documents. Thus, while in IR it makes sense to evaluate the retrieved documents for each particular query; in ER it is more important to evaluate the ability of the similarity measure to separate the entire dataset into two groups of correct and incorrect pairs. In order to provide a consistent evaluation measure we used a single rank for both IR and ER tasks.

The ranking of text pairs is provided by ordering them from most to least similar using the similarity score obtained using the text similarity function to be evaluated. Precision at the position i in such ranking is $precision(i) = \frac{c(i)}{i}$, where $c(i)$ is the number of correct pairs ranked before position i. Recall at the position i is $recall(i) = \frac{c(i)}{m}$, where m is the total number of correct pairs. Interpolated precision at recall r is $\max_i (precision(i))$, where max is taken over all ranks i such that $recall(i) \geq r$. Values of interpolated precision are obtained at eleven evenly separated recall points: 0.0, 0.1, ..., 1.0. The obtained values can be used to plot a recall-precision curve such as those shown in Figure 6.2. Finally, interpolated average precision (IAP) is the area under the resulting recall-precision curve that can be obtained averaging the eleven interpolated precision values.

6.1.3. *Experiments.* For the experiments, 55 similarity functions were constructed with all possible SC spectra using q-spectrum ranging q from 1 to 10 in combination with 5.2. The weighting mechanism used in all experiments was weighting by local context, i.e. c in Table 2. Therefore, all the used measures based on SC spectra in this subsection were static measures.

Each similarity measure obtained was evaluated using all text pairs throughout the Cartesian product between both text sets in the 18 data sets. In addition, the following three padding approaches were tested: *single padding*, *full padding* and *not padding*.

For each one of the 2,970 (55 SC $[q_s : q_e]$-spectra by 18 data sets by 3 padding approaches) experiments carried out the IAP performance measure was calculated. Figure 6.1 shows a sample of the results for two data sets (*hotels* and *adi*) using *single padding* and *not padding* configurations respectively.

[2]http://secondstring.sourceforge.net/

[3]http://people.ischool.berkeley.edu/~hearst/irbook/

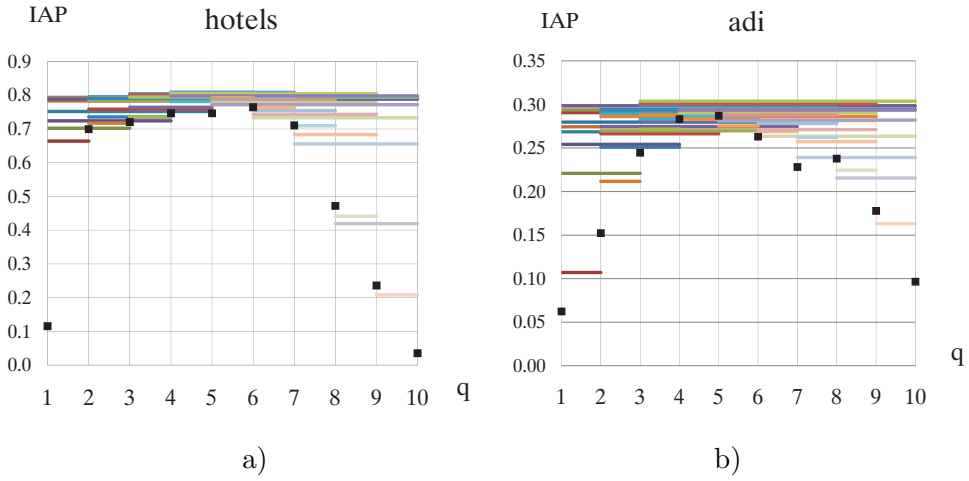

FIGURE 6.1. IAP performance for all SC $[q_s : q_e]$-spectra form $q_s = 1$ to $q_e = 10$ for data sets *hotels* and *adi*. Spectra with single q-spectrum are depicted with black squares (e.g. $[3 : 3]$). Wider SC spectra are depicted with horizontal colored bars.

TABLE 3. Results for best SC spectra using ER data sets

	full padding		single padding		not padding	
DATA SET	spectra	IAP	spectra	IAP	spectra	IAP
birds-scott1	**[1:2]***	**0.9091**	**[1:2]***	**0.9091**	**[1:2]***	**0.9091**
birds-scott2	[7:8]*	0.9005	**[6:10]**	**0.9027**	[5:9]	0.9007
birds-kunkel	[5:7]*	0.8804	**[6:6]**	**0.8995**	[4:4]	0.8947
birds-nybird	[4:6]	0.7746	**[1:7]**	**0.7850**	[4:5]	0.7528
business	[1:3]	0.7812	**[1:4]**	**0.7879**	[1:4]	0.7846
demos	**[2:2]**	**0.8514**	**[2:2]**	**0.8514**	[1:3]	0.8468
parks	[2:2]	0.8823	[1:9]	0.8879	**[2:4]**	**0.8911**
restaurant	[1:6]	0.9056	**[3:7]**	**0.9074**	**[1:6]**	**0.9074**
ucd-people	**[1:2]***	**0.9091**	**[1:2]***	**0.9091**	**[1:2]***	**0.9091**
hotels	[3:4]	0.7279	[4:7]	0.8083	**[2:5]**	**0.8147**
census	[2:2]	0.8045	**[1:2]**	**0.8110**	[1:2]	0.7642
Average		0.8478		**0.8599**		0.8522

* Asterisks indicate that another wider SC-spectra also got the same IAP performance.

6.1.4. *Results.* Tables 3 and 4 show the best SC spectra for each data set using the three proposed padding approaches. Figure 6.2 shows comparison of recall-precision curves for SC spectra against other measures. The series named "best SC spectra c" is the average of the best SC spectra for each data set using *single padding* for ER and *not padding* for IR. The SoftTFIDF measure was used with the same configuration proposed in [8]: $\theta = 0.9$ and the Jaro-Winkler measure as auxiliary similarity function. The series labeled "Soft Cardinality" in Figure 6.2 a) used 3.3 with $p = 2$ and the auxiliary inter-term similarity function was the Jaccard coefficient of the set of character bigrams for each term. The series labeled as "Cosine boolean" and "SoftTFIDF boolean" are the measures described in Section 2 but using Boolean instead of *tf-idf* weights.

TABLE 4. Results for best SC spectra using IR collections.

	full padding		single padding		not padding	
DATA SET	spectra	IAP	spectra	IAP	spectra	IAP
cran	**[7:9]**	**0.0070**	[3:4]	0.0064	[3:3]	0.0051
med	[4:5]	0.2939	**[5:7]***	**0.3735**	[4:6]	0.3553
cacm	**[4:5]**	**0.1337**	[2:5]	0.1312	[2:4]	0.1268
cisi	[1:10]	0.1368	[5:8]	0.1544	**[5:5]**	**0.1573**
adi	[3:4]	0.2140	[5:10]	0.2913	**[3:10]**	**0.3037**
lisa	[3:5]	0.1052	[5:8]	0.1244	**[4:6]**	**0.1266**
npl	[7:8]	0.0756	[3:10]	0.1529	**[3:6]**	**0.1547**
time	[1:1]	0.0077	[8:8]	0.0080	**[6:10]**	**0.0091**
cf	[7:9]	0.1574	[5:10]	0.1986	**[4:5]**	**0.2044**
Average		0.1257		0.1601		**0.1603**

* Asterisks indicate that another wider SC-spectra also got the same IAP performance.

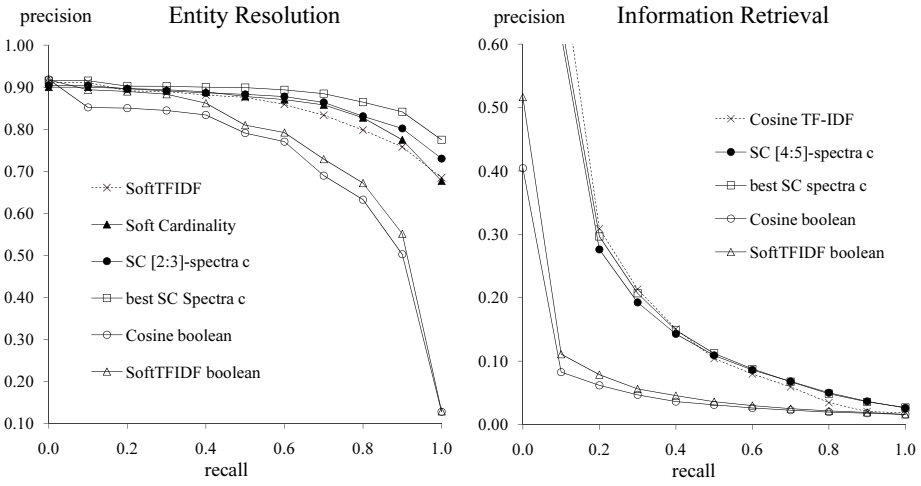

FIGURE 6.2. Recall-precision curves of SC spectra compared with other measures.

Tables 5 and 6 present the results of different weighting schemes listed in Table 2 for each of the data sets in the ER and IR groups. A single spectrum similarity measure SC [4 : 4]-spectra (i.e. SC 4-spectrum or quadgrams) was used to obtain all the results in both tables.

6.1.5. *Discussion.* Results Tables 3 and 4 indicate that the use of a single padding character seem to be more useful in ER data sets than in IR collections. Apparently, the effect of the addition of padding characters is important only in collections with relatively short texts.

Best performance settings (shown in bold in the tables) were reached in most cases (15 over 18) using SC spectra instead of a single SC q-spectrum. This result can also be observed in Figure 6.1, where SC spectra results (represented as horizontal bars) tended to overcome SC q-spectrum (represented as small black squares). The average relative improvement of the best SC spectra for each data set compared to the best SC q-spectrum was 1.33% for ER data sets and 4.48% for IR collections. In addition, Figure 6.1 qualitatively shows that the SC spectra measures outperformed the SC q-spectrum measures. For instance, SC [7 : 9]-spectra at *adi* collection outperforms all SC 7-spectrum, SC 8-spectrum and SC 9-spectrum measures.

TABLE 5. IAP results in *entity resolution* (ER) data sets for different weighting schemas using quadgrams.

DATA SET	*c.idf*	*c.idf.qidf*	*c.qidf*	*idf.qidf*	*idf*	*c*	*qidf*	*none*
birds-scott1	0.9132	**0.9181**	0.9140	0.9167	0.9135	0.9136	0.9143	0.9180
birds-scott2	**0.9419**	0.9372	0.9331	0.9070	0.9082	0.9288	0.9106	0.9115
birds-kunkel	0.9205	**0.9445**	0.9014	0.8880	0.8703	0.7117	0.8592	0.8113
birds-nybird	**0.7971**	0.7908	0.7817	0.7312	0.7633	0.8077	0.7222	0.7531
business	**0.8126**	0.8038	0.8006	0.7793	0.7231	0.7385	0.7115	0.4948
demos	**0.4530**	0.4481	0.4518	0.4503	0.4184	0.4485	0.4226	0.4120
parks	0.9320	**0.9427**	0.9144	0.9304	0.9230	0.8531	0.9214	0.9024
restaurant	0.9752	**0.9820**	0.9614	0.9791	0.9530	0.9102	0.9311	0.8836
ucd-people	**1.0000**	**1.0000**	0.9980	0.9818	0.9755	0.9729	0.9720	0.9606
hotels	0.7369	0.7303	0.7229	0.7572	**0.7651**	0.6721	0.7558	0.7322
census	0.6253	0.6049	0.6225	0.4927	0.4985	**0.6474**	0.4808	0.4788
Average	**0.8280**	0.8275	0.8184	0.8012	0.7920	0.7822	0.7820	0.7508

TABLE 6. IAP results in *information retrieval* (IR) data sets for different weighting schemes using quadgrams.

DATA SET	*qidf*	*c.idf.qidf*	*c.idf*	*none*	*idf.qidf*	*c.qidf*	*idf*	*c*
cf	0.2135	**0.2346**	0.2320	0.2006	0.2205	0.1848	0.2149	0.1659
cacm	**0.1383**	0.1112	0.1082	0.0847	0.0854	0.1367	0.0839	0.1220
cisi	**0.1584**	0.1087	0.1089	0.1526	0.0939	0.1526	0.0873	0.1438
adi	**0.3255**	0.2848	0.2979	0.3056	0.3197	0.2631	0.3006	0.1896
cran	0.0028	0.0028	0.0028	0.0027	0.0029	0.0035	0.0029	**0.0040**
lisa	0.1473	**0.1609**	0.1519	0.1302	0.1684	0.1155	0.1600	0.1044
med	0.3389	**0.3484**	0.3412	0.3287	0.3127	0.3173	0.2995	0.2868
time	0.0058	0.0060	0.0060	0.0060	0.0063	0.0063	0.0064	**0.0065**
Average	**0.1663**	0.1572	0.1561	0.1514	0.1512	0.1475	0.1445	0.1279

In recall-precision curves, better measures are closer to the upper bound precision level of 1. As shown in Figure 6.2 a) the similarity measures obtained using the best SC spectra for each data set overcame the other tested measures in ER data sets. As for IR, Figure 6.2 b) shows that SC spectra reached almost the same performance than *cosine tf-idf*. This result is also remarkable because we are reaching equivalent performance using considerably less information (not term weighting). Finally, the series for ER show that SC spectra is a better soft cardinality approximation than the approximation using 3.3. Besides, SC spectra require considerably less computational effort than that approximation (linear versus quadratic complexity). The results obtained with the SC spectra approach showed that it is possible to recombine the information contained only in the pair of texts being compared (static approach) in a better way to reach a performance equivalent (or even better in ER) in comparison with approaches that use additional information gathered from the entire text collection (adaptive approach).

The results obtained with the ER data sets for the different weighting criteria (Table 5) were ranked as it was expected to be. That is, the approaches that combined two or three weighting evidences (*c.idf*, *c.idf.qidf*, *c.qidf* and *idf.qidf*) outperformed the approaches that used only one evidence (*idf*, *c* and *qidf*), and in turn the approach that did not use weights at all (*none*) obtained the lowest performance. Unlike ER, the results using IR collections (Table 6) were less predictable. However, the approach that combined all the proposed weighting evidences (*c.idf.qidf*) reached the second best performance proving effective in this unexpected scenario.

6.2. **Paraphrase and textual entailment recognition.** Paraphrase and textual entailment recognition are tasks in which a pair of text passages is considered and it is necessary to decide whether one text is a paraphrase of the other, or if one text is entailed by the other. While paraphrasing is a symmetrical task, in the textual entailment task one the texts is labeled as "T" (the entailing text) and the other as "H" (the entailed text). Thus, "T" entails "H" if a human reading "T" would infer that "H" is probably true. Each text pair is provided with a gold standard obtained from human judgments.

Given that those tasks require a higher "understanding" of the text that may require information not included in the surface text information, the methods that deal with those tasks generally use semantic approaches. For these tasks, static and adaptive approaches are considered as baselines because they cannot reveal any underlying semantic structure hidden in the text. Semantic methods aim to reveal that structure using an assortment of resources that exploit statistical evidence (e.g. large corpora) and/or knowledge (e.g. parsers, semantic networks).

To evaluate the baseline role of the measures outlined in Section 2 and the SC spectra approach, we used standard data sets for the tasks and compared our results against the results already published.

6.2.1. *Data sets.* The data sets selected for evaluation were the Microsoft Research Paraphrase Corpus [15] (MSR paraphrase) and the RTE-3 data set [18] from the third PASCAL recognizing textual entailment challenge (2007). Both data sets have been extensively studied and dozens of papers have reported results using them. See [15, 18] for a comprehensive description of the data sets. We used the test partition of the RTE-3 to allow comparison of performance with the results published in [18].

Both data sets were preprocessed by tokenizing, lowercase character conversion and stemming using the Porter stemmer [36]. Besides, q-gram partitions were obtained using *not padding* because this approach is the simplest and it obtained the best results for the IR task, whose text type (documents) is similar to those used in paraphrase and textual entailment.

6.2.2. *Performance measure.* The similarity scores provided a ranking of text pairs where higher scores were considered as valid paraphrases and entailments. Accuracy, precision, recall and F-measure [2] standard metrics were calculated at each position on the ranking using a gold standard. The results are reported on the ranking position with the best F-measure (i.e. F1-score). The main performance measure for comparison was accuracy, i.e. the number or correct predictions over the total number of predictions.

6.2.3. *Experiments.* The used q-gram partition in all experiments was SC $[1:4]$-spectra using *not padding* approach. The generalized mean coefficient 5.1 was used as similarity measure and the parameter p was adjusted to obtain the best performance. That is, $p = 10$ for paraphrase and $p = -1.5$ for textual entailment. We present the results only for the *idf* weighting scheme for SC spectra because this approach slightly outperformed the *tf-idf* weights in all experiments.

We also tested two additional adaptive baselines such as *softTFIDF* and *cosine TF-IDF.* SoftTFIDF had the same configuration used in the previous subsection, but using the threshold $\theta = 0.7$, which was the threshold with the best results.

6.2.4. *Results.* Table 7 shows the accuracies obtained for the different weighting schemas listed in Table 2. Table 8 shows a sorted summary of the published results for the MSR paraphrase corpus (extracted in part from [1]) including the best results obtained using SC $[1:4]$-spectra, softTFIDF and cosine TF-IDF.

Table 9 shows a summary (extracted from [18]) of the accuracies and resources used to obtain each listed result. The table shows only the participating systems in the *third PASCAL recognizing textual entailment challenge* (2007) that included in the used resources a similarity measures based on q-grams or words.

TABLE 7. Accuracy results in *paraphrase* and *textual entailment recognition* data sets for different weighting schemes using SC [1 : 4]-spectra.

DATA SET	*c.idf.qidf*	*idf.qidf*	*qidf*	*none*	*idf*	*c.idf*	*c.qidf*	*c*
MSR paraphrase	0.7252	0.7102	0.7064	0.7045	0.7128	0.7250	0.7232	**0.7331**
RTE-3	0.6688	0.6800	**0.6825**	0.6800	0.6713	0.6575	0.6575	0.6275
Average	**0.6970**	0.6951	0.6945	0.6923	0.6920	0.6913	0.6903	0.6803

TABLE 8. *Paraphrase recognition* SC spectra results in MSR-paraphrase corpus compared with published results.

Method	accu.	prec.	recall	F1	Used resources
Malakasiotis [31]	**0.762**	0.794	0.868	0.829	WordNet, ML, dependency parser
Das & Smith [10]	0.761	**0.796**	0.861	0.829	dependency grammars, ML
Wan *et al.* [42]	0.756	0.770	0.900	**0.830**	syntactic dependencies, ML
Finch *et al.* [17]	0.750	0.766	0.898	0.827	machine translation, POS tagger, ML
SC [1 : 4]-spectra (*c*)	0.733	0.739	0.931	0.824	surface text (this paper)
SC [1 : 4]-spectra (*c.idf.qidf*)	0.725	0.726	0.951	0.823	surface text (this paper)
Lintean *et al.* [30]	0.724	0.739	0.903	0.672	dependency parser, WordNet
Qiu *et al.* [37]	0.720	0.725	0.934	0.816	syntactic parser, thesaurus, ML
Zhang & Patrick [45]	0.719	0.743	0.882	0.807	text canonicalization, ML
softTFIDF [8]	0.716	0.717	0.955	0.819	surface text (this paper)
Coreley & Mihalcea [9]	0.715	0.723	0.925	0.812	WordNet, BNC statistics
Do *et al.* [14]	0.711	0.748	0.861	0.800	WordNet
cosine TF-IDF [38]	0.707	0.706	**0.968**	0.816	surface text (this paper)

6.2.5. *Discussion.* The results in Table 7 show that, even though *c* and *qidf* obtained the higher accuracy using MSR paraphrase and RTE-3 respectively, in average the *c.idf.qidf* and *idf.qidf* combination obtained the best results, i.e. average accuracies of 0.6970 and 0.6951 respectively. This result showed that adaptiveness combined at term and *q*-gram level is an effective approach. It is interesting to note that, even though the *q*-gram context weighting approach *c* obtained the lowest average accuracy, it contributed to the highest accuracy obtained by *c.idf.qidf.* Also it is important to note that, the adaptiveness at character *q*-gram level (*qidf*) overcame the adaptiveness at term level (*idf*). This result is remarkable because (to the best fo our knowledge) while the weighting at term level is a common practice, this is the first attempt to use weighting at character *q*-gram level on these tasks. Gravano *et al.* [19] tested bigrams and trigrams as tokens instead of terms in the softTFIDF measure, but they obtained worse results using *q*-grams than using terms as tokens.

The comparison of the results obtained by SC [1 : 4]-spectra (Table 8) against the already published results for the MSR paraphrase data set are encouraging. The measure SC [1 : 4]-spectra (*c*), which is A static approach, obtained a result with a difference of only 0.029 in accuracy versus the best results published to the date. Besides, our approach used considerably less information and its computation and reproducibility is rather simple (linear complexity). Our static and adaptive baselines also outperformed the other tested baselines softTFIDF and cosine TF-IDF.

The proposed baselines used in the RTE-3 data set also obtained encouraging results. As Table 9 shows, SC [1 : 4]-spectra (*qidf*) which is an adaptive measure, outperformed most of the other approaches that also included "*q*-gram\word similarity" in their resources. In fact, our measure only used that resource.

TABLE 9. Textual entailment recognition on RTE-3 data set.

First author (see [18])	accuracy	q-gram\word similarity	Lexical Relation, WordNet	Syntactic Matching\Aligning	Large corpus statistics	Machine Learning	Logical Inference	Anaphora resolution	Entailment Corpora
Hickl	0.8000	×	×			×	×	×	×
SC $[1:4]$-spectra ($qidf$)	0.6825	×							
Adams	0.6700	×	×		×	×			
SoftTFIDF (idf)	0.6638	×							
Ferrandez	0.6563	×	×	×					
Li	0.6400	×	×			×			
cosine TF-IDF (idf)	0.6325	×							
Rodrigo	0.6312	×	×	×		×			
Roth	0.6262	×	×						×
Settembre	0.6262	×	×			×			
Malakasiotis	0.6175	×				×			
Ferrándes	0.6150	×	×			×			
Montejo-Ráez	0.6038	×	×	×		×			
Burek	0.5500	×			×				

7. Related work

The proposed weighting scheme that gives smaller weights to q-grams according to the length in characters of each term (c) is similar to the approach proposed in [11] that assigned a variable cost to character edit operations to Levenshtein edit distance. Using this approach in a text classification task an improved performance was obtained versus the original edit distance. This approach is equivalent to ours because the contribution of each q-gram to the soft cardinality depends on the total number of q-grams in the term, which in turn depends on the length in characters of the term.

The approach of aggregating information from sets of subsequences of different sizes was recently proposed in [3] using time series sequences. Even though, the application of this approach was considerably different, our approach of aggregating q-grams of different sizes is analogous.

Leslie *et al.* [27] proposed a k-spectrum kernel for comparing sequences using substrings of k-length in a protein classification task. Similarly to them, we use the same metaphor to name our approach.

8. Conclusions

We found that the proposed SC spectra method offers good baselines for various tasks of natural language processing. In particular, when using local context q-gram weights, the SC spectra approach provided a new static measure that outperformed other static baselines such as the cosine similarity with binary vectors. This new static baseline, which only used the surface information in the pair of texts being compared, obtained better performance than other adaptive approaches that used weights collected from the entire text collection (e.g. cosine tf-idf).

For the paraphrase and textual entailment recognition tasks, which involve semantics, we proposed a SC spectra adaptive baseline. This new text similarity measure used a weighting mechanism at character q-gram level based on a combination of evidence from: i) the local context of the q-gram, ii) the *idf* weight of the term in which the q-gram occurred, and iii) *idf*-like weights at q-gram level. This combined weighting scheme obtained better results in both tasks when compared with weighting approaches based on single evidence. In addition, the proposed adaptive measure was a fairly good baseline for other semantic measures that used additional linguistic resources based on knowledge and/or large corpora. The proposed baseline reached performances close to the best published results while outperformed many other semantic approaches.

We have shown that the soft cardinality *spectra* (SC spectra) approach has the necessary characteristics to be a good baseline method: simplicity, speed (linear complexity in the length of the text being compared) and performance.

Acknowledgements

This research was funded in part by the Systems and Industrial Engineering Department, the Office of Student Wellfare of the National University of Colombia, Bogota, and throught a grant from the Colombian Department for Science, Technology and Innovation Colciencias, project 110152128465. The second author acknowledges support from Mexican Government (SNI, COFAA-IPN, SIP 20121823, CONACYT 50206-H) and CONACYT–DST India project 122030 2011–2014 "Answer Validation through Textual Entailment".

References

[1] Ion Androutsopoulos and Prodromos Malakasiotis. A survey of paraphrasing and textual entailment methods. *J. Artif. Int. Res.*, 38(1):135–187, May 2010.

[2] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley & ACM Press, 1999.

[3] Ildar Batyrshin, Raul Herrera-Avelar, Leonid Sheremetov, and Aleksandra Panova. Moving approximation transform and local trend associations in time series data bases. In Ildar Batyrshin, Janusz Kacprzyk, Leonid Sheremetov, and Lotfi Zadeh, editors, *Perception-based Data Mining and Decision Making in Economics and Finance*, volume 36 of *Studies in Computational Intelligence*, pages 55–83. Springer Berlin / Heidelberg, 2007.

[4] Mikhail Bilenko, Raymond Mooney, William Cohen, Pradeep Ravikumar, and Stephen Fienberg. Adaptive name matching in information integration. *IEEE Intelligent Systems*, 18(5):16–23, 2003.

[5] Surajit Chaudhuri, Kris Ganjam, Venkatesh Ganti, and Rajeev Motwani. Robust and efficient fuzzy match for online data cleaning. In *Proceedings of the 2003 ACM SIGMOD international conference on management of data*, pages 313–324, San Diego, California, 2003. ACM.

[6] Peter Christen. A comparison of personal name matching: Techniques and practical issues. In *Data Mining Workshops, International Conference on*, pages 290–294, Los Alamitos, CA, USA, 2006. IEEE Computer Society.

[7] Rudi Cilibrasi and Paul Vitányi. Clustering by compression. *IEEE Transactions on Information Theory*, pages 1523–1545, 2005.

[8] William W Cohen, Pradeep Ravikumar, and Stephen E Fienberg. A comparison of string distance metrics for Name-Matching tasks. In *Proceedings of the IJCAI2003 Workshop on Information Integration on the Web IIWeb03*, pages 73–78, August 2003.

[9] Courtney Corley and Rada Mihalcea. Measuring the semantic similarity of texts. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, EMSEE '05, pages 13–18, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[10] Dipanjan Das and Noah A. Smith. Paraphrase identification as probabilistic quasi-synchronous recognition. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, page 468–476, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[11] Colin de la Higuera and Luisa Mico. A contextual normalised edit distance. In *2008 IEEE 24th International Conference on Data Engineering Workshop*, pages 354–361, Cancun, Mexico, 2008.

[12] Lee R. Dice. Measures of the amount of ecologic association between species. *Ecology*, pages 297–302, 1945.

[13] Thomas Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems*, volume 1857 of *Lecture Notes in Computer Science*, pages 1–15. Springer Berlin / Heidelberg, 2000.
[14] Quang Xuan Do, Dan Roth, Mark Sammons, Yuancheng Tu, and V. G. Vinod Vydiswaran. Robust, light-weight approaches to compute lexical similarity. Technical report, 2010.
[15] Bill Dolan, Chris Quirk, and Chris Brockett. Unsupervised construction of large paraphrase corpora: exploiting massively parallel news sources. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.
[16] Susan Duma. Latent semantic analysis. *ARIST Review of Information Science and Technology*, 2004.
[17] Andrew Finch, Young-Sook Hwang, and Eiichiro Sumita. Using machine translation evaluation techniques to determine sentence-level semantic equivalence. In *Proceedings of the 3rd Int. Workshop on Paraphrasing*, Jeju Island, Korea, 2005.
[18] Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, RTE '07, pages 1–9, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.
[19] L. Gravano, Panagiotis G. Ipeirotis, Nick Koudas, and Divesh Srivastava. Text joins in a RDBMS for web data integration. 2003.
[20] Paul Jaccard. Etude comparative de la distribution florare dans une portion des alpes et des jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, pages 547–579, 1901.
[21] Sergio Jimenez, Claudia Becerra, and Alexander Gelbukh. Soft cardinality: A parameterized similarity function for text comparison. In *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, 2012.
[22] Sergio Jimenez, Fabio Gonzalez, and Alexander Gelbukh. Text comparison using soft cardinality. In Edgar Chavez and Stefano Lonardi, editors, *String Processing and Information Retrieval*, volume 6393 of *LNCS*, pages 297–302. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
[23] Sergio Jiménez Vargas and Alexander Gelbukh. SC spectra: A Linear-Time soft cardinality approximation for text comparison. In *Advances in Soft Computing*, volume 7095 of *Lecture Notes in Computer Science*, pages 213–224. Springer Berlin / Heidelberg, 2011.
[24] H. Keskustalo, A. Pirkola, K. Visala, and E. Leppanen. Non-adjacent digrams improve matching of cross-lingual spelling variants. In *LNCS 2857*, pages 252–265, Manaus, Brazil, 2003.
[25] Karen Kukich. Techniques for automatically correcting words in text. *ACM Computing Surveys*, 24:377–439, December 1992.
[26] Yulia Ledeneva and Grigori Sidorov. Recent advances in computational linguistics. *Informatica. International Journal of Computing and Informatics*, 34(1):3–18, 2010.
[27] Christina Leslie, Eleazar Eskin, and William Stafford Noble. The spectrum kernel: A string kernel for SVM protein classification. In *Biocomputing 2002 - Proceedings of the Pacific Symposium*, pages 564–575, Kauai, Hawaii, USA, 2001.
[28] Vladimir I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966.
[29] Dekang Lin. An Information-Theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann Publishers Inc., 1998.
[30] Mihai Lintean and Vasile Rus. Paraphrase identification using weighted dependencies and word semantics. In *Proceedings of the Twenty-Second International FLAIRS Conference*, 2009.
[31] Prodromos Malakasiotis. Paraphrase recognition using machine learning to combine similarity measures. In *Proceedings of the ACL-IJCNLP 2009 Student Research Workshop*, ACLstudent '09, pages 27–35, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.
[32] Alvaro E. Monge and Charles Elkan. The field matching problem: Algorithms and applications. In *Proceeding of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*, pages 267–270, Portland, OR, 1996.
[33] Erwan Moreau, François Yvon, and Olivier Cappé. Robust similarity measures for named entities matching. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, pages 593–600, Manchester, United Kingdom, 2008. Association for Computational Linguistics.
[34] Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. WordNet::Similarity: measuring the relatedness of concepts. In *Proceedings HLT-NAACL–Demonstration Papers*, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.
[35] Jakub Piskorski and Marcin Sydow. Usability of string distance metrics for name matching tasks in polish. 2008.
[36] Martin Porter. An algorithm for suffix stripping. *Program*, 3(14):130–137, October 1980.

[37] Long Qiu, Min-Yen Kan, and Tat-Seng Chua. Paraphrase recognition via dissimilarity significance classification. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, page 18–26, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[38] Gerard Salton, Andrew K. C. Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, 1975.

[39] B Sarker. The resemblance coefficients in group technology: A survey and comparative study of relational metrics. *Computers & Industrial Engineering*, 30(1):103–116, January 1996.

[40] Grigori Sidorov, Alberto Barrón-Cedeño, and Paolo Rosso. English-Spanish large statistical dictionary of inflectional forms. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 277–281, Valletta, Malta, 2010. European Language Resources Association (ELRA).

[41] Julian R. Ullmann. A binary N-Gram technique for automatic correction of substitution, deletion, insertion and reversal errors in words. *The Computer Journal*, 20(2):141–147, January 1977.

[42] Stephen Wan, Mark Dras, Robert Dale, and Cécile Paris. Using Dependency-Based features to take the para-farce out of paraphrase. In *Proceedings of the Australasian Language Technology Workshop*, Sydney, Australia, 2006.

[43] S. J Westerman, T. Cribbin, and J. Collins. Human assessments of document similarity. *Journal of the American Society for Information Science and Technology*, 61(8):1535–1542, April 2010.

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

[45] Yitao Zhang and Jon Patrick. Paraphrase identification by text canonicalization. In *Proceedings of the Australasian Language Technology Workshop*, Sydney, Australia, 2005.

Sergio Jimenez was graduated from the Department of Systems and Computer Engineering in the National University of Colombia (Universidad Nacional de Colombia), Bogota in 1993. After a career of 12 years in software development and IT (information technology) marketing industry, he returned to his alma mater in 2006 to pursue a Masters in Systems Engineering and Computing obtaining a cum ladue master's degree in 2009. Since 2010, he has been PhD candidate at the Universidad Nacional de Colombia, Bogota, Colombia.

Alexander Gelbukh received a PhD degree (Computer Science) from the All-Russian Institute of Scientific and Technical Information (VINITI) in 1995. Since 1997 he is a Research Professor and Head of the Natural Language and Text Processing Laboratory of the Center for Computing Research of the National Polytechnic Institute (IPN), Mexico, since 1998 he is a National Researcher of Mexico, currently with excellence level 2, and since 2000 he is an member of the Mexican Academy of Sciences. He is author of more than 400 publications, and editor of more than 50 books and special issues of journals, in the areas of computational linguistics and artificial intelligence. See more details on www.Gelbukh.com.