IRIT: Textual Similarity Combining Conceptual Similarity with an N-Gram Comparison Method

Davide Buscaldi, Ronan Tournier, Nathalie Aussenac-Gilles and **Josiane Mothe**
IRIT
118 Route de Narbonne
Toulouse (France)
{davide.buscaldi,ronan.tournier}@irit.fr,
{nathalie.aussenac,josiane.mothe}@irit.fr

Abstract

This paper describes the participation of the IRIT team to SemEval 2012 Task 6 (Semantic Textual Similarity). The method used consists of a n-gram based comparison method combined with a conceptual similarity measure that uses WordNet to calculate the similarity between a pair of concepts.

1 Introduction

The system used for the participation of the IRIT team (composed by members of the research groups SIG and MELODI) to the Semantic Textual Similarity (STS) task (Agirre et al., 2012) is based on two sub-modules:

- a module that calculates the similarity between sentences using n-gram based similarity;
- a module that calculates the similarity between concepts in the two sentences, using a concept similarity measure and WordNet (Miller, 1995) as a resource.

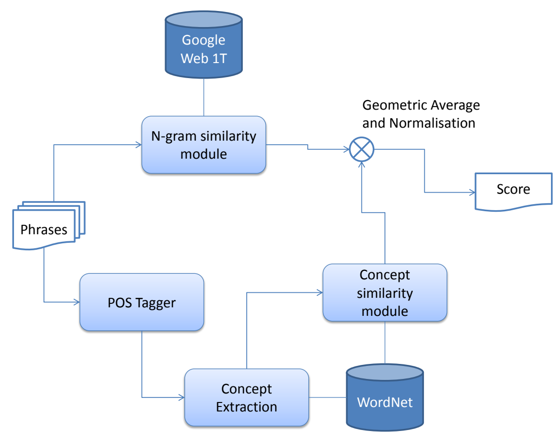

Figure 1: Schema of the system.

In Figure 1, we show the structure of the system and the connections between the main components. The input phrases are passed on one hand directly to the n-gram similarity module, and on the other they are annoted with the Stanford POS Tagger (Toutanova et al., 2003). All nouns and verbs are extracted from the tagged phrases and WordNet is searched for synsets corresponding to the extracted nouns and nouns associated to the verbs by the *derived terms* relationship. The synsets are the concepts used by the conceptual similarity module to calculate the concept similarity. Each module calculates a similarity score using its own method; the final similarity value is calculated as the geometric average between the two scores, multiplied by 5 in order to comply with the task specifications.

The n-gram based similarity relies on the idea that two sentences are semantically related if they contain a long enough sub-sequence of non-empty terms. Google Web 1T (Brants and Franz, 2006) has been used to calculate term idf, which is used as a measure of the importance of the terms. The conceptual similarity is based on the idea that, given an ontology, two concepts are semantically similar if their distance from a common ancestor is small enough. We used three different measures: the Wu-Palmer similarity measure (Wu and Palmer, 1994) and two "Proxigenea" measures (Dudognon et al., 2010). In the following we will explain in detail how

*First Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 552–556,
Montréal, Canada, June 7-8, 2012. ©2012 Association for Computational Linguistics

each similarity module works.

2 N-Gram based Similarity

N-gram based similarity is based on the Clustered Keywords Positional Distance (CKPD) model proposed in (Buscaldi et al., 2009). This model was originally proposed for passage retrieval in the field of Question Answering (QA), and it has been implemented in the JIRS system[1]. In (Buscaldi et al., 2006), JIRS showed to be able to obtain a better answer coverage in the Question Answering task than other traditional passage retrieval models based on Vector Space Model, such as *Lucene*[2]. The model has been adapted for this task by calculating the idf weights for each term using the frequency value provided by Google Web 1T.

The similarity between a text fragment (or passage) p and another text fragment q is calculated as:

$$Sim(p,q) = \frac{\sum_{\forall x \in Q} h(x,P)\frac{1}{d(x,x_{max})}}{\sum_{i=1}^{n} w_i} \quad (1)$$

Where P is the set of *n*-grams with the highest weight in p, where all terms are also contained in q; Q is the set of all the possible j-grams in q and n is the total number of terms in the longest passage. The weights for each term and each n-gram are calculated as:

- w_i calculates the weight of the term t_I as:

$$w_i = 1 - \frac{log(n_i)}{1 + log(N)} \quad (2)$$

  Where n_i is the frequency of term t_i in the Google Web 1T collection, and N is the frequency of the most frequent term in the Google Web 1T collection.

- the function $h(x,P)$ measures the weight of each *n*-gram and is defined as:

$$h(x,P_j) = \begin{cases} \sum_{k=1}^{j} w_k & \text{if } x \in P_j \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

  Where w_k is the weight of the *k-th* term (see Equation 2) and *j* is the number of terms that compose the n-gram x;

- $\frac{1}{d(x,x_{max})}$ is a distance factor which reduces the weight of the *n*-grams that are far from the heaviest *n*-gram. The function $d(x,x_{max})$ determines numerically the value of the separation according to the number of words between a *n*-gram and the heaviest one. That function is defined as show in Equation 4 :

$$d(x,x_{max}) = 1 + k \cdot ln(1+L) \quad (4)$$

  Where *k* is a factor that determines the importance of the distance in the similarity calculation and *L* is the number of words between a *n*-gram and the heaviest one (see Equation 3). In our experiments, k was set to 0.1, a default value used in JIRS.

For instance, given the following two sentences: "*Mr. President, enlargement is essential for the construction of a strong and united European continent*" and "*Mr. President, widening is essential for the construction of a strong and plain continent of Europe*", the longest n-grams shared by the two sentences are: "*Mr. President*", "*is essential for the construction of a strong and*", "*continent*".

term	$w(term)$
Mr	0.340
President	0.312
is	0.159
essential	0.353
for	0.153
the	0.104
construction	0.332
of	0.120
a	0.139
strong	0.329
and	0.121
continent	0.427
of	0.120
Europe	0.308
widening	0.464

Table 1: Term weights (idf) calculated using the frequency for each term in Google Web 1T unigrams set.

[1] http://sourceforge.net/projects/jirs/

[2] http://lucene.apache.org/

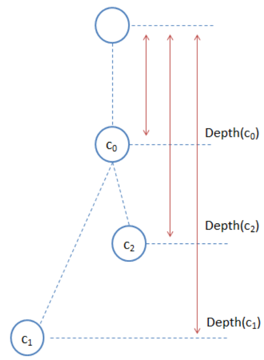

Figure 2: Visualisation of depth calculation.

The weights have been calculated with Formula 2, using the frequencies from Google Web 1T. The weights for each of the longest n-grams are 0.652, 1.809 and 0.427 respectively; their sum is 2.888 which divided by all the term weights contained in the sentence gives 0.764 which is the similarity score between the two sentences as calculated by the n-gram based method.

3 Conceptual Similarity

Given C_p and C_q as the sets of concepts contained in sentence p and q, respectively, with $|C_p| \geq |C_q|$, the conceptual similarity between p and q is calculated as:

$$ss(p,q) = \frac{\sum_{c_1 \in C_p} \max_{c_2 \in C_q} s(c_1, c_2)}{|C_p|} \tag{5}$$

where $s(c_1, c_2)$ is a concept similarity measure. Concept similarity can be calculated by different ways. Wu and Palmer introduced in (Wu and Palmer, 1994) a concept similarity measure defined as:

$$s(c_1, c_2) = \frac{2 \cdot d(c_0)}{d(c_1) + d(c_2)} \tag{6}$$

c_0 is the most specific concept that is present both in the synset path of c_1 and c_2 (see Figure 2 for details). The function returning the depth of a concept is noted with d.

3.1 ProxiGenea

By making an analogy between a family tree and the concept hierarchy in WordNet, (Dudognon et al., 2010; Ralalason, 2010) proposed a concept similarity measure based on the principle of evaluating the proximity between two members of the same family. The measure has been named "ProxiGenea" (from the french Proximité Généalogique, genealogical proximity). We took into account three versions of the ProxiGenea measure:

$$pg_1(c_1, c_2) = \frac{d(c_0)^2}{d(c_1) * d(c_2)} \tag{7}$$

This measure is very similar to the Wu-Palmer similarity measure, but it emphasizes the distances between concepts;

$$pg_2(c_1, c_2) = \frac{d(c_0)}{d(c_1) + d(c_2) - d(c_0)} \tag{8}$$

In this measure, the more are the elements which are not shared between the paths of c_1 and c_2, the more the score decreases. However, if the elements are placed more deeply in the ontology, the decrease is less important.

$$pg_3(c_1, c_2) = \frac{1}{1 + d(c_1) + d(c_2) - 2 \cdot d(c_0)} \tag{9}$$

In Table 2 we show the weights that have been calculated for each concept, using all the above similarity measures, and the concept that provided the maximum weight. No Word Sense Disambiguation process is carried out; therefore, the scores are calculated taking into account all the possible senses for the word. If the same concept is present in both sentences, it obtains always a score of 1. In the other cases, the maximum similarity value obtained with any other concept is retained.

From the example in Table 2 we can see that Wu-Palmer tends to give to the concepts a higher similarity value than Proxigenea3.

The final score for the above example is calculated as the geometric mean between the scores obtained in Table 2 and 0.764 obtained from the n-gram based similarity module, multiplied by 5. Therefore, for each similarity measure, the final scores of the example are, respectively: 4.029, 3.869, 3.921 and 3.703. The correct similarity value, according to the gold standard, was 4.600.

c_1, c_2	wp	pg_1	pg_2	pg_3
Mr Mr	1.000	1.000	1.000	1.000
President President	1.000	1.000	1.000	1.000
construction construction	1.000	1.000	1.000	1.000
continent continent	1.000	1.000	1.000	1.000
Europe continent	0.400	0.160	0.250	0.143
widening enlargement	0.737	0.544	0.583	0.167
score	0.850	0.784	0.805	0.718

Table 2: Maximum conceptual similarity weights using the different formulae for the concepts in the example. c_1: first concept, c_2: concept for which the maximum similarity value was calculated. wp: Wu-Palmer similarity; pg_X: Proxigenea similarity. *score* is the result of (5).

4 Evaluation

Before the official runs we carried out an evaluation to select the best similarity measures over the training set provided by the organisers. The results of this evaluation are shown in Table 3. The measure selected is the normalised Pearson correlation (Agirre et al., 2012). We evaluated also the use of the product instead of the geometric mean for the combination of the two scores.

Geometric mean				
	MSRpar	MSRvid	SMT-Eur	All
pg1	0.489	0.602	0.587	0.559
pg2	0.490	0.596	0.586	0.558
pg3	0.470	**0.657**	0.552	**0.560**
wp	**0.494**	0.572	**0.592**	0.552
Scalar product				
	MSRpar	MSRvid	SMT-Eur	All
pg1	0.469	0.601	0.487	**0.519**
pg2	0.471	0.597	0.487	0.518
pg3	0.447	**0.637**	0.459	0.514
wp	**0.476**	0.577	**0.492**	0.515

Table 3: Results on training corpus, comparison of different conceptual similarity measures and combination method. Top: geometric mean, bottom: product.

We used these results to select the final configurations for our participation to the STS task: we selected to exclude Proxigenea 2 and to use the geometric mean to combine the scores of the n-gram based similarity module and the conceptual similarity module. Wu-Palmer similarity allowed to obtain the best results on two train sets but Proxigenea 3 was the similarity measure that obtained the best average score thanks to the good result on MSRvid.

The official results obtained by our system are shown in Table 4, with the ranking obtained for each test set. We could observe that the system was well behind the best system in most test sets, except for SMTeur. This was expected since our system does not use a machine learning approach and is completely unsupervised, while the best systems used supervised learning. We observed also that the behaviour of the concept similarity measures was different from the behaviour on the training sets. In the competition, the best results were always obtained with Proxigenea3 instead of Wu-Palmer, except for the MSRpar test set.

	r	best	pg3	pg1	wp
MSRPar	60	0.734	0.417	0.429	**0.433**
MSRvid	58	0.880	**0.673**	0.612	0.583
SMTeur	**7**	0.567	**0.518**	0.495	0.486
OnWN	64	0.727	**0.553**	0.539	0.532
SMTnews	55	0.608	**0.369**	0.361	0.348
All	58	0.677	**0.520**	0.501	0.490

Table 4: Results obtained on each test set, grouped by conceptual similarity method. r indicates the ranking among all the participants teams.

In Table 4 we extrapolated the results for the composing methods and compared them with the result obtained after their combination. We used the pg3 configuration for the conceptual similarity measure. From these results, we can observe that MSRvid was a test set where the conceptual similarity alone would have resulted better than the combination of scores, while SMT-news was the test set where the CKPD measure obtained the best results in comparison to the result obtained by the conceptual similarity alone. It was quite surprising to observe such a good result for a method that does not take into account any information about the structure of the sentences, actually viewing them as "bags of con-

	Combined	pg3	CKPD
MSRPar	**0.417**	0.412	**0.417**
MSRvid	0.673	**0.777**	0.548
SMTeuroparl	**0.518**	0.486	0.467
OnWN	**0.553**	0.544	0.505
SMTnews	0.369	0.266	**0.408**

Table 5: Results obtained for each test set using only the conceptual similarity measure ($pg3$) and only the structural similarity measure ($CKPD$), compared to the result obtained by the complete system ($Combined$).

cepts". This is probably due to the fact that SMTnews is a corpus composed of automatically translated sentences, where structural similarity is an important clue for determining overall semantic similarity. On the other hand, MSRvid sentences are very short, and CKPD is in most cases unable to capture the semantic similarity.

5 Conclusions

The proposed method combined a measure of structural similarity and a measure of conceptual similarity based on WordNet. With the participation to this task, we were interested in studying the differences between different conceptual similarity measures and in determining whether they can be used to effectively measure the semantic similarity of text fragments. The obtained results showed that Proxigenea 3 allowed us to obtain the best results, indicating that under the test conditions and with WordNet as a resource it overperforms the Wu-Palmer measure. Further studies may be required in order to determine if these results can be generalised to other collections and in using different ontologies. We are also interested in comparing the method to the Lin concept similarity measure (Lin, 1998) which takes into account also the importance of the local root concept.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez. 2012. A pilot on semantic textual similarity. In *Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012), in conjunction with the First Joint Conference on Lexical and Computational Semantcis (*SEM 2012)*, Montreal, Quebec, Canada.

Thorsten Brants and Alex Franz. 2006. Web 1t 5-gram corpus version 1.1.

Davide Buscaldi, José Manuel Gómez, Paolo Rosso, and Emilio Sanchis. 2006. N-gram vs. keyword-based passage retrieval for question answering. In *CLEF*, pages 377–384.

Davide Buscaldi, Paolo Rosso, José Manuel Gómez, and Emilio Sanchis. 2009. Answering questions with an n-gram based passage retrieval engine. *Journal of Intelligent Information Systems (JIIS)*, 34(2):113–134.

Damien Dudognon, Gilles Hubert, and Bachelin Jhonn Victorino Ralalason. 2010. Proxigénéa : Une mesure de similarité conceptuelle. In *Proceedings of the Colloque Veille Stratégique Scientifique et Technologique (VSST 2010)*.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, pages 296–304, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

George A. Miller. 1995. Wordnet: a lexical database for english. *Commun. ACM*, 38(11):39–41, November.

Bachelin Ralalason. 2010. *Représentation multi-facette des documents pour leur accès sémantique*. Ph.D. thesis, Université Paul Sabatier, Toulouse, September. in French.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 173–180.

Zhibiao Wu and Martha Palmer. 1994. Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, ACL '94, pages 133–138, Stroudsburg, PA, USA. Association for Computational Linguistics.