Towards a Probabilistic Model for Lexical Entailment

Eyal Shnarch
Computer Science Department
Bar-Ilan University
Ramat-Gan, Israel
shey@cs.biu.ac.il

Jacob Goldberger
School of Engineering
Bar-Ilan University
Ramat-Gan, Israel
goldbej@eng.biu.ac.il

Ido Dagan
Computer Science Department
Bar-Ilan University
Ramat-Gan, Israel
dagan@cs.biu.ac.il

Abstract

While modeling entailment at the lexical-level is a prominent task, addressed by most textual entailment systems, it has been approached mostly by heuristic methods, neglecting some of its important aspects. We present a probabilistic approach for this task which covers aspects such as differentiating various resources by their reliability levels, considering the length of the entailed sentence, the number of its covered terms and the existence of multiple evidence for the entailment of a term. The impact of our model components is validated by evaluations, which also show that its performance is in line with the best published entailment systems.

1 Introduction

Textual Entailment was proposed as a generic paradigm for applied semantic inference (Dagan et al., 2006). Given two textual fragments, termed *hypothesis* (H) and *text* (T), the text is said to textually entail the hypothesis ($T \rightarrow H$) if a person reading the text can infer the meaning of the hypothesis. Since it was first introduced, the six rounds of the Recognizing Textual Entailment (RTE) challenges[1] have become a standard benchmark for entailment systems.

Entailment systems apply various techniques to tackle this task, including logical inference (Tatu and Moldovan, 2007; MacCartney and Manning, 2007), semantic analysis (Burchardt et al., 2007) and syntactic parsing (Bar-Haim et al., 2008; Wang et al., 2009). Inference at these levels usually requires substantial processing and resources, aiming at high performance. Nevertheless, simple *lexical* level entailment systems pose strong baselines which most complex entailment systems did not outperform (Mirkin et al., 2009a; Majumdar and Bhattacharyya, 2010). Additionally, within a complex system, lexical entailment modeling is one of the most effective component. Finally, the simpler lexical approach can be used in cases where complex systems cannot be used, e.g. when there is no parser for a targeted language.

For these reasons lexical entailment systems are widely used. They derive *sentence-level* entailment decision base on *lexical-level* entailment evidence. Typically, this is done by quantifying the degree of lexical coverage of the hypothesis terms by the text terms (where a term may be multi-word). A hypothesis term is covered by a text term if either they are identical (possibly at the stem or lemma level) or there is a lexical entailment *rule* suggesting the entailment of the former by the latter. Such rules are derived from lexical semantic resources, such as WordNet (Fellbaum, 1998), which capture lexical entailment relations.

Common heuristics for quantifying the degree of coverage are setting a threshold on the percentage of coverage of H's terms (Majumdar and Bhattacharyya, 2010), counting the absolute number of uncovered terms (Clark and Harrison, 2010), or applying an Information Retrieval-style vector space similarity score (MacKinlay and Baldwin, 2009). Other works (Corley and Mihalcea, 2005; Zanzotto and Moschitti, 2006) have applied heuristic formu-

[1] http://www.nist.gov/tac/

las to estimate the similarity between text fragments based on a similarity function between their terms.

The above mentioned methods do not capture several important aspects of entailment. Such aspects include the varying reliability levels of entailment resources and the impact of rule chaining and multiple evidence on entailment likelihood. An additional observation from these and other systems is that their performance improves only moderately when utilizing lexical-semantic resources[2].

We believe that the textual entailment field would benefit from more principled models for various entailment phenomena. In this work we formulate a concrete generative probabilistic modeling framework that captures the basic aspects of lexical entailment. A first step in this direction was proposed in Shnarch et al. (2011) (a short paper), where we presented a base model with a somewhat complicated and difficult to estimate extension to handle coverage. This paper extends that work to a more mature model with new extensions.

We first consider the "logical" structure of lexical entailment reasoning and then interpret it in probabilistic terms. Over this base model we suggest several extensions whose significance is then assessed by our evaluations. Learning the parameters of a lexical model poses a challenge since there are no lexical-level entailment annotations. We do, however, have sentence-level annotations available for the RTE data sets. To bridge this gap, we formulate an instance of the EM algorithm (Dempster et al., 1977) to estimate hidden lexical-level entailment parameters from sentence-level annotations.

Overall, we suggest that the main contribution of this paper is in presenting a probabilistic model for lexical entailment. Such a model can better integrate entailment indicators and has the advantage of being able to utilize well-founded probabilistic methods such as the EM algorithm. Our model's performance is in line with the best entailment systems, while opening up directions for future improvements.

2 Background

We next review several entailment systems, mostly those that work at the lexical level and in particular those with which we compare our results on the RTE data sets.

The 5^{th} Recognizing Textual Entailment challenge (RTE-5) introduced a new pilot task (Bentivogli et al., 2009) which became the main task in RTE-6 (Bentivogli et al., 2010). In this task the goal is to find all sentences that entail each hypothesis in a given document cluster. This task's data sets reflect a natural distribution of entailments in a corpus and demonstrate a more realistic scenario than the earlier RTE challenges.

As reviewed in the following paragraphs there are several characteristic in common to most entailment systems: (1) lexical resources have a minimal impact on their performance, (2) they heuristically utilize lexical resources, and (3) there is no principled method for making the final entailment decision.

The best performing system of RTE-5 was presented by Mirkin et. al (2009a). It applies supervised classifiers over a parse tree representations to identify entailment. They reported that utilizing lexical resources only slightly improved their performance.

MacKinlay and Baldwin (2009) presented the best lexical-level system at RTE-5. They use a vector space method to measure the lexical overlap between the text and the hypothesis. Since usually texts of RTE are longer than their corresponding hypotheses, the standard cosine similarity score came out lower than expected. To overcome this problem they suggested a simple ad-hoc variant of the cosine similarity score which removed from the text all terms which did not appear in the corresponding hypothesis. While this heuristic improved performance considerably, they reported a decrease in performance when utilizing synonym and derivation relations from WordNet.

On the RTE-6 data set, the syntactic-based system of Jia et. al (2010) achieved the best results, only slightly higher than the lexical-level system of (Majumdar and Bhattacharyya, 2010). The latter utilized several resources for matching hypothesis terms with text terms: WordNet, VerbOcean (Chklovski and Pantel, 2004), utilizing two of its relations, as well as an acronym database, number matching module, co-reference resolution and named entity recognition tools. Their final entailment decision was based on a threshold over the

[2]See ablation tests reports in http://aclweb.org/aclwiki/ index.php?title=RTE_Knowledge_Resources#Ablation_Tests

number of matched hypothesis terms. They found out that hypotheses of different length require different thresholds.

While the above systems measure the number of hypothesis terms matched by the text, Clark and Harrison (2010) based their entailment decision on the number of *mismatched* hypothesis terms. They utilized both WordNet and the DIRT paraphrase database (Lin and Pantel, 2001). With WordNet, they used one set of relations to identify the concept of a term while another set of relations was used to identify entailment between concepts. Their results were inconclusive about the overall effect of DIRT while WordNet produced a net benefit in most configurations. They have noticed that setting a global threshold for the entailment decision, decreased performance for some topics of the RTE-6 data set. Therefore, they tuned a varying threshold for each topic based on an idiosyncracy of the data, by which the total number of entailments per topic is approximately a constant.

Glickman et al. (2005) presented a simple model that recasted the lexical entailment task as a variant of text classification and estimated entailment probabilities solely from co-occurrence statistics. Their model did not utilize any lexical resources.

In contrary to these systems, our model shows improvement when utilizing high quality resources such as WordNet and the CatVar (Categorial Variation) database (Habash and Dorr, 2003). As Majumdar and Bhattacharyya (2010), our model considers the impact of hypothesis length, however it does not require the tuning of a unique threshold for each length. Finally, most of the above systems do not differentiate between the various lexical resources they use, even though it is known that resources reliability vary considerably (Mirkin et al., 2009b). Our probabilistic model, on the other hand, learns a unique reliability parameter for each resource it utilizes. As mentioned above, this work extends the base model in (Shnarch et al., 2011), which is described in the next section.

3 A Probabilistic Model

We aim at obtaining a probabilistic score for the likelihood that the hypothesis terms are entailed by the terms of the text. There are several prominent aspects of entailment, mostly neglected by previous lexical methods, which our model aims to capture: (1) the reliability variability of different lexical resources; (2) the effect of the length of transitive rule application chain on the likelihood of its validity; and (3) addressing cases of multiple entailment evidence when entailing a term.

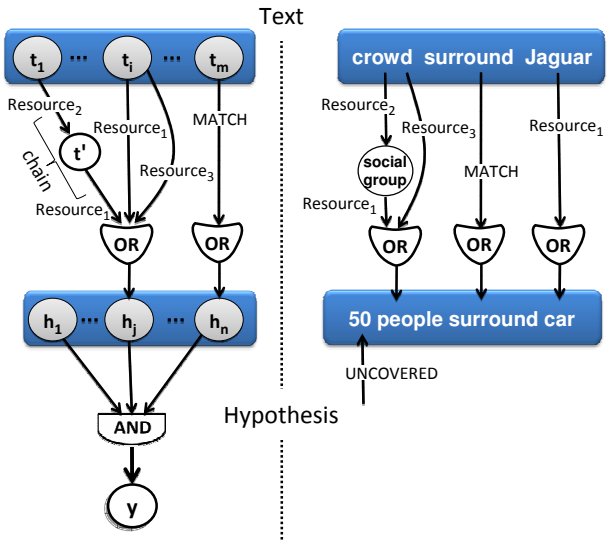

Figure 1: **Left:** the base model of entailing a hypothesis from a text; **Right:** a concrete example for it (stop-words removed). Edges in the upper part of the diagram represent entailment rules. Rules compose chains through AND gates (omitted for visual clarity). Chains are gathered by OR gates to entail terms, and the final entailment decision y is the result of their AND gate.

3.1 The Base Model

Our base model follows the one presented in (Shnarch et al., 2011), which is described here in detail to make the current paper self contained.

3.1.1 Entailment generation process

We first specify the process by which a decision of lexical entailment between T and H using knowledge resources should be determined, as illustrated in Figure 1 (a general description on the left and a concrete example on the right). There are two ways by which a term $h \in H$ is entailed by a term $t \in T$. A direct MATCH is the case in which t and h are identical terms (possibly at the stem or lemma level). Alternatively, lexical entailment can be established based on knowledge of entailing lexical-

semantic relations, such as synonyms, hypernyms and morphological derivations, available in lexical resources. These relations provide *lexical entailment rule*s, e.g. *Jaguar* → *car*. We denote the resource which provided the rule r by $R(r)$.

It should be noticed at this point that such rules specify a lexical entailment relation that might hold for *some* (T, H) pairs but not necessarily for all pairs, e.g. the rule *Jaguar* → *car* does not hold in the wildlife context. Thus, the application of an available rule to infer lexical entailment in a given (T, H) pair might be either valid or invalid. We note here the difference between *covering* a term and *entailing* it. A term is covered when the available resources suggest its entailment. However, since a rule application may be invalid for the particular (T, H) context, a term is entailed only if there is a valid rule application from T to it.

Entailment is a transitive relation, therefore rules may compose transitive *chain*s that connect t to h via intermediate term(s) t' (e.g. *crowd* → *social group* → *people*). For a chain to be valid for the current (T, H) pair, *all* its composing rule applications should be valid for this pair. This corresponds to a logical AND gate (omitted in Figure 1 for visual clarity) which takes as input the validity values (1/0) of the individual rule applications.

Next, multiple chains may connect t to h (as for t_i and h_j in Figure 1) or connect several terms in T to h (as t_1 and t_i are indicating the entailment of h_j in Figure 1), thus providing multiple evidence for h's entailment. For a term h to be entailed by T it is enough that *at least one* of the chains from T to h would be valid. This condition is realized in the model by an OR gate. Finally, for T to *lexically* entail H it is usually assumed that *every* $h \in H$ should be entailed by T (Glickman et al., 2006). Therefore, the final decision follows an AND gate combining the entailment decisions for all hypothesis terms. Thus, the 1-bit outcome of this gate y corresponds to the sentence-level entailment status.

3.1.2 Probabilistic Setting

When assessing entailment for (T, H) pair, we do not know for sure which rule applications are valid. Taking a probabilistic perspective, we assume a parameter θ_R for each resource R, denoting its reliability, i.e. the prior probability that applying a rule from R for an arbitrary (T, H) pair corresponds to valid entailment[3]. Under this perspective, direct MATCHs are considered as rules coming from a special "resource", for which θ_{MATCH} is expected to be close to 1. Additionally, there could be a term h which is not covered by any of the resources at hand, whose coverage is inevitably incomplete. We assume that each such h is covered by a single rule coming from a dummy resource called UNCOVERED, while expecting $\theta_{\text{UNCOVERED}}$ to be relatively small. Based on the θ_R values we can now estimate, for each entailment inference step in Figure 1, the probability that this step is valid (the corresponding bit is 1).

Equations (1) - (3) correspond to the three steps in calculating the probability for entailing a hypothesis.

$$p(t \xrightarrow{c} h) = \prod_{r \in c} p(L \xrightarrow{r} R) = \prod_{r \in c} \theta_{R(r)} \quad (1)$$

$$p(T \rightarrow h) = 1 - p(T \nrightarrow h) = 1 - \prod_{c \in C(h)} [1 - p(t \xrightarrow{c} h)] \quad (2)$$

$$p(T \rightarrow H) = \prod_{h \in H} p(T \rightarrow h) \quad (3)$$

First, Eq. (1) specifies the probability of a particular chain c, connecting a text term t to a hypothesis term h, to correspond to a valid entailment between t and h. This event is denoted by $t \xrightarrow{c} h$ and its probability is the joint probability that the applications of all rules $r \in c$ are valid. Note that every rule r in a chain c connects two terms, its left-hand-side *L* and its right-hand-side *R*. The left-hand-side of the first rule in c is $t \in T$ and the right-hand-side of the last rule in it is $h \in H$. Let us denote the event of a valid rule application by $L \xrightarrow{r} R$. Since a-priori a rule r is valid with probability $\theta_{R(r)}$, and assuming independence of all $r \in c$, we obtain Eq. (1).

Next, Eq. (2) utilizes Eq. (1) to specify the probability that T entails h (at least by one chain). Let $C(h)$ denote the set of chains which suggest the entailment of h. The requested probability is equal to 1 minus the probability of the complement event, that is, T does not entail h by any chain. The latter probability is the product of probabilities that all

[3]Modeling a conditional probability for the validity of r, which considers contextual aspects of r's validity in the current (T, H) context, is beyond the scope of this paper (see discussion in Section 6)

chains $c \in C(h)$ are not valid (again assuming independence of chains).

Finally, Eq. (3) gives the probability that T entails all of H ($T \rightarrow H$), assuming independence of H's terms. This is the probability that every $h \in H$ is entailed by T, as specified by Eq. (2).

Altogether, these formulas fall out of the standard probabilistic estimate for the output of AND and OR gates when assuming independence amongst their input bits.

As can be seen, the base model distinguishes varying resource reliabilities, as captured by θ_R, decreases entailment probability as rule chain grows, having more elements in the product of Eq. (1), and increases it when entailment of a term is supported by multiple chains with more inputs to the OR gate. Next we describe two extensions for this base model which address additional important phenomena of lexical entailment.

3.2 Relaxing the AND Gate

Based on term-level decisions for the entailment of each $h \in H$, the model has to produce a sentence-level decision of $T \rightarrow H$. In the model described so far, for T to entail H it must entail *all* its terms. This demand is realized by the AND gate at the bottom of Figure 1. In practice, this demand is too strict, and we would like to leave some option for entailing H even if not every $h \in H$ is entailed. Thus, it is desired to relax this strict demand enforced by the AND gate in the model.

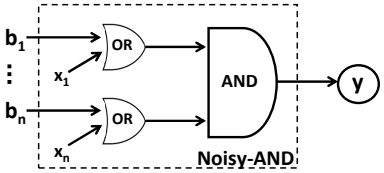

Figure 2: A noisy-AND gate

The Noisy-AND model (Pearl, 1988), depicted in Figure 2, is a soft probabilistic version of the AND gate, which is often used to describe the interaction between causes and their common effect. In this variation, each one of the binary inputs $b_1, ..., b_n$ of the AND gate is first joined with a "noise" bit x_i by an OR gate. Each "noise" bit is 1 with probability p, which is the parameter of the gate. The output bit y is defined as:

$$y = (b_1 \vee x_1) \wedge (b_2 \vee x_2) \wedge \cdots \wedge (b_n \vee x_n)$$

and the conditional probability for it to be 1 is:

$$p(y = 1|b_1, ..., b_n, n) = \prod_{i=1}^{n} p^{(1-b_i)} = p^{(n-\sum_i b_i)}$$

If all the binary input values are 1, the output is deterministically 1. Otherwise, the probability that the output is 1 is proportional to the number of ones in the input, where the distribution depends on the parameter p. In case $p = 0$ the model reduces to the regular AND.

In our model we replace the final strict AND with a noisy-AND, thus increasing the probability of T to entail H, to account for the fact that sometimes H might be entailed from T even though some $h \in H$ is not directly entailed.

The input size n for the noisy-AND is the length of the hypotheses and therefore it varies from H to H. Had we used the same model parameter p for all lengths, the probability to output 1 would have depended solely on the number of 0 bits in the input without considering the number of ones. For example, the probability to entail a hypothesis with 10 terms given that 8 of them are entailed by T (and 2 are not) is p^2. The same probability is obtained for a hypothesis of length 3 with a single entailed term. We, however, expect the former to have a higher probability since a larger portion of its terms is entailed by T.

There are many ways to incorporate the length of a hypothesis into the noisy-AND model in order to normalize its parameter. The approach we take is defining a separate parameter p_n for each hypothesis length n such that $p_n = \theta_{NA}^{\frac{1}{n}}$, where θ_{NA} becomes the underlying parameter value of the noisy-AND, i.e.

$$p(y = 1|b_1, ..., b_n, n) = p_n^{(n-\sum b_i)} = \theta_{NA}^{\frac{n-\sum b_i}{n}}$$

This way, if non of the hypothesis terms is entailed, the probability for its entailment is θ_{NA}, independent of its length:

$$p(y = 1|0, 0, ..., 0, n) = p_n^n = \theta_{NA}$$

As can be seen from Figure 1, replacing the final AND gate by a noisy-AND gate is equivalent to adding an additional chain to the OR gate of each hypothesis term. Therefore we update Eq. (2) to:

$$\begin{aligned} p(T \rightarrow h) =& 1 - p(T \nrightarrow h) \\ =& 1 - [(1 - \theta_{NA}^{\frac{1}{n}}) \cdot \prod_{c \in C(h)} [1 - p(t \xrightarrow{c} h)]] \end{aligned} \quad (2^*)$$

In the length-normalized noisy-AND model the value of the parameter p becomes higher for longer hypotheses. This increases the probability to entail such hypotheses, compensating for the lower probability to strictly entail all of their terms.

3.3 Considering Coverage Level

The second extension of the base model follows our observation that the prior validity likelihood for a rule application, increases as more of H's terms are covered by the available resources. In other words, if we have a hypothesis H_1 with k covered terms and a hypothesis H_2 in which only $j < k$ terms are covered, then an arbitrary rule application for H_1 is more likely to be valid than an arbitrary rule application for H_2.

We chose to model this phenomenon by normalizing the reliability θ_R of each resource according to the number of covered terms in H. The normalization is done in a similar manner to the length-normalized noisy-AND described above, obtaining a modified version of Eq. (1):

$$p(t \xrightarrow{c} h) = \prod_{r \in c} \theta_{R(r)}^{\frac{1}{\#covered}} \quad (1^*)$$

As a results, the larger the number of covered terms is, the larger θ_R values our model uses and, in total, the entailment probability increases.

To sum up, we have presented the base model, providing a probabilistic estimate for the entailment status in our generation process specified in 3.1. Two extensions were then suggested: one that relaxes the strict AND gate and normalizes this relaxation by the length of the hypothesis; the second extension adjusts the validity of rule applications as a function of the number of the hypothesis covered terms. Overall, our *full model* combines both extensions over the base probabilistic model.

4 Parameter Estimation

The difficulty in estimating the θ_R values from training data arises because these are term-level parameters while the RTE-training entailment annotation is given for the sentence-level, each (T, H) pair in the training is annotated as either entailing or not. Therefore, we use an instance of the EM algorithm (Dempster et al., 1977) to estimate these hidden parameters.

4.1 E-Step

In the E-step, for each application of a rule r in a chain c for $h \in H$ in a training pair (T, H), we compute $w_{hcr}(T, H)$, the posterior probability that the rule application was valid given the training annotation:

$$w_{hcr}(T, H) = \begin{cases} p(L \xrightarrow{r} R | T \rightarrow H) & \text{if } T \rightarrow H \\ p(L \xrightarrow{r} R | T \nrightarrow H) & \text{if } T \nrightarrow H \end{cases} \quad (4)$$

where the two cases refer to whether the training pair is annotated as entailing or non-entailing. For simplicity, we write w_{hcr} when the (T, H) context is clear.

The E-step can be efficiently computed using dynamic programming as follows; For each training pair (T, H) we first compute the probability $p(T \rightarrow H)$ and keep all the intermediate computations (Eq. (1)- (3)). Then, the two cases of Eq. (4), elaborated next, can be computed from these expressions. For computing Eq. (4) in the case that $T \rightarrow H$ we have:

$$p(L \xrightarrow{r} R | T \rightarrow H) = p(L \xrightarrow{r} R | T \rightarrow h) = \frac{p(T \rightarrow h | L \xrightarrow{r} R) p(L \xrightarrow{r} R)}{p(T \rightarrow h)}$$

The first equality holds since when T entails H every $h \in H$ is entailed by it. Then we apply Bayes' rule. We have already computed the denominator (Eq. (2)), $p(L \xrightarrow{r} R) \equiv \theta_{R(r)}$ and it can be shown[4] that:

$$p(T \rightarrow h | L \xrightarrow{r} R) = 1 - \frac{p(T \nrightarrow h)}{1 - p(t \xrightarrow{c} h)} \cdot (1 - \frac{p(t \xrightarrow{c} h)}{\theta_{R(r)}}) \quad (5)$$

[4]The first and second denominators reduce elements from the products in Eq. 2 and Eq. 1 correspondingly

where c is the chain which contains the rule r.

For computing Eq. (4), in the second case, that $T \nrightarrow H$, we have:

$$p(L \xrightarrow{r} R | T \nrightarrow H) = \frac{p(T \nrightarrow H | L \xrightarrow{r} R) p(L \xrightarrow{r} R)}{p(T \nrightarrow H)}$$

In analogy to Eq. (5) it can be shown that

$$p(T \nrightarrow H | L \xrightarrow{r} R) = 1 - \frac{p(T \rightarrow H)}{p(T \rightarrow h)} \cdot p(T \rightarrow h | L \xrightarrow{r} R) \quad (6)$$

while the expression for $p(T \rightarrow h | L \xrightarrow{r} R)$ appears in Eq. (5).

This efficient computation scheme is an instance of the belief-propagation algorithm (Pearl, 1988) applied to the entailment process, which is a loop-free directed graph (Bayesian network).

4.2 M-Step

In the M-step we need to maximize the EM auxiliary function $Q(\theta)$ where θ is the set of all resources reliability values. Applying the derivation of the auxiliary function to our model (first without the extensions) we obtain:

$$Q(\theta) = \sum_{T,H} \sum_{h \in H} \sum_{c \in C(h)} \sum_{r \in c} (w_{hcr} \log \theta_{R(r)} + (1 - w_{hcr}) \log(1 - \theta_{R(r)}))$$

We next denote by n_R the total number of applications of rules from resource R in the training data. We can maximize $Q(\theta)$ for each R separately to obtain the M-step parameter-updating formula:

$$\theta_R = \frac{1}{n_R} \sum_{T,H} \sum_{h \in H} \sum_{c \in C(h)} \sum_{r \in c | R(r) = R} w_{hcr} \quad (7)$$

The updated parameter value averages the posterior probability that rules from resource R have been validly applied, across all its utilizations in the training data.

4.3 EM for the Extended Model

In case we normalize the noisy-AND parameter by the hypothesis length, for each length we use a different parameter value for the noisy-AND and we cannot simply merge the information from all the training pairs (T, H). To find the optimal parameter value for θ_{NA}, we need to maximize the following expression (the derivation of the auxiliary function to the hypothesis-length-normalized noisy-AND "resource"):

$$Q(\theta_{NA}) = \sum_{T,H} \sum_{h \in H} (w_{hNA} \log(\theta_{NA}^{\frac{1}{n}}) + (1 - w_{hNA}) \log(1 - \theta_{NA}^{\frac{1}{n}})) \quad (8)$$

where n is the length of H, θ_{NA} is the parameter value of the noisy-AND model and w_{hNA} is the posterior probability that the noisy-AND was used to validly entail the term h[5], i.e.

$$w_{hNA}(T, H) = \begin{cases} p(T \xrightarrow{NA} h | T \rightarrow H) & \text{if } T \rightarrow H \\ p(T \xrightarrow{NA} h | T \nrightarrow H) & \text{if } T \nrightarrow H \end{cases}$$

The two cases of the above equation are similar to Eq. (4) and can be efficiently computed in analogy to Eq. (5) and Eq. (6).

There is no close-form expression for the parameter value θ_{NA} that maximizes expression (8). Since $\theta_{NA} \in [0, 1]$ is a scalar parameter, we can find θ_{NA} value that maximizes $Q(\theta_{NA})$ using an exhaustive grid search on the interval $[0, 1]$, in each iteration of the M-step. Alternatively, for an iterative procedure to maximize expression (8), see Appendix A.

In the same manner we address the normalization of the reliability θ_R of each resources R by the number of H's covered terms. Expression (8) becomes:

$$Q(\theta_R) = \sum_{T,H} \sum_{h \in H} \sum_{c \in C(h)} \sum_{r \in c | R(r) = R} (w_{hcr} \log(\theta_R^{cov}) + (1 - w_{hcr}) \log(1 - \theta_R^{cov}))$$

were $\frac{1}{cov}$ is the number of H terms which are covered. We can find the θ_R that maximizes this equation in one of the methods described above.

5 Evaluation and Results

For our evaluation we use the RTE-5 pilot task and the RTE-6 main task data sets described in Section 2. In our system, sentences are tokenized and stripped of stop words and terms are tagged for part-of-speech and lemmatized. We utilized two lexical resources, WordNet (Fellbaum, 1998) and CatVar

[5] In contrary to Eq. 4, here there is no specific $t \in T$ that entails h, therefore we write $T \xrightarrow{NA} h$

(Habash and Dorr, 2003). From WordNet we took as entailment rules synonyms, derivations, hyponyms and meronyms of the first senses of T and H terms. CatVar is a database of clusters of uninflected words (lexemes) and their categorial (i.e. part-of-speech) variants (e.g. announce (verb), announcer and announcement(noun) and announced (adjective)). We deduce an entailment relation between any two lexemes in the same cluster. Model's parameters were estimated from the development set, taken as training. Based on these parameters, the entailment probability was estimated for each pair (T, H) in the test set, and the classification threshold was tuned by classification over the development set.

We next present our evaluation results. First we investigate the impact of utilizing lexical resources and of chaining rules. In section 5.2 we evaluate the contribution of each extension of the base model and in Section 5.3 we compare our performance to that of state-of-the-art entailment systems.

5.1 Resources and Rule-Chaining Impact

As mentioned in Section 2, in the RTE data sets it is hard to show more than a moderate improvement when utilizing lexical resources. Our analysis ascribes this fact to the relatively small amount of rule applications in both data sets. For instance, in RTE-6 there are 10 times more direct matches of identical terms than WordNet and CatVar rule applications combined, while in RTE-5 this ratio is 6. As a result the impact of rule applications can be easily shadowed by the large amount of direct matches.

Table 1 presents the performance of our (full) model when utilizing *no resources* at all, *WordNet*, *CatVar* and both, with chains of a single step. We also considered rule chains of length up to 4 and present here the results of 2 chaining steps with *WordNet-2* and *(WordNet+CatVar)-2*.

Overall, despite the low level of rule applications, we see that incorporating lexical resources in our model significantly[6] and quite consistently improves performance over using no resources at all. Naturally, the optimal combination of resources may vary somewhat across the data sets.

In RTE-6 *WordNet-2* significantly improved performance over the single-stepped WordNet. However, mostly chaining did not help, suggesting the need for future work to improve chain modeling in our framework.

Model	$F_1\%$	
	RTE-5	RTE-6
no resources	41.6	44.9
WordNet	45.8	44.6
WordNet-2	45.7	45.5
CatVar	46.9	**45.6**
WordNet + CatVar	**48.3**	**45.6**
(WordNet + CatVar)-2	47.1	44.0

Table 1: Evaluation of the impact of resources and chaining.

5.2 Model Components impact

We next assess the impact of each of our proposed extensions to the base probabilistic model. To that end, we incorporate *WordNet+CatVar* (our best configuration above) as resources for the *base model* (Section 3.1) and compare it with the *noisy-AND* extension (Eq. (2*)), the *covered-norm* extension which normalizes the resource reliability parameter by the number of covered terms (Eq. (1*)) and the *full model* which combines both extensions. Table 2 presents the results: both *noisy-AND* and *covered-norm* extensions significantly increase F_1 over the base model (by 4.5-8.4 points). This scale of improvement was observed with all resources and chain-length combinations. In both data sets, the combination of *noisy-AND* and *covered-norm* extensions in the full model significantly outperforms each of them separately[7], showing their complementary nature. We also observed that applying noisy-AND without the hypothesis length normalization hardly improved performance over the base model, emphasising the importance of considering hypothesis length. Overall, we can see that both base model extensions improve performance.

Table 3 illustrates a set of maximum likelihood parameters that yielded our best results (*full model*). The parameter value indicates the learnt reliability of the corresponding resource.

[6] All significant results in this section are according to McNemar's test with $p < 0.01$ unless stated otherwise

[7] With the following exception: in RTE-5 the full model is better than the *noisy-AND* extension with significance of only $p = 0.06$

Model	$F_1\%$	
	RTE-5	RTE-6
base model	36.2	38.5
noisy-AND	44.6	43.1
covered-norm	42.8	44.7
full model	**48.3**	**45.6**

Table 2: Impact of model components.

θ_{MATCH}	θ_{WORDNET}	θ_{CATVAR}	$\theta_{\text{UNCOVERED}}$	θ_{NA}
0.80	0.70	0.65	0.17	0.05

Table 3: A parameter set of the *full model* which maximizes the likelihood of the training set.

5.3 Comparison to Prior Art

Finally, in Table 4, we put these results in the context of the best published results on the RTE task. We compare our model to the *average* of the best runs of all systems, the *best* and *second best* performing lexical systems and the *best full system* of each challenge. For both data sets our model is situated high above the average system. For the RTE-6 data set, our model's performance is third best with Majumdar and Bhattacharyya (2010) being the only lexical-level system which outperforms it. However, their system utilized additional processing that we did not, such as named entity recognition and coreference resolution[8]. On the RTE-5 data set our model outperforms any other published result.

Model	$F_1\%$	
	RTE-5	RTE-6
full model	48.3	45.6
avg. of all systems	30.5	33.8
2^{nd} best lexical system	40.3^a	44.0^b
best lexical system	44.4^c	47.6^d
best full system	45.6^c	48.0^e

Table 4: Comparison to RTE-5 and RTE-6 best entailment systems: (a)(MacKinlay and Baldwin, 2009), (b)(Clark and Harrison, 2010), (c)(Mirkin et al., 2009a)(2 submitted runs), (d)(Majumdar and Bhattacharyya, 2010) and (e)(Jia et al., 2010).

[8]We note that the submitted run which outperformed our result utilized a threshold which was a manual modification of the threshold obtained systematically in another run. The latter run achieved F_1 of 42.4% which is below our result.

We conclude that our probabilistic model demonstrates quality results which are also consistent, without applying heuristic methods of the kinds reviewed in Section 2

6 Conclusions and Future Work

We presented, a probabilistic model for lexical entailment whose innovations are in (1) considering each lexical resource separately by associating an individual reliability value for it, (2) considering the existence of multiple evidence for term entailment and its impact on entailment assessment, (3) setting forth a probabilistic method to relax the strict demand that all hypothesis terms must be entailed, and (4) taking account of the number of covered terms in modeling entailment reliability.

We addressed the impact of the various components of our model and showed that its performance is in line with the best state-of-the-art inference systems. Future work is still needed to reflect the impact of transitivity. We consider replacing the AND gate on the rules of a chain by a noisy-AND, to relax its strict demand that all its input rules must be valid. Additionally, we would like to integrate Contextual Preferences (Szpektor et al., 2008) and other works on Selectional Preference (Erk and Pado, 2010) to verify the validity of the application of a rule in a specific (T, H) context. We also intend to explore the contribution of our model within a complex system that integrates multiple levels of inference as well as its contribution for other applications, such as Passage Retrieval.

References

Roy Bar-Haim, Jonathan Berant, Ido Dagan, Iddo Greental, Shachar Mirkin, Eyal Shnarch, and Idan Szpektor. 2008. Efficient semantic deduction and approximate matching over compact parse forests. In *Proc. of TAC*.

Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. 2009. The fifth PASCAL recognizing textual entailment challenge. In *Proc. of TAC*.

Luisa Bentivogli, Peter Clark, Ido Dagan, Hoa Trang Dang, and Danilo Giampiccolo. 2010. The sixth PASCAL recognizing textual entailment challenge. In *Proc. of TAC*.

Aljoscha Burchardt, Nils Reiter, Stefan Thater, and Anette Frank. 2007. A semantic approach to textual

entailment: System evaluation and task analysis. In *Proc. of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*.

Timothy Chklovski and Patrick Pantel. 2004. Verbocean: Mining the web for fine-grained semantic verb relations. In *Proc. of EMNLP*.

Peter Clark and Phil Harrison. 2010. BLUE-Lite: a knowledge-based lexical entailment system for RTE6. In *Proc. of TAC*.

Courtney Corley and Rada Mihalcea. 2005. Measuring the semantic similarity of texts. In *Proc. of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Lecture Notes in Computer Science*, volume 3944, pages 177–190.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society, series [B]*, 39(1):1–38.

Katrin Erk and Sebastian Pado. 2010. Exemplar-based models for word meaning in context. In *Proc. of the ACL*.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press.

Oren Glickman, Ido Dagan, and Moshe Koppel. 2005. A probabilistic classification approach for lexical textual entailment. In *Proc. of AAAI*.

Oren Glickman, Eyal Shnarch, and Ido Dagan. 2006. Lexical reference: a semantic matching subtask. In *Proceedings of the EMNLP*.

Nizar Habash and Bonnie Dorr. 2003. A categorial variation database for english. In *Proc. of NAACL*.

Houping Jia, Xiaojiang Huang, Tengfei Ma, Xiaojun Wan, and Jianguo Xiao. 2010. PKUTM participation at TAC 2010 RTE and summarization track. In *Proc. of TAC*.

Dekang Lin and Patrick Pantel. 2001. Discovery of inference rules for question answering. *Natural Language Engineering*, 7:343–360.

Bill MacCartney and Christopher D. Manning. 2007. Natural logic for textual inference. In *Proc. of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*.

Andrew MacKinlay and Timothy Baldwin. 2009. A baseline approach to the RTE5 search pilot. In *Proc. of TAC*.

Debarghya Majumdar and Pushpak Bhattacharyya. 2010. Lexical based text entailment system for main task of RTE6. In *Proc. of TAC*.

Shachar Mirkin, Roy Bar-Haim, Jonathan Berant, Ido Dagan, Eyal Shnarch, Asher Stern, and Idan Szpektor. 2009a. Addressing discourse and document structure in the RTE search task. In *Proc. of TAC*.

Shachar Mirkin, Ido Dagan, and Eyal Shnarch. 2009b. Evaluating the inferential utility of lexical-semantic resources. In *Proc. of EACL*.

Judea Pearl. 1988. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.

Eyal Shnarch, Jacob Goldberger, and Ido Dagan. 2011. A probabilistic modeling framework for lexical entailment. In *Proc. of ACL*, pages 558–563.

Idan Szpektor, Ido Dagan, Roy Bar-Haim, and Jacob Goldberger. 2008. Contextual preferences. In *Proc. of ACL-08: HLT*.

Marta Tatu and Dan Moldovan. 2007. COGEX at RTE 3. In *Proc. of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*.

Rui Wang, Yi Zhang, and Guenter Neumann. 2009. A joint syntactic-semantic representation for recognizing textual relatedness. In *Proc. of TAC*.

Fabio Massimo Zanzotto and Alessandro Moschitti. 2006. Automatic learning of textual entailments with cross-pair similarities. In *Proc. of ACL*.

A Appendix: An Iterative Procedure to Maximize $Q(\theta_{NA})$

There is no close-form expression for the parameter value θ_{NA} that maximizes expression (8) from Section 4.3. Instead we can apply the following iterative procedure. The derivative of $Q(\theta_{NA})$ is:

$$\frac{dQ(\theta_{NA})}{d\theta_{NA}} = \sum \left(\frac{l \cdot w_{hNA}}{\theta_{NA}} - \frac{(1 - w_{hNA}) l \cdot \theta_{NA}^{(l-1)}}{1 - \theta_{NA}^{l}} \right)$$

where $\frac{1}{l}$ is the hypothesis length and the summation is over all terms h in the training set. Setting this derivative to zero yields an equation which the optimal value satisfies:

$$\theta_{NA} = \frac{\sum l \cdot w_{hNA}}{\sum \frac{(1 - w_{hNA}) l \cdot \theta_{NA}^{(l-1)}}{1 - \theta_{NA}^{l}}} \tag{9}$$

Eq. (9) can be utilized as a heuristic iterative procedure to find the optimal value of θ_{NA}:

$$\theta_{NA} \leftarrow \frac{\sum l \cdot w_{hNA}}{\sum \frac{(1 - w_{hNA}) l \cdot \theta_{NA}^{(l-1)}}{1 - \theta_{NA}^{l}}}$$