Textual Entailment as Syntactic Graph Distance: a rule based and a SVM based approach

Maria Teresa Pazienza, Marco Pennacchiotti
DISP, University of Rome ”Tor Vergata”,
Viale del Politecnico 1, Roma, Italy,
{pazienza, pennacchiotti}@info.uniroma2.it

Fabio Massimo Zanzotto
DISCo, University of Milano-Bicocca,
Via Bicocca degli Arcimboldi 8, Milano, Italy,
zanzotto@disco.unimib.it

Abstract

In this paper we define a measure for textual entailment recognition based on the *graph matching theory* applied to syntactic graphs. We describe the experiments carried out to estimate measure's parameters with SVM and we report the results obtained on the Textual Entailment Challenge development and testing set.

1 Introduction

Graph distance/similarity measures are widely recognized to be powerful tools for *matching problems* in computer vision and pattern recognition applications (Bunke and Shearer, 1998). Objects to be matched (two images, patterns, etc.) are represented as graphs, turning the recognition problem into a graph matching task. As hypothesis (*H*) and text (*T*) may be seen as two syntactic graphs we can reduce the *textual entailment* (Dagan and Glickman, 2004) recognition problem to a graph similarity measure estimation even if textual entailment has particular properties: *a)* unlike the classical graph problems, is not symmetric; *b)* node similarity can not be reduced to the *label level* (e.g. token similarity); *c)* similarity should be estimated considering also linguistically motivated *graph transformations* (e.g., nominalization and passivization).

In principle, textual entailment is a transitive oriented relation holding in one of the following cases:

1. T *semantically subsumes* H (e.g., in *H:*[The cat eats the mouse] and *T:*[the cat devours the mouse], *eat* generalizes *devour*).
2. T *syntactically subsumes* H (e.g., in *H:*[The cat eats the mouse] and *T:*[the cat eats the mouse in the garden], T contains a specializing prepositional phrase).
3. T *directly implies* H (e.g., *H:*[The cat killed the mouse], *T:*[the cat devours the mouse]).

Taking this into account we define a measure $\mathcal{E}(\mathcal{XDG}_T, \mathcal{XDG}_H)$ for the entailment relation based on $\mathcal{XDG}_T$ and $\mathcal{XDG}_H$, i.e., the syntactic representation of the two sentences T and H. We work under two simplifying assumptions: H is supposed to be a sentence describing completely a fact in an assertive or negative way and H should be a simple S-V-O sentence. Our measure has to satisfy the following properties: (a) having a range between 0 and 1, assigning higher values to couples that are more likely in entailment relation, and a specific orientation, $\mathcal{E}(\mathcal{XDG}_T, \mathcal{XDG}_H) \neq \mathcal{E}(\mathcal{XDG}_H, \mathcal{XDG}_T)$; (b) the overlap between $\mathcal{XDG}_T$ and $\mathcal{XDG}_H$ has to describe if a subgraph of $\mathcal{XDG}_T$ implies the graph $\mathcal{XDG}_H$. Linguistic transformations (such as nominalization, passivization, and argument movement), as well as negation, must be also considered, as they can play a very important role.

2 Basic Definitions

For the syntactic representation we rely on the extended dependency graph (XDG) (Basili and Zanzotto, 2002). An $\mathcal{XDG} = (C, D)$ is basically a dependency graph whose nodes C are *constituents* and whose edges D are the *grammatical relations* among the constituents. Constituents are lexicalised syntactic trees with explicit *syntactic heads* and *potential semantic governors* (gov). Dependencies in D represent typed and ambiguous relations among a constituent, the $head$, and one of its $modifiers$. Ambiguity is represented using $plausibility$ (between 0 and 1).

Having the formalism it is possible to define how two structurally similar graphs are one subsumption of the other. Given $\mathcal{XDG}_H = (C_H, D_H)$ and $\mathcal{XDG}_T = (C_T, D_T)$, $\mathcal{XDG}_H$ is in a *isomorphic subsumption* relation with $\mathcal{XDG}_T$ ($\mathcal{XDG}_H \preceq \mathcal{XDG}_T$), if two bijective functions f_C and f_D exist respectively related to the constituents C and the dependencies D ($f_C : C_T \rightarrow C_H$ and $f_D : D_T \rightarrow D_H$). They describe the oriented relation of subsumption between nodes and edges of H and T. *Isomorphic subsumption* will capture textual entailment cases 1 and 3, that is, circumstances in which each node and edge of H has a correspondent in T, and vice-versa.

We denote with $\mathcal{XDG}'_T \subseteq \mathcal{XDG}_T$ a subgraph of $\mathcal{XDG}_T$. A *subgraph subsumption isomorphism* between $\mathcal{XDG}_H$ and $\mathcal{XDG}_T$, written as $\mathcal{XDG}_H \sqsubseteq \mathcal{XDG}_T$, holds if it exists $\mathcal{XDG}'_T \subseteq \mathcal{XDG}_T$ so that $\mathcal{XDG}_H \preceq \mathcal{XDG}'_T$. *Subgraph subsumption isomophism* correspond to textual entailment case 2, i.e, when there are nodes/edges of T not mapped in H, but all nodes/edges of H are mapped in T. Indeed, as the text entailment definition suggests, T can contain more information than H.

To tackle the problem of distortions in the syntactic and semantic interpretation, we can imagine an entailment measure based on the maximal subgraph $\mathcal{XDG}'_H$ of $\mathcal{XDG}_H$ (hereafter *maximal common subsumer subgraph, mcss*) that is in a *subgraph subsumption isomorphism* relation with $\mathcal{XDG}_T$, i.e. $\mathcal{XDG}'_H \sqsubseteq \mathcal{XDG}_T$. The measure should consider both the distance between $\mathcal{XDG}'_H$ and $\mathcal{XDG}_H$ and the generalisation steps necessary to draw the relation $\mathcal{XDG}'_H \sqsubseteq \mathcal{XDG}_T$.

3 A Rule-based Similarity Measure

To settle the measure the first problem is to extract $\mathcal{XDG}'_T$, i.e., the maximal subgraph of $\mathcal{XDG}_T$ that is in a subgraph isomorphism relation with $\mathcal{XDG}_H$, through the definition of the functions f_C (Sec.3.1) and f_D (Sec.3.2).

3.1 Node subsumption

To find the $mcss$ graph, we need to check that $\mathcal{XDG}'_H \subseteq \mathcal{XDG}_H$ and $\mathcal{XDG}'_T \subseteq \mathcal{XDG}_T$ are in the isomorphic relation $\mathcal{XDG}'_H \preceq \mathcal{XDG}'_T$. This is possible if the selection process of the subsets of the graphs nodes guarantees the possibility of defining the function f_C. This procedure should try to map each constituent of $\mathcal{XDG}_H$ to its most similar constituent in $\mathcal{XDG}_T$. If this is done, the bijective function f_C is derived by construction. The mapping process is based on the notion of *anchors*, defined as $a = (ch, ct, sm)$, holding an hypothesis and a text constituent (ch and ct), and the degree of *semantic similarity* $sm \in [0, 1]$ between the two. The set of anchors A for an entailment pair contains an anchor for each one of the hypothesis constituents having a correspondences in the text T. For example in the entailment pair of Fig. 1, f_C produces the mapping pairs *[The red cat - The carmine cat], [killed - devours], [the mouse - the mouse]*.

To determine the best set A, it is necessary to define the semantic similarity sm. If ch is a noun or a prepositional phrase, similarity is evaluated as:

$$sm(ch, ct) = \alpha * sim(gov_{ch}, gov_{ct}) + (1 - \alpha) * simsub(ch, ct)$$

where gov is the constituent governor, α is an empirically evaluated parameter used to weight the importance of the governor, and $simsub$ takes into account similarity among the all the other subcostituents of ch and ct. This latter is defined as:

$$simsub(ch, ct) = \frac{\sum_{sh \in S_{ch}} \max_{st \in S_{ct}} sim(sh, st)}{|S_{ch}|}$$

where S_{ch} and S_{ct} are the set of remaining simple constituents respectively of ch and ct. Finally, sim expresses the similarity among two simple constituents (set to 1 if simple constituents have the same surface or stem); otherwise, a semantic similarity weight β ∈(0,1) is assigned looking at possible WordNet relations (synonymy, entailment and generalization).

When ch is a verb phrase a different analysis occurs. In fact, a verb anchor can assume different *levels* of similarity, according to the semantic value of its modal. For example *must go-could go* should get a lower similarity than *must go-should go*. A verb phrase is thus composed by its governor gov and its

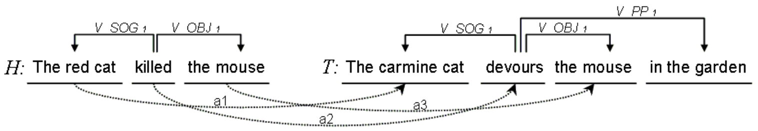

Figure 1: An example of entailment couple in the XDG formalism. Solid lines express grammatical relations D (with *type* and *plausibility)*; dotted lines express anchors a_i between H and T constituents.

modal constituents mod. The overall similarity is thus:

$$sm(ch, ct) = \gamma * sim(gov_{ch}, gov_{ct}) + (1 - \gamma) * dist(mod_{ch}, mod_{ct})$$

where $dist \in [0, 1]$ is empirically derived as the semantic distance between two modals (e.g., $must$ is nearer to $should$ than to $could$) (classified as generic auxiliaries, auxiliaries of possibility and auxiliaries of obligation). Specific cases of syntactic variations, such as active/passive alternation and nominalization are properly treated.

3.2 Edge subsumption

The anchor set A represents the nodes of the $mcss$. We will use f_D to derive the edges of the $mcss$. As XDG edges represent syntactic dependencies among constituents, for each anchor $a \in A$ the syntactic structure of ch and ct is checked, and a related *syntactic similarity* $ss(ch, ct) \in [0, 1]$ is evaluated. In order to obtain ss, it must be firstly defined the set of edges E_{ch} coming out from ch (in Figure 1 example, $E_{killed} = \{V\_sog, V\_obj\}$) and the corresponding set of connected nodes l_{ch} (e.g. $l_{killed} = \{[the\_red\_cat], [the\_mouse]\}$). In the same way, E_{ct} and l_{ct} are defined (e.g. $E_{devour} = \{V\_sog, V\_obj, V\_PP\}$ and $l_{ct} = \{[the\_carmine\_cat], [the\_mouse], [in\_the\_garden]\}$). A^L is defined as the set of anchors that contain overlapping linked constituents, that is, constituents linked with the same syntactic dependency to ch and ct respectively (for example, $a = ([the\_red\_cat], [the\_carmine\_cat], 0.95) \in A^L$, as the two constituents are both linked to *killed* and *devour* via a V_sog edge). ss is defined as:

$$ss(ch, ct) = \frac{\displaystyle\sum_{a \in A^L} sm_a}{|l_{ch}|}$$

Syntactic similarity, defined by f_D, will capture how much similar the syntactic structure accompanied to two constituents (i.e., the edges of the graphs) are, by considering both their syntactic properties (i.e., the common dependencies) and the semantic properties of the constituents to which they are linked (i.e., the similarity sm_a of the anchor of the linked constituents).

3.3 Graph Similarity Measure

Both semantic (sm) and syntactic (ss) similarity (derived respectively from f_C and f_D) must be taken into consideration to evaluate the overall graph similarity measure, as the former captures the notion of node subsumption, and the latter the notion of edge subsumption. For each pair (ch, ct) belonging to the set of anchors A a global similarity is evaluated as:

$$S(ch, ct) = \delta * sm(ch, ct) + (1 - \delta) * ss(ch, ct)$$

where δ is a manually tuned parameter. The overall graph similarity is thus estimated as the average similarity of the anchors $a \in A$ over total number of anchors:

$$\mathcal{E}(\mathcal{XDG}_T, \mathcal{XDG}_H) = \frac{\displaystyle\sum_{A} S(ch, ct)}{|A|}$$

It is possible to predict if an entailment relation holds between H and T couple, verifyng $\mathcal{E}(\mathcal{XDG}_T, \mathcal{XDG}_H)$ against a manually tuned threshold t.

4 Applying SVM to Evaluate Parameters

As clear from the previous sections, our measure depends on many parameters (α, β, γ, and δ). These parameters may be evaluated by a machine learning algorithm such as SVM. Due to the basic assumption that H should be a S-V-O sentence, feature spaces can be easily set. In order to comparatively evaluate the importance of different features we defined these feature sets: the features $\mathcal{G}$ related to the graph equivalence measure, i.e. $\mathcal{G}$ = { S_{sim}, S_{simsub}, S_{ss}, V_{sm}, V_{ss}, O_{sim}, O_{simsub}, O_{ss} }; the features $\mathcal{A}$ related to the number of commonly anchored dependencies within constituents to the graph equivalence

		D1	D3	D4	D5	D6
$\mathcal{L}$		51.16(±3.98)	-	-	-	-
$\mathcal{L}$,$\mathcal{T}$,$\mathcal{G}$	$\beta = 0.5$	-	55.28(±2.44)	56.14(±2.51)	56.40(±2.71)	56.72(±2.92)
$\mathcal{L}$,$\mathcal{T}$,$\mathcal{G}$	$\beta = 1$	-	56.37(±2.45)	57.14(±2.94)	57.37(±3.45)	57.12(±3.56)
$\mathcal{L}$,$\mathcal{T}$,$\mathcal{G}$,$\mathcal{A}$	$\beta = 1$	-	-	57.20(±3.01)	57.42(±3.36)	57.12(±3.38)

Table 1: Preliminary analysis on the develpment set using SVM

measure, i.e. $\mathcal{A}$ = { $|l_{ch}|$, $|l_{ct}|$}; $\mathcal{T}$ that are the features related to the textual entailment subtasks (CD, MT, etc.) Feature values are defined in Sec. 3. A final and less complex feature set is $\mathcal{L}$ that represents the percentage of H tokens and of H lemmas in common with T.

5 Results and preliminary evaluation

Before submitting the two runs of the two systems we estimated the parameters over the development set. For the first system referred as *rule-based* we set the parameters at the best value, i.e. $\alpha = 0.85$, $\gamma = 0.85$, and $\delta = 0.5$. Moreover, the *threshold* for predicting a true entailment relation has been set to $t = 0.65$. For the second system referred as *SVM-based* the experiments reported in Tab. 1 have been carried out. The table reports the accuracy of the classifier over the different parameterizations. Rows represent different feature spaces and when necessary the value of the parameter β. Columns represent different degree of the SVM type 1 polynomial kernel. For these preliminary experiments α and γ have been set respectively to 1 and 0.85. This preliminary setting of α, β, and γ seems to be in contrast with the aim of using SVM to estimate the measure parameters, but it is necessary to establish the initial set A of anchors over with values of the features may be computed. These experiments have been made in 3-fold cross validation repeated 10 times. The development set has been randomly divided 10 times (with a pseudo-random function and with 10 fix seeds). The results are reported as mean and standard deviation over 30 runs. All the feature spaces are better than the baseline feature space $\mathcal{L}$. We submitted the system that had the best result in this investigation.

Results over the competition test set are reported in Table 5. As expected by the preliminary analysis over the two development set results are not extremely high. Some trend has been somehow respected. The precision of the *SVM-based* is higher than the precision of the *rule-based* approach. However, it loses many points with respect to the preliminary evaluations, more than the expected standard deviation. The recall of the method is instead in line with the preliminary experiments. On this final set the accuracy of the *rule-based* approach has been higher of the *SVM-based* approach as happened on the development set. Further analysis are needed to better explain these results.

measure	*rule-based*	*SVM-based*
cws	0.5574	0.5591
accuracy	0.5245	0.5182
precision	0.5265	0.5532
recall	0.4975	0.1950
f	0.5116	0.2884

	rule-based		*SVM-based*	
TASK	cws	$accuracy$	cws	$accuracy$
CD	0.8381	0.7651	0.7174	0.6443
IE	0.4559	0.4667	0.4632	0.4917
MT	0.5914	0.5210	0.4961	0.4790
QA	0.4408	0.3953	0.4571	0.4574
RC	0.5167	0.4857	0.5898	0.5214
PP	0.5583	0.5400	0.5768	0.5000
IR	0.4405	0.4444	0.4882	0.4889

Table 2: Competition results

References

Roberto Basili and Fabio Massimo Zanzotto. 2002. Parsing engineering and empirical robustness. *Natural Language Engineering*, 8/2-3.

Horst Bunke and Kim Shearer. 1998. A graph distance metric based on the maximal common subgraph. *Pattern Recogn. Lett.*, 19(3-4):255–259.

Ido Dagan and Oren Glickman. 2004. Probabilistic textual entailment: Generic applied modeling of language variability. In *Learning Methods for Text Understanding and Mining*, Grenoble, France.