10.12871/CLICIT2014133

Distributed Smoothed Tree Kernel

Lorenzo Ferrone
University of Rome "Tor Vergata"
Via del Politecnico 1
00133 Roma, Italy
lorenzo.ferrone@gmail.com

Fabio Massimo Zanzotto
University of Rome "Tor Vergata"
Via del Politecnico 1
00133 Roma, Italy
fabio.massimo.zanzotto@uniroma2.it

Abstract

English. In this paper we explore the possibility to merge the world of Compositional Distributional Semantic Models (CDSM) with Tree Kernels (TK). In particular, we will introduce a specific tree kernel (*smoothed tree kernel*, or STK) and then show that is possibile to approximate such kernel with the dot product of two vectors obtained compositionally from the sentences, creating in such a way a new CDSM.

Italiano. *In questo paper vogliamo esplorare la possibilità di unire il mondo dei metodi di semantica distribuzione composizionale (CDSM) con quello dei tree Kernel (TK). In particolare introdurremo un particolare tree kernel e poi mostreremo che possibile approssimare questo kernel tramite il prodotto scalare tra due vettori ottenuti composizionalmente a partire dalle frasi di partenza, creando così di fatto un nuovo modello di semantica distribuzionale composizionale.*

1 Introduction

Compositional distributional semantics is a flourishing research area that leverages distributional semantics (see Baroni and Lenci (2010)) to produce meaning of simple phrases and full sentences (hereafter called *text fragments*). The aim is to scale up the success of word-level relatedness detection to longer fragments of text. Determining similarity or relatedness among sentences is useful for many applications, such as multi-document summarization, recognizing textual entailment (Dagan et al., 2013), and semantic textual similarity detection (Agirre et al., 2013; Jurgens et al., 2014). Compositional distributional semantics models (CDSMs) are functions mapping text fragments to vectors (or higher-order tensors). Functions for simple phrases directly map distributional vectors of words to distributional vectors for the phrases (Mitchell and Lapata, 2008; Baroni and Zamparelli, 2010; Zanzotto et al., 2010). Functions for full sentences are generally defined as recursive functions over the ones for phrases (Socher et al., 2011). Distributional vectors for text fragments are then used as inner layers in neural networks, or to compute similarity among text fragments via dot product.

CDSMs generally exploit structured representations t^x of text fragments x to derive their meaning $f(t^x)$, but the structural information, although extremely important, is obfuscated in the final vectors. Structure and meaning can interact in unexpected ways when computing cosine similarity (or dot product) between vectors of two text fragments, as shown for full additive models in (Ferrone and Zanzotto, 2013).

Smoothed tree kernels (STK) (Croce et al., 2011) instead realize a clearer interaction between structural information and distributional meaning. STKs are specific realizations of convolution kernels (Haussler, 1999) where the similarity function is recursively (and, thus, compositionally) computed. Distributional vectors are used to represent word meaning in computing the similarity among nodes. STKs, however, are not considered part of the CDSMs family. As usual in kernel machines (Cristianini and Shawe-Taylor, 2000), STKs directly compute the similarity between two text fragments x and y over their tree representations t^x and t^y, that is, $STK(t^x, t^y)$. The function f that maps trees into vectors is

only implicitly used, and, thus, $STK(t^x, t^y)$ is not explicitly expressed as the dot product or the cosine between $f(t^x)$ and $f(t^y)$.

Such a function f, which is the underlying reproducing function of the kernel (Aronszajn, 1950), is a CDSM since it maps trees to vectors by using distributional meaning. However, the huge nality of $\mathbb{R}^n$ (since it has to represent the set of all possible subtrees) prevents to actually compute the function $f(t)$, which thus can only remain *implicit*.

Distributed tree kernels (DTK) (Zanzotto and Dell'Arciprete, 2012) partially solve the last problem. DTKs approximate standard tree kernels (such as (Collins and Duffy, 2002)) by defining an *explicit* function DT that maps trees to vectors in $\mathbb{R}^m$ where $m \ll n$ and $\mathbb{R}^n$ is the explicit space for tree kernels. DTKs approximate standard tree kernels (TK), that is, $\langle DT(t^x), DT(t^y)\rangle \approx TK(t^x, t^y)$, by approximating the corresponding reproducing function. Thus, these distributed trees are small vectors that encode structural information. In DTKs tree nodes u and v are represented by nearly orthonormal vectors, that is, vectors $\overrightarrow{u}$ and $\overrightarrow{v}$ such that $\langle \overrightarrow{u}, \overrightarrow{v}\rangle \approx \delta(\overrightarrow{u}, \overrightarrow{v})$ where δ is the Kroneker's delta. This is in contrast with distributional semantics vectors where $\langle \overrightarrow{u}, \overrightarrow{v}\rangle$ is allowed to be any value in $[0, 1]$ according to the similarity between the words v and u. In this paper, leveraging on distributed trees, we present a novel class of CDSMs that encode both structure and distributional meaning: the distributed smoothed trees (DST). DSTs carry structure and distributional meaning on a rank-2 tensor (a matrix): one dimension encodes the structure and one dimension encodes the meaning. By using DSTs to compute the similarity among sentences with a generalized dot product (or cosine), we implicitly define the distributed smoothed tree kernels (DSTK) which approximate the corresponding STKs. We present two DSTs along with the two smoothed tree kernels (STKs) that they approximate. We experiment with our DSTs to show that their generalized dot products approximate STKs by directly comparing the produced similarities and by comparing their performances on two tasks: recognizing textual entailment (RTE) and semantic similarity detection (STS). Both experiments show that the dot product on DSTs approximates STKs and, thus, DSTs encode both structural and distributional semantics of text fragments in tractable rank-2 tensors. Experiments on STS and RTE show that distributional semantics encoded in DSTs increases performance over structure-only kernels. DSTs are the first positive way of taking into account both structure and distributional meaning in CDSMs. The rest of the paper is organized as follows. Section 2.1 introduces the basic notation used in the paper. Section 2 describe our distributed smoothed trees as compositional distributional semantic models that can represent both structural and semantic information. Section 4 reports on the experiments. Finally, Section 5 draws some conclusions.

2 Distributed Smoothed Tree Kernel

We here propose a model that can be considered a compositional distributional semantic model as it transforms sentences into matrices that can then used by the learner as feature vectors. Our model is called *Distributed Smoothed Tree Kernel* (Ferrone and Zanzotto, 2014) as it mixes the distributed trees (Zanzotto and Dell'Arciprete, 2012) representing syntactic information with distributional semantic vectors representing semantic information.

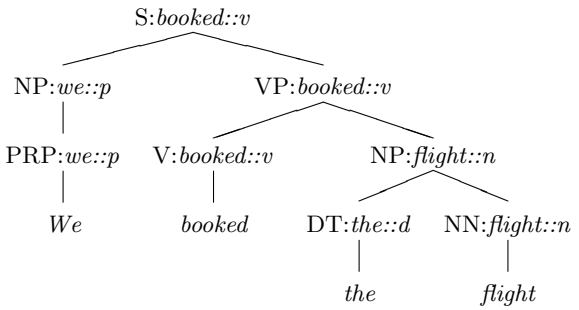

Figure 1: A lexicalized tree

2.1 Notation

Before describing the *distributed smoothed trees* (DST) we introduce a formal way to denote constituency-based *lexicalized parse trees*, as DSTs exploit this kind of data structures. *Lexicalized trees* are denoted with the letter t and $N(t)$ denotes the set of non terminal nodes of tree t. Each non-terminal node $n \in N(t)$

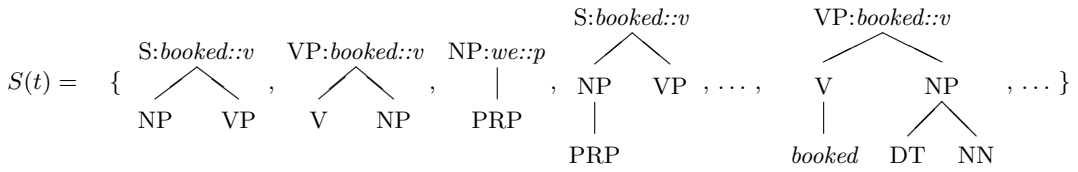

Figure 2: Subtrees of the tree t in Figure 1 (a non-exhaustive list)

has a label l_n composed of two parts $l_n = (s_n, w_n)$: s_n is the syntactic label, while w_n is the semantic headword of the tree headed by n, along with its part-of-speech tag. Terminal nodes of trees are treated differently, these nodes represent only words w_n without any additional information, and their labels thus only consist of the word itself (see Fig. 1). The structure of a DST is represented as follows: Given a tree t, $\mathsf{h}(t)$ is its root node and $\mathsf{s}(t)$ is the tree formed from t but considering only the syntactic structure (that is, only the s_n part of the labels), $c_i(n)$ denotes i-th child of a node n. As usual for constituency-based parse trees, pre-terminal nodes are nodes that have a single terminal node as child.

Finally, we use $\vec{w_n} \in \mathbb{R}^k$ to denote the *distributional* vector for word w_n.

2.2 The method at a glance

We describe here the approach in a few sentences. In line with tree kernels over structures (Collins and Duffy, 2002), we introduce the set $S(t)$ of the subtrees t_i of a given lexicalized tree t. A subtree t_i is in the set $S(t)$ if $\mathsf{s}(t_i)$ is a subtree of $\mathsf{s}(t)$ and, if n is a node in t_i, all the siblings of n in t are in t_i. For each node of t_i we only consider its syntactic label s_n, except for the head $\mathsf{h}(t_i)$ for which we also consider its semantic component w_n (see Fig. 2). The functions DSTs we define compute the following:

$$DST(t) = \mathbf{T} = \sum_{t_i \in S(t)} \mathbf{T}_i$$

where $\mathbf{T}_i$ is the matrix associated to each subtree t_i. The similarity between two text fragments a and b represented as lexicalized trees t^a and t^b can be computed using the Frobenius product between the two matrices $\mathbf{T}^a$ and $\mathbf{T}^b$, that is:

$$\langle \mathbf{T}^a, \mathbf{T}^b \rangle_F = \sum_{\substack{t_i^a \in S(t^a) \\ t_j^b \in S(t^b)}} \langle \mathbf{T}_i^a, \mathbf{T}_j^b \rangle_F \qquad (1)$$

We want to obtain that the product $\langle \mathbf{T}_i^a, \mathbf{T}_j^b \rangle_F$ approximates the dot product between the distributional vectors of the head words ($\langle \mathbf{T}_i^a, \mathbf{T}_j^b \rangle_F \approx \langle \vec{\mathsf{h}}(t_i^a), \vec{\mathsf{h}}(t_j^b) \rangle$) whenever the syntactic structure of the subtrees is the same (that is $\mathsf{s}(t_i^a) = \mathsf{s}(t_j^b)$), and $\langle \mathbf{T}_i^a, \mathbf{T}_j^b \rangle_F \approx 0$ otherwise. This property is expressed as:

$$\langle \mathbf{T}_i^a, \mathbf{T}_j^b \rangle_F \approx \delta(\mathsf{s}(t_i^a), \mathsf{s}(t_j^b)) \cdot \langle \vec{\mathsf{h}}(t_i^a), \vec{\mathsf{h}}(t_j^b) \rangle \quad (2)$$

To obtain the above property, we define

$$\mathbf{T}_i = \vec{\mathsf{s}}(t_i) \vec{w}_{\mathsf{h}(t_i)}^{\top}$$

where $\vec{\mathsf{s}}(t_i)$ are distributed tree fragment (Zanzotto and Dell'Arciprete, 2012) for the subtree t and $\vec{w}_{\mathsf{h}(t_i)}$ is the distributional vector of the head of the subtree t. Distributed tree fragments have the property that $\vec{\mathsf{s}}(t_i)\vec{\mathsf{s}}(t_j) \approx \delta(t_i, t_j)$. Thus, exploiting the fact that: $\langle \vec{a}\vec{w}^{\top}, \vec{b}\vec{v}^{\top} \rangle_F = \langle \vec{a}, \vec{b} \rangle \cdot \langle \vec{w}, \vec{v} \rangle$, we have that Equation 2 is satisfied as:

$$\begin{aligned} \langle \mathbf{T}_i, \mathbf{T}_j \rangle_F &= \langle \vec{\mathsf{s}}(t_i), \vec{\mathsf{s}}(t_j) \rangle \cdot \langle \vec{w}_{\mathsf{h}(t_i)}, \vec{w}_{\mathsf{h}(t_j)} \rangle \\ &\approx \delta(\mathsf{s}(t_i), \mathsf{s}(t_j)) \cdot \langle \vec{w}_{\mathsf{h}(t_i)}, \vec{w}_{\mathsf{h}(t_j)} \rangle \end{aligned}$$

It is possible to show that the overall compositional distributional model $DST(t)$ can be obtained with a recursive algorithm that exploits vectors of the nodes of the tree.

3 The Approximated Smoothed Tree Kernels

The CDSM we proposed approximates a specific tree kernel belonging to the smoothed tree kernels class. This recursively computes (but, the recursive formulation is not given here) the following general equation:

$$STK(t^a, t^b) = \sum_{\substack{t_i \in S(t^a) \\ t_j \in S(t^b)}} \omega(t_i, t_j)$$

		RTE1	RTE2	RTE3	RTE5	headl	FNWN	OnWN	SMT
STK vs DSTK	1024	0.86	0.84	0.90	0.84	0.87	0.65	0.95	0.77
	2048	0.87	0.84	0.91	0.84	0.90	0.65	0.96	0.77

Table 1: Spearman's correlation between Distributed Smoothed Tree Kernels and Smoothed Tree Kernels

where $\omega(t_i, t_j)$ is the similarity weight between two subtrees t_i and t_j. $DTSK$ approximates STK, where the weights are defined as follows:

$$\omega(t_i, t_j) = \alpha \cdot \langle \vec{w_{\mathsf{h}(t_i)}}, \vec{w_{\mathsf{h}(t_j)}} \rangle \cdot \delta(\mathsf{s}(t_i), \mathsf{s}(t_j))$$

Where $\alpha = \sqrt{\lambda^{|N(t_i)|+|N(t_j)|}}$ and λ is a parameter.

4 Experimental investigation

Generic settings We experimented with two datasets: the Recognizing Textual Entailment datasets (RTE) (Dagan et al., 2006) and the the Semantic Textual Similarity 2013 datasets (STS) (Agirre et al., 2013). The STS task consists of determining the degree of similarity (ranging from 0 to 5) between two sentences. The STS datasets contains 5 datasets: headlines, OnWN, FNWN and SMT which contains respectively 750, 561, 189 and 750 RTE is instead the task of deciding whether a long text T entails a shorter text, typically a single sentence, called hypothesis H. It has been often seen as a classification task. We used four datasets: RTE1, RTE2, RTE3, and RTE5. We parsed the sentence with the Stanford Parser (Klein and Manning, 2003) and extracted the heads for use in the lexicalized trees with Collins' rules (Collins, 2003). Distributional vectors are derived with DISSECT (Dinu et al., 2013) from a corpus obtained by the concatenation of ukWaC, a mid-2009 dump of the English Wikipedia and the British National Corpus for a total of about 2.8 billion words. The raw count vectors were transformed into positive Pointwise Mutual Information scores and reduced to 300 dimensions by Singular Value Decomposition. This setup was picked without tuning, as we found it effective in previous, unrelated experiments. To build our DTSKs we used the implementation of the distributed tree kernels[1]. We used 1024 and 2048 as the dimension of the distributed vectors, the weight λ is set to 0.4 as it is a value generally considered optimal for many applications (see also (Zanzotto and Dell'Arciprete, 2012)). To test the quality of the approximation we computed the Spearman's correlation between values produced by our $DSTK$ and by the standard versions of the smoothed tree kernel. We obtained text fragment pairs by randomly sampling two text fragments in the selected set. For each set, we produced exactly the number of examples in the set, e.g., we produced 567 pairs for RTE1, etc.

Results Table 1 reports the results for the correlation experiments. We report the Spearman's correlations over the different sets (and different dimensions of distributed vectors) between our $DSTK$ and the STK. The correlation is above 0.80 in average for both RTE and STS datasets. The approximation also depends on the size of the distributed vectors. Higher dimensions yield to better approximation: if we increase the distributed vectors dimension from 1024 to 2048 the correlation between $DSTK$ and STK increases. This direct analysis of the correlation shows that our CDSM are approximating the corresponding kernel function and there is room of improvement by increasing the size of distributed vectors.

5 Conclusions and future work

Distributed Smoothed Trees (DST) are a novel class of Compositional Distributional Semantics Models (CDSM) that effectively encode structural information and distributional semantics in tractable rank-2 tensors, as experiments show. The paper shows that DSTs contribute to close the gap between two apparently different approaches: CDSMs and convolution kernels. This contribute to start a discussion on a deeper understanding of the representation power of structural information of existing CDSMs.

[1] http://code.google.com/p/distributed-tree-kernels/

References

[Agirre et al.2013] Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *sem 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

[Aronszajn1950] N. Aronszajn. 1950. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404.

[Baroni and Lenci2010] Marco Baroni and Alessandro Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Comput. Linguist.*, 36(4):673–721, December.

[Baroni and Zamparelli2010] Marco Baroni and Roberto Zamparelli. 2010. Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1183–1193, Cambridge, MA, October. Association for Computational Linguistics.

[Collins and Duffy2002] Michael Collins and Nigel Duffy. 2002. New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron. In *Proceedings of ACL02*.

[Collins2003] Michael Collins. 2003. Head-driven statistical models for natural language parsing. *Comput. Linguist.*, 29(4):589–637.

[Cristianini and Shawe-Taylor2000] Nello Cristianini and John Shawe-Taylor. 2000. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, March.

[Croce et al.2011] Danilo Croce, Alessandro Moschitti, and Roberto Basili. 2011. Structured lexical similarity via convolution kernels on dependency trees. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1034–1046, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Dagan et al.2006] Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The pascal recognising textual entailment challenge. In Quiñonero-Candela et al., editor, *LNAI 3944: MLCW 2005*, pages 177–190. Springer-Verlag, Milan, Italy.

[Dagan et al.2013] Ido Dagan, Dan Roth, Mark Sammons, and Fabio Massimo Zanzotto. 2013. *Recognizing Textual Entailment: Models and Applications*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

[Dinu et al.2013] Georgiana Dinu, Nghia The Pham, and Marco Baroni. 2013. DISSECT: DIStributional SEmantics Composition Toolkit. In *Proceedings of ACL (System Demonstrations)*, pages 31–36, Sofia, Bulgaria.

[Ferrone and Zanzotto2013] Lorenzo Ferrone and Fabio Massimo Zanzotto. 2013. Linear compositional distributional semantics and structural kernels. In *Proceedings of the Joint Symposium of Semantic Processing (JSSP)*.

[Ferrone and Zanzotto2014] Lorenzo Ferrone and Fabio Massimo Zanzotto. 2014. Towards syntax-aware compositional distributional semantic models. In *Proceedings of Coling 2014*. COLING, Dublin, Ireland, Aug 23–Aug 29.

[Haussler1999] David Haussler. 1999. Convolution kernels on discrete structures. Technical report, University of California at Santa Cruz.

[Jurgens et al.2014] David Jurgens, Mohammad Taher Pilehvar, and Roberto Navigli. 2014. Semeval-2014 task 3: Cross-level semantic similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 17–26, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

[Klein and Manning2003] Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 423–430, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Mitchell and Lapata2008] Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition. In *Proceedings of ACL-08: HLT*, pages 236–244, Columbus, Ohio, June. Association for Computational Linguistics.

[Socher et al.2011] Richard Socher, Eric H. Huang, Jeffrey Pennington, Andrew Y. Ng, and Christopher D. Manning. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Advances in Neural Information Processing Systems 24*.

[Zanzotto and Dell'Arciprete2012] F.M. Zanzotto and L. Dell'Arciprete. 2012. Distributed tree kernels. In *Proceedings of International Conference on Machine Learning*, pages 193–200.

[Zanzotto et al.2010] Fabio Massimo Zanzotto, Ioannis Korkontzelos, Francesca Fallucchi, and Suresh Manandhar. 2010. Estimating linear models for compositional distributional

semantics. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*, August,.