Statistical Relational Learning to Recognise Textual Entailment

Abstract

We propose a novel approach to recognise textual entailment (RTE) following a two-stage architecture – alignment and decision – where both stages are based on semantic representations. In the alignment stage the entailment candidate pairs are represented and aligned using predicate-argument structures. In the decision stage, a Markov Logic Network (MLN) is learnt using rich relational information from the alignment stage to predict an entailment decision. We evaluate this approach using the RTE Challenge datasets. It shows comparable results against the average performance across participating systems, and very promising results for a subset of the datasets for which a semantic alignment can be found, evidencing the potential of MLNs for RTE.

1 Introduction

Recognising Textual Entailment (RTE) consists in deciding, given two text segments, whether the meaning of one segment (the (H)ypothesis) is entailed from the meaning of the other segment (the (T)ext) (Dagan and Glickman, 2005).

In order to address the task of RTE, most methods rely on machine learning algorithms. For example, a baseline method proposed by Mehdad and Magnini (2009) measures the word overlap between the T-H pairs. An overlap threshold is computed over the training data, and the test data is classified based on the learned threshold.

Another approach for RTE is to determine some kind of alignment between the T-H pairs. Since T is usually longer, H is aligned to a portion of T, and the best alignment is used to compute a similarity score. A limitation of such approaches is that instead of recognising a non-entailment, an alignment that fits an optimisation criterion will be returned (Marneffe et al., 2006), and thus the alignment by itself is a poor predictor for non-entailment. To solve this problem, Marneffe et al. (2006) divide the RTE task such that the alignment and the entailment decision are separate processes. The alignment phase is based on matching graph representations (i.e. dependency parse trees) of the T-H pair. For the entailment decision, rules that strongly suggest implications are designed. A specific rewrite rule between T and H can be positive if they represent entailment or negative otherwise.

Except for (Garrette et al., 2011), previous work using machine learning is based on propositional representations with simple attribute-value pairs as features. Garrette et al. (2011) combines first order logic and statistical methods for RTE. The approach uses discourse structures to represent T-H pairs, and a Markov Logic Network (MLN) model to perform inference in a probabilistic manner over implicativity and factivity, word meaning, and coreference. A threshold to decide the entailment given the MLN model output is manually set. Since their phenomena of interest are not present in the standard RTE datasets, they use handmade datasets. For other related work in the field, we refer the reader to (Androutsopoulos and Malakasiotis, 2010).

In this paper we describe an RTE approach following a multi-stage architecture. In contrast to Marneffe et al. (2006), both stages are based on semantic representations in an attempt to measure entailment based on the similarity of answers to the questions *Who did what to whom, when, where, why and how*. This is done through shallow semantic parsing using a Semantic Role Labelling (SRL) tool. Furthermore, instead of using simple similarity metrics to predict the entailment decision, we use rich relational features extracted from output of the predicate-argument alignment structures between T-H pairs. These are fed to an MLN

framework, which learns a model to reward pairs with similar predicates and similar arguments, and penalise pairs otherwise.

Different from (Garrette et al., 2011), we do not use a manually set threshold for the entailment decision and we evaluate our method on the standard RTE Challenge datasets, which are larger and contain naturally occurring linguistic constructions that can have an effect on the entailment decision. We compare our approach to baselines based on both MLN and standard machine learning algorithms such as SVM. We also compare our approach against the state of the art results from past editions of the RTE Challenge. Our approach shows a competitive performance for all datasets and promising results for a subset of them.

2 Proposed Approach

Our approach to RTE is based on a two-stage architecture: i) alignment, where predicate-argument structures of H and T are aligned; and ii) entailment decision, where the alignments are used to extract features (i.e. first order logic predicates) and these are used to build an MLN model.

2.1 Alignment Stage

We represent the T-H pair with SRLs as generated by SENNA (Collobert et al., 2011) and use TINE (Rios et al., 2011, 2012) to align any number of predicates and arguments between T and H. Instead of simply matching surface forms, TINE performs a flexible alignment of verb predicates by measuring (i) how similar their arguments are ($argScore$), (ii) and how related the predicates realisations are ($lexScore$). Both scores are combined as shown in Equation 1 to measure the similarity between the two predicates (Av, Bv) from a pair of sentences (A, B).

$$sim(Av, Bv) = wlex \times lexScore(Av, Bv) + warg \times argScore(Aarg, Barg) \quad (1)$$

where $wlex$ and $warg$ are the weights for each component, $argScore(Aarg, Barg)$ is the similarity between the arguments, computed as the cosine distance between the bag-of-words of the predicates' arguments Av, Bv. $lexScore(Av, Bv)$ is the similarity score of the predicates extracted using Dekang Lin's thesaurus (Lin, 1998). The pair of predicates that maximise Equation 1 produces an alignment with a one-to-one verb-arguments relation.

2.2 Entailment Decision Stage

In the entailment decision stage we use an MLN to predict the entailment relation of a given T-H pair. As an inherently semantic task, RTE should naturally benefit from knowledge about the relationships among elements in a text, in particular to check whether (some of) these relationships are equivalent in both T and H. It is extremely difficult to fully capture relational knowledge using standard propositional formalisms (attribute-value pairs), as it is hard to predict how many elements are involved in a relationship (e.g. a compound argument) or all possible values of these elements, and it is not possible to represent the sharing of values across attributes (e.g. the agent of a predicate which is also the object of another predicate).

MLN (Richardson and Domingos, 2006) provides a natural choice to address this task as it unifies first order logic and probabilistic graphical models in a framework that enables the representation of rich relational information (such as syntactic and semantic relations) and inference under uncertainty. This framework learns weights for first order logic formulas, which are then used to build Markov networks that can be queried in the presence of new instances.

The basis for our first order logic formulas are the alignments produced in the previous stage. At inference time, an aligned pair with similar situations and similar participants will likely hold an entailment relation. An alignment consists of a pair of verbs and their corresponding arguments. Several features extracted from these alignments are used as predicates to build a Markov Network. We formulate three variants with these predicates: a baseline model with simpler, non-relational features, a relational model, and a variant that adds back-off strategy to the relational model.

2.2.1 Baseline Model

Our baseline models the entailment decision using the following non-relational features:

Bag-of-words and Part of Speech (PoS) tags For each token in the T-H pair we extract their lemmas and part of speech tags. We represent it by the predicate $TokenBaseline(pid, token)$.

Word Overlap For each T-H pair we compute the number of lemmas shared between the T and H: $Overlap(pid, num)$.

pid is the id of a T-H pair, $token$ is the lemma or the PoS tag (each one has a separate predicate), and num is the overlap score.

We define the following MLN formulas for the entailment decision:

$$TokenBaseline(pid, +token) \Rightarrow Entailment(+d, pid)$$
$$Overlap(pid, +n) \Rightarrow Entailment(+d, pid)$$

where the predicate $Entailment(+d, pid)$ takes two possible values for the decision d: $true$ or $false$. The $+$ operator indicates that a weight will be learned for each grounding of the formula. The entailment decision is a hidden variable in the MLN model and it is used to query the MLN.

2.2.2 Relational Model

This variant takes advantage of the MLN ability to handle relational information. New predicates and formulas that take into consideration the semantic relations between the arguments and verbs are added to the baseline. The following variables are created to represent this information: Arg and $Verb$. Figure 1 shows the relationships between these variables in a Markov Network. The variable Back-off is described in Section 2.2.3.

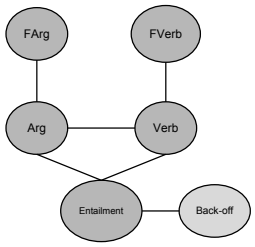

Figure 1: Markov network of our RTE model

The value of Arg is the label given by the SRL parser for the aligned arguments (e.g. ARG1). The value of $Verb$ is the lexical realisation of the verbs, i.e., the aligned verbs themselves. Furthermore, the aligned arguments and the aligned verbs have features related to them: $FArg$ is the set of features related to the arguments, and $FVerb$ is the set of features related to the verbs.

The features for each token of the aligned arguments are as follows:

Lexical Word, lemma and PoS of each token.

Synonyms The 20 most similar words from Dekang Lin's thesaurus for each token. A predicate is created for each similar word.

Hypernyms The complete hypernym tree of each noun in its first sense in WordNet. A predicate for each hypernym is created.

These argument features are represented by the following formula:

$$Token(aid, pid, +tfeature) \wedge Arg(aid, vid, pid) \Rightarrow Entailment(+d, pid)$$

where $tfeature$ takes the value of each of the previous features, aid and vid are the values of the Arg and $Verb$ variables

For the aligned verbs, the following features are extracted:

Bag-of-words VerbNet $bowfeature$ is the lexical realisation of the classes shared between the verbs in VerbNet. Looking at the semantic classes of the aligned verbs brings extra information about how similar they are:

$$BowVN(vid, +bowfeature) \wedge Verb(vid, pid) \Rightarrow Entailment(+d, pid)$$

Strong Context $strfeature$ compares components in Equation 1. If the value of $argScore(Aarg, Barg)$ is larger than that of $lexScore(Av, Bv)$, this feature is set to 1, i.e., the similarity of the context of the aligned verbs is stronger than the relationship between them; it is 0 otherwise:

$$StrongCon(vid, +strfeature) \wedge Verb(vid, pid) \Rightarrow Entailment(+d, pid)$$

Similarity VerbNet $simvnfeature$ is set to 1 if the verbs share at least one class in VerbNet; 0 otherwise:

$$SimVN(vid, +simvnfeature) \wedge Verb(vid, pid) \Rightarrow Entailment(+d, pid)$$

Similarity VerbOcean $simvofeature$ is 1 if the verbs have the *similar* relation as given by VerbOcean (Chklovski and Pantel, 2004);[1] 0 otherwise:

$$SimVO(vid, +simvofeature) \wedge Verb(vid, pid) \Rightarrow Entailment(+d, pid)$$

Similarity Directional $simdfeature$ is 1 if the verbs hold an entailment relation as given by the Directional Database (Kotlerman et al., 2010);[2] 0 otherwise:

$$SimD(vid, +simdfeature) \wedge Verb(vid, pid) \Rightarrow Entailment(+d, pid)$$

Token Verbs The predicate contains the lemmas of the aligned verbs:

$$TokenVerb(vid, +tokenvfeature) \wedge Verb(vid, pid) \Rightarrow Entailment(+d, pid)$$

[1] VerbOcean contains different relations between verbs.
[2] It contains directional lexical entailment rules.

Finally, the relation between Arg and $Verb$ is defined by the formula:

$$Arg(aid, vid, pid) \wedge Verb(vid, pid) \Rightarrow Entailment(+d, pid)$$

The formulas sharing variables vid and aid indicate relationships between the aligned arguments and the aligned verbs, as well as their corresponding features given the SRL structure. pid relates the previous predicates to the decision of an entailment pair. Many of these formulas can take up multiple values through multiple groundings (e.g. the hypernyms of nouns). With these formulas the MLN builds a Markov Network, which we can be queried for an entailment decision. For a new T-H pair the model can predict a decision based on the type of arguments it has, the features of the words in the arguments, the alignment between its verbs, and the relations (i.e. features) between such verbs.

2.2.3 Back-off Model

In the alignment stage the metric cannot align some of the T-H pairs, mostly because SENNA does not produce any SRL structure for certain pairs. In order to be able to make a decision for these pairs using MLNs, we add a back-off strategy based on the baseline model: whenever a T-H pair is not aligned we use solely the predicates computed by the baseline model. Therefore, a new node – *Back-off* – is attached to the entailment decision in Figure 1.

3 Experiments and Results

We use the Alchemy[3] toolkit and the datasets from the RTE challenges 1-3 (Dagan and Glickman, 2005; Bar-Haim et al., 2006; Giampiccolo et al., 2007) to evaluate our MLN models. To predict the entailment decision we take the marginal probabilities that Alchemy outputs for a given query, i.e., the $Entailment$ predicate. The query with the highest probability gives the entailment decision.

For comparison, we developed a common baseline that computes the overlap of lemmas between T-H pairs as features and uses SVM algorithm with a linear kernel for the binary entailment decision (Mehdad and Magnini, 2009).

Table 1 shows the performance of our baseline and back-off models against that of the baseline SVM. It also shows the top system and the average accuracy scores for all systems reported in the RTE challenges. The back-off model achieves a competitive performance compared to the average of the participating systems, particularly on the RTE-1 dataset (Avg. systems). However, its performance is far from that of the best system (Top). In an attempt to understand whether this problem is down to the alignment stage or the entailment decision stage, we selected only the T-H test pairs for which the TINE finds alignments: 162 pairs (out of 287) for RTE-1, 463 pairs (out of 800) for RTE-2, and 385 pairs (out of 800) for RTE-3. We test the relational model on these subsets and compare it against the SVM baseline. Table 2 shows the results, where the relational model clearly outperforms the SVM baseline, by a particularly large margin on the RTE-3 dataset. This shows the potential of the relational features and MLNs for RTE.

Algorithm	RTE-1	RTE-2	RTE-3
Top system	70%	75%	80%
Avg. systems	55%	59%	61%
SVM	49%	53%	57%
Baseline model	56%	54%	51%
Back-off model	57%	49%	55%

Table 1: Accuracy on RTE 1-3 datasets

Algorithm	RTE-1	RTE-2	RTE-3
SVM	50%	51%	56%
Relational model	57%	55%	78%

Table 2: Accuracy on a subset of RTE 1-3 where an alignment is produced by TINE for T-H

4 Conclusions

Our preliminary results on using a relational statistical learning framework for the RTE task showed promising results: while a gap in accuracy is observed with respect to the state of the art approaches, we showed that this is mostly due to the poor performance of the semantic alignment tool used in the pre-processing stage. This yields a low coverage of the relational model, and as a consequence the use of a very simple approach as back-off for cases of T-H without an alignment. Future work will include improvements in the alignment stage, such as using syntactic structures as opposed to semantics as in (de Marneffe et al., 2007), and the use of better back-off strategies.

[3] http://alchemy.cs.washington.edu/

References

Androutsopoulos, I., Malakasiotis, P.: A survey of paraphrasing and textual entailment methods. J. Artif. Int. Res. 38(1), 135–187 (2010)

Bar-Haim, R., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B., Szpektor, I.: The second pascal recognising textual entailment challenge. In: Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment. Venice, Italy (2006)

Chklovski, T., Pantel, P.: Verbocean: Mining the web for fine-grained semantic verb relations. In: EMNLP. pp. 33–40. ACL (2004)

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.P.: Journal of Machine Learning Research 12, 2493–2537 (2011)

Dagan, I., Glickman, O.: The pascal recognising textual entailment challenge. In: In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment (2005)

Garrette, D., Erk, K., Mooney, R.: Integrating logical representations with probabilistic information using Markov logic. In: Proceedings of the Ninth International Conference on Computational Semantics (IWCS 2011). pp. 105–114 (2011)

Giampiccolo, D., Magnini, B., Dagan, I., Dolan, B.: The third pascal recognizing textual entailment challenge. In: Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. pp. 1–9. Prague (2007)

Kotlerman, L., Dagan, I., Szpektor, I., Zhitomirsky-geffet, M.: Directional distributional similarity for lexical inference. Nat. Lang. Eng. 16(4), 359–389 (2010)

Lin, D.: Automatic retrieval and clustering of similar words. In: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics. pp. 768–774. Montréal, Canada (1998)

de Marneffe, M.C., Grenager, T., MacCartney, B., Cer, D.M., Ramage, D., Kiddon, C., Manning, C.D.: Robust graph alignment methods for textual inference and machine reading. In: AAAI Spring Symposium: Machine Reading. pp. 36–42 (2007)

Marneffe, M.C.D., MacCartney, B., Grenager, T., Cer, D., Rafferty, A., Manning, C.D.: Learning to distinguish valid textual entailments. In: Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment. Venice, Italy (2006)

Mehdad, Y., Magnini, B.: A word overlap baseline for the recognizing textual entailment task (2009)

Richardson, M., Domingos, P.: Markov logic networks. Mach. Learn. 62(1-2), 107–136 (2006)

Rios, M., Aziz, W., Specia, L.: TINE: A metric to assess MT adequacy. In: Proceedings of the Sixth Workshop on Statistical Machine Translation. pp. 116–122. Edinburgh, Scotland (2011)

Rios, M., Aziz, W., Specia, L.: UOW: Semantically informed text similarity. In: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012). pp. 673–678. Montréal, Canada (2012)