Resource-lean modeling of coherence in commonsense stories

Niko Schenk, Christian Chiarcos

Angaben zur Veröffentlichung / Publication details:

Schenk, Niko, and Christian Chiarcos. 2017. "Resource-lean modeling of coherence in commonsense stories." In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics, April 3, 2017, Valencia, Spain*, edited by Michael Roth, Nasrin Mostafazadeh, Nathanael Chambers, and Annie Louis, 68–73. Stroudsburg, PA: Association for Computational Linguistics. https://doi.org/10.18653/v1/w17-0910.

Resource-Lean Modeling of Coherence in Commonsense Stories

Niko Schenk and **Christian Chiarcos**
Applied Computational Linguistics Lab
Goethe University Frankfurt am Main, Germany
{n.schenk,chiarcos}@em.uni-frankfurt.de

Abstract

We present a resource-lean neural recognizer for modeling coherence in commonsense stories. Our lightweight system is inspired by successful attempts to modeling discourse relations and stands out due to its simplicity and easy optimization compared to prior approaches to narrative script learning. We evaluate our approach in the Story Cloze Test[1] demonstrating an absolute improvement in accuracy of 4.7% over state-of-the-art implementations.

1 Introduction

Semantic applications related to Natural Language Understanding have seen a recent surge of interest within the NLP community, and *story understanding* can be regarded as one of the supreme disciplines in that field. Closely related to Machine Reading (Hovy, 2006) and script learning (Schank and Abelson, 1977; Mooney and DeJong, 1985), it is a highly challenging task which is built on top of a cascade of core NLP applications, including—among others—causal/temporal relation recognition (Mirza and Tonelli, 2016), event extraction (UzZaman and Allen, 2010), (implicit) semantic role labeling (Gerber and Chai, 2012; Schenk and Chiarcos, 2016) or inter-sentential discourse parsing (Mihaylov and Frank, 2016).

Recent progress has been made in the field of *narrative understanding*: a variety of successful approaches have been introduced, ranging from narrative chains (Chambers and Jurafsky, 2008) to script learning techniques (Regneri et al., 2010), or event schemas (Nguyen et al., 2015). What all these approaches have in common is that they ultimately seek to find a way to prototypically model the causal and correlational relationships between events, and also to obtain a structured (ideally more compact and abstract) representation of the underlying commonsense knowledge which is encoded in the respective story. The downside of these approaches is that they are feature-rich (potentially hand-crafted) and therefore costly and domain-specific to a large extent. On a related note, Mostafazadeh et al. (2016a) demonstrate that there is still room for improvement when testing the performances of these state-of-the-art techniques for learning procedural knowledge on an independent evaluation set.

Our Contribution: In this paper, we propose a lightweight, resource-lean framework for modeling procedural knowledge in commonsense stories whose only source of information are distributed word representations. We cast the problem of modeling text coherence as a special case of discourse processing in which our model jointly learns to distinguish correct from incorrect story endings. Our approach is inspired by promising related attempts using event embeddings and neural methods for script learning (Modi and Titov, 2014; Pichotta and Mooney, 2016). Our system is an end-to-end implementation of the ideas sketched in Mostafazadeh et al. (2016b) of the *joint paragraph and sentence level* model (cf. Section 3 for details). We evaluate our approach in the Story Cloze Test, a task for predicting story continuations. Despite its simplicity, our system demonstrates superior performance on the designated data over previous approaches to script learning and—due to its language and genre-independence—it also represents a solid basis for further optimization towards other textual domains.

[1]The shared task of the LSDSem 2017 workshop on *Linking Models of Lexical, Sentential and Discourse-level Semantics:*
http://www.coli.uni-saarland.de/~mroth/LSDSem/,
http://cs.rochester.edu/nlp/rocstories/LSDSem17/,
https://competitions.codalab.org/competitions/15333

Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics, pages 68–73,
Valencia, Spain, April 3, 2017.

Four-Sentence Core Story	Quiz 1	Quiz 2
I asked Sarah out on a date. She said yes. I was so excited for our date together. We went to dinner and then a movie.	I had a terrible time. (***wrong*** ending)	I got to kiss Sarah goodnight. (***correct*** ending)

Table 1: An example of a *ROCStory* consisting of a core story and two alternative continuations.

2 The Story Cloze Test

2.1 Task Description

In the *Story Cloze Test* a participating system is presented with a four-sentence *core story* along with two alternative single-sentence endings, i.e. a correct and a wrong one. The system is then supposed to select the correct ending based on a semantic analysis of the individual story components. For this binary choice, outputs are evaluated on accuracy level.

2.2 Data

The shared task organizers provide participants with a large corpus of approx. 98k five-sentence everyday life stories (Mostafazadeh et al., 2016a, *ROCStories*[2]) for training their narrative story understanding models. Also a validation and a test set are available (each containing 1,872 instances). The former serves for parameter optimization, whereas final performance is evaluated on the test set. The instances in all three sets are mutually exclusive. Note that in addition to the *ROCStories*, both validation and test sets include an additional *wrong* 5th-sentence story ending (either in first or second position) plus hand-annotated decisions about which story ending is the right one. As an illustration, consider the example in Table 1 consisting of a core story and two alternative continuations (quizzes). The global semantics of this *ROCStory* is driven by two factors: i) a latent discoursive, temporal/causal relationship between the individual events in each sentence and ii) a resulting positive outcome of the story. Clearly, the right ending is the second quiz. Note that for all stories in the data set, the task of choosing the correct ending is human solvable with perfect agreement (Mostafazadeh et al., 2016a).

[2] http://cs.rochester.edu/nlp/rocstories/

3 Approach

Our proposed model architecture for finding the right story continuation is inspired by novel works from (shallow) discourse parsing, most notably by the recent success of neural network-based frameworks in that field (Xue et al., 2016; Schenk et al., 2016; Wang and Lan, 2016). Specifically for *implicit* discourse relations, i.e. for those sentence pairs which, for instance, can signal a temporal, contrast or contingency relation, but which suffer from the absence of an explicit discourse marker (such as *but* or *because*), it has been shown that the interaction of properly tuned distributed representations over adjacent text spans can be particularly powerful in the relation classification task. We cast the Story Cloze test as a special case of implicit discourse relation recognition and attempt to model an underlying, latent connection between a core story and its correct vs. incorrect continuation. For instance, the final example sentence in the core story in Table 1 and the two adjacent quizzes could be treated as argument pairs (*Arg1* and *Arg2*) in the classical view of the Penn Discourse Treebank (Prasad et al., 2008), distinguishing different types of implicit discourse relations that hold between them.[3]

> ***Arg1***: We went to dinner and then a movie.
> ***Arg2***: I had a terrible time.
> TEMP.**SYNCHRONOUS**
>
> ***Arg1***: We went to dinner and then a movie.
> ***Arg2***: I got to kiss Sarah goodnight.
> TEMP.**ASYNCHRONOUS**.PRECEDENCE

Here, in the first example, the label SYNCHRONOUS indicates that the two situations in both arguments overlap temporally (which could be signaled explicitly by *while*, for instance), whereas in the second example ASYNCHRONOUS.PRECEDENCE implies a temporal or-

[3] For details, see https://www.seas.upenn.edu/~pdtb/PDTBAPI/pdtb-annotation-manual.pdf

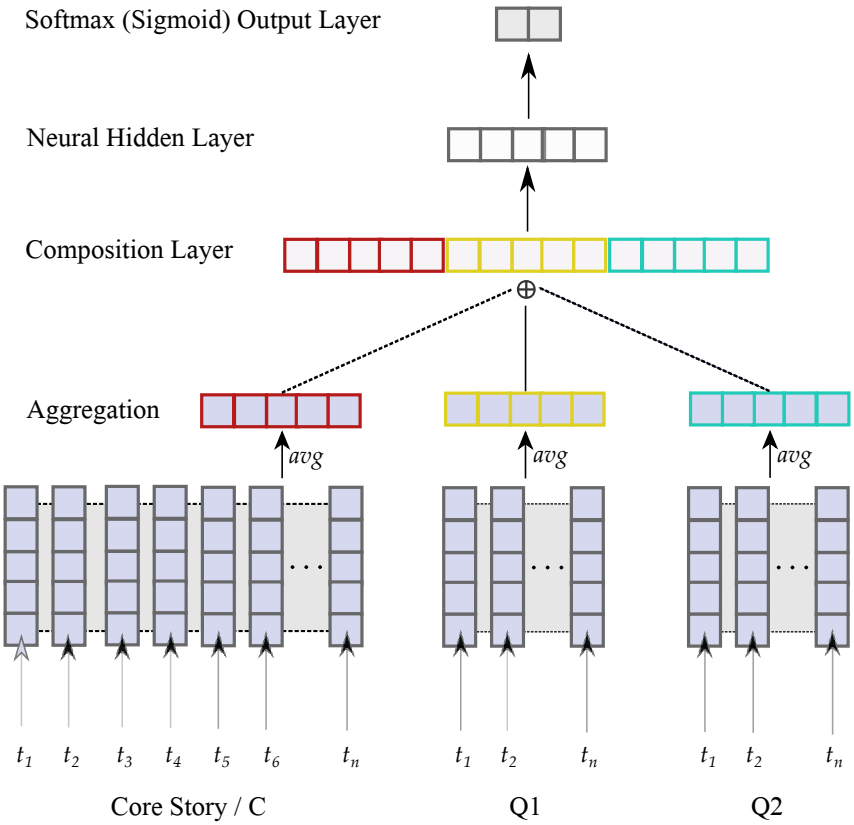

Figure 1: The proposed architecture for the Story Cloze Test. Depicted is a training instance consisting of three distributed word representation matrices for core story (C), quiz 1 (Q1) and quiz 2 (Q2), each component of varying length n. Note that either Q1 or Q2 is a wrong story ending. Matrices are first individually aggregated by average computation. Resulting vectors are then concatenated to form a composition unit which serves as input to the network with one hidden layer and binary output classification.

der of both events. The distinction between different implicit discourse senses are subtle nuances and are highly challenging to detect automatically; however, they are typical of the *ROCStories*, as almost no explicit discourse markers are present between the individual story sentences. Finally, note that our motivation for this approach is also related to the classical view of recognizing *textual entailment* which would treat correct and wrong endings as the entailing and contradicting hypotheses, respectively (Giampiccolo et al., 2007; Mostafazadeh et al., 2016a).

3.1 Training Instances

For the Story Cloze Test, we model a training instance as a triplet consisting of the four-sentence core story (C), a first quiz sentence (Q1) *and* a second quiz sentence (Q2) from which either Q1 or Q2 is the correct continuation of the story. Note that the original *ROCStories* contain only valid five-sentence sequences but the evaluation data requires a system to select from a pool of two alternatives. Therefore, for each single story in *ROCStories*, we randomly sample one negative (wrong) continuation Q_{wrong} from all last sentences, and generate two training instances with the following patterns:
$[\text{C}, \text{Q1}, \text{Q2}_{\text{wrong}}]$:Label_1,$[\text{C}, \text{Q1}_{\text{wrong}}, \text{Q2}]$:Label_2, where the label indicates the position of the correct quiz. Our motivation is to jointly learn core stories together with their true ending while at the same time discriminating them from semantically irrelevant continuations.

For each component in the triplet, we have experimented with a variety of different calculations in order to capture their idiosyncratic syntactic and semantic properties. We found the vector average over their respective words $\vec{v}^{avg} = \frac{1}{N}\sum_{i=1}^{N} E(t_i)$ to perform reasonably well, where N is the total number of tokens filling either of C, Q1 or Q2, respectively, resulting in three individual vector representations. Here, we define $E(\cdot)$ as an embedding function which maps a token t_i to its dis-

tributed representation, i.e., a precomputed vector of d dimensions. As distributed word representations, we chose out of the box vectors; GloVe vectors (Pennington et al., 2014), dependency-based word embeddings (Levy and Goldberg, 2014) and the pre-trained Google News vectors with $d = 300$ from *word2vec*[4] (Mikolov et al., 2013). Using the same tool, we also trained custom embeddings (bag-of-words and skip-gram) with 300 dimensions on the *ROCStories* corpus.[5]

3.2 Network Architecture

The feature construction process and the neural network architecture are depicted in Figure 1. The bottom part illustrates how tokens are mapped through three stacked embedding matrices for C, Q1 and Q2, each of dimensionality $\mathbb{R}^{d \times n}$. A second step applies the average aggregation and concatenates the so-obtained vectors $\vec{c}^{avg}$, $\vec{q_1}^{avg}$, $\vec{q_2}^{avg}$ (each $\vec{v}^{avg} \in \mathbb{R}^d$) into an overall composed story representation of dimensionality $\mathbb{R}^{3*d}$ which in turn serves as input to a feedforward neural network. The network is set up with one hidden layer and one sigmoid output layer for binary label classification for the position of the correct ending, i.e. first or second.

3.3 Implementational Details

The network is trained only on the *ROCStories* (and the negative training items), totaling approx. 200k training instances, over 30 iterations and 35 epochs with pretraining and a mini batch size of 120. All (hyper-)parameters are chosen and optimized on the validation set. We conduct data normalization, Xavier weight initialization (Glorot and Bengio, 2010) on the input layer, and employ rectified linear unit activation functions to both the composition layer and hidden layer with 220-250 nodes, and finally apply a sigmoid output layer for label classification. The learning rate is set to 0.04, l2 regularization = 0.0002 for penalizing network weights using the cross entropy error loss function. The network is trained using stochastic gradient descent and backpropagation implemented in the toolkit *deeplearning4j*.[6]

[4] https://code.google.com/p/word2vec/

[5] We remove punctuation symbols in all settings.

[6] https://deeplearning4j.org/

	Performance	
System	Validation	Test
DSSM	0.604	0.585
Narrative-Chains	0.510	0.494
Majority Class	0.514	0.513
Neural-ROCStoriesOnly	**0.629**	0.632
SVM-ManualLabels	–	**0.700**

Table 2: Performances (in % accuracy) on the validation and test sets of The Story Cloze Test.

4 Evaluation

We evaluate our model intrinsically on both validation and test set provided by the shared task organizers. As a reference, we also provide three baselines borrowed from Mostafazadeh et al. (2016a) at the time when the data set was released, namely the best-performing algorithms inspired by Huang et al. (2013, Deep Structured Semantic Model/DSSM) and Chambers and Jurafsky (2008, Narrative-Chains). Table 2 shows that correct endings appear almost equally often in either first or second position in the annotated data sets. The majority class is only significantly beaten by the DSSM model. Our approach (denoted by *Neural-ROCStoriesOnly*), however, can further improve upon the best system by an absolute increase in accuracy of 4.7%. Only the best configuration is shown and has been achieved with the 300-dimensional pre-trained Google News embeddings. Interestingly, the performance of the model on the test set is slightly better that on the validation set but also very similar which suggests that it is able to generalize well to unseen data and is not prone to overfitting training or validation data. A manual inspection of a subset of the misclassified items reveals that our neural recognizer is struggling to properly handle story continuations which change the underlying sentiment of the core story either towards negative or positive, e.g. *fail test, study hard → pass test*. In future work we plan to address this issue in closer detail.

A Note on the Evaluation & Training Procedure: Although the task has been stated differently, it stands to reason that one could exploit the tiny amount of hand-annotated data in the validation set directly to train a classifier. We have done so as a side experiment using as features the

same 900-dimensional composition layer embeddings from Section 3.2 and optimized a minimalist SVM classifier by 10-fold cross-validation, with feature and parameter selection on the validation set.[7] The final model achieves a test set accuracy of 70.02%, cf. *SVM-ManualLabels* in Table 2. Besides the relatively good performance obtained here, however, we want to emphasize that—when no hand-annotated labels for the correct position of the quizzes are available—the *Neural-ROCStories* approach introduced in Section 3 represents a promising and more generic framework for coherence learning, incorporating the plain text *ROCStories* as only source of information.

5 Conclusion & Outlook

In this paper, we have introduced a highly generic and resource-lean neural recognizer for modeling text coherence, which has been adapted to a designated data set—the *ROCStories* for modeling story continuations. Our approach is inspired by successful models for (implicit) discourse relation classification and only relies on the carefully tuned interaction of distributed word representations between story components.

An evaluation shows that state-of-the-art algorithms for script learning can be outperformed by our model. Future work should address the incorporation of linguistic knowledge into the currently rather rigid representations of the story sentences, including sentiment polarities or weighted syntactic dependencies (Schenk et al., 2016). Even though it has been claimed that the simpler feedforward neural networks do perform better in the discourse modeling task (Rutherford and Xue, 2016), it remains an open and challenging topic for future experiments on the *ROCStories*, whether *recurrent* architectures (Pichotta and Mooney, 2016) can have additional value towards a deeper story understanding.[8]

Acknowledgments

We want to thank Philip Schulz for constructive feedback regarding the neural network setup and the two anonymous reviewers for their very helpful remarks and insightful comments.

[7] We used *libsvm* (`https://www.csie.ntu.edu.tw/~cjlin/libsvm/`), RBF kernel, $c = 1.85$, $g = 0.63$.

[8] The code for this study is publicly available from the following URL: `http://www.acoli.informatik.uni-frankfurt.de/resources/`.

References

Nathanael Chambers and Daniel Jurafsky. 2008. Unsupervised Learning of Narrative Event Chains. In *ACL 2008, Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA*, pages 789–797.

Matthew Gerber and Joyce Chai. 2012. Semantic Role Labeling of Implicit Arguments for Nominal Predicates. *Comput. Linguist.*, 38(4):755–798, December.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The Third PASCAL Recognizing Textual Entailment Challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, RTE '07, pages 1–9, Stroudsburg, PA, USA. Association for Computational Linguistics.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS10). Society for Artificial Intelligence and Statistics*.

Eduard H. Hovy. 2006. Learning by Reading: An Experiment in Text Analysis. In Petr Sojka, Ivan Kopecek, and Karel Pala, editors, *TSD*, volume 4188 of *Lecture Notes in Computer Science*, pages 3–12. Springer.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning Deep Structured Semantic Models for Web Search Using Clickthrough Data. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, CIKM '13, pages 2333–2338, New York, NY, USA. ACM.

Omer Levy and Yoav Goldberg. 2014. Dependency-Based Word Embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 302–308.

Todor Mihaylov and Anette Frank. 2016. Discourse Relation Sense Classification Using Cross-argument Semantic Similarity Based on Word Embeddings. In *Proceedings of the CoNLL-16 shared task*, pages 100–107. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of Workshop at International Conference on Learning Representations*.

Paramita Mirza and Sara Tonelli. 2016. CATENA: causal and temporal relation extraction from natural language texts. In *COLING 2016, 26th International Conference on Computational Linguistics,*

Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan, pages 64–75.

Ashutosh Modi and Ivan Titov. 2014. Learning Semantic Script Knowledge with Event Embeddings. In *Proceedings of the 2nd International Conference on Learning Representations (Workshop track)*, Banff, Canada.

Raymond J. Mooney and Gerald DeJong. 1985. Learning Schemata for Natural Language Processing. In *Proceedings of the 9th International Joint Conference on Artificial Intelligence. Los Angeles, CA, USA, August 1985*, pages 681–687.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016a. A Corpus and Cloze Evaluation for Deeper Understanding of Commonsense Stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California, June. Association for Computational Linguistics.

Nasrin Mostafazadeh, Lucy Vanderwende, Wen-tau Yih, Pushmeet Kohli, and James Allen. 2016b. Story Cloze Evaluator: Vector Space Representation Evaluation by Predicting What Happens Next. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 24–29, Berlin, Germany, August. Association for Computational Linguistics.

Kiem-Hieu Nguyen, Xavier Tannier, Olivier Ferret, and Romaric Besançon. 2015. Generative Event Schema Induction with Entity Disambiguation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 188–197.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Karl Pichotta and Raymond J. Mooney. 2016. Using sentence-level LSTM language models for script inference. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings, 6th International Conference on Language Resources and Evaluation*, pages 2961–2968, Marrakech, Morocco.

Michaela Regneri, Alexander Koller, and Manfred Pinkal. 2010. Learning script knowledge from web experiments. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, Uppsala.

Attapol Rutherford and Nianwen Xue. 2016. Robust Non-Explicit Neural Discourse Parser in English and Chinese. In *Proceedings of the CoNLL-16 shared task*, pages 55–59. Association for Computational Linguistics.

Roger C. Schank and Robert P. Abelson. 1977. *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*. L. Erlbaum, Hillsdale, NJ.

Niko Schenk and Christian Chiarcos. 2016. Unsupervised learning of prototypical fillers for implicit semantic role labeling. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 1473–1479.

Niko Schenk, Christian Chiarcos, Kathrin Donandt, Samuel Rönnqvist, Evgeny Stepanov, and Giuseppe Riccardi. 2016. Do We Really Need All Those Rich Linguistic Features? A Neural Network-Based Approach to Implicit Sense Labeling. In *Proceedings of the CoNLL-16 shared task*, pages 41–49. Association for Computational Linguistics.

Naushad UzZaman and James F Allen. 2010. Extracting Events and Temporal Expressions from Text. In *Semantic Computing (ICSC), 2010 IEEE Fourth International Conference on*, pages 1–8. IEEE.

Jianxiang Wang and Man Lan. 2016. Two End-to-end Shallow Discourse Parsers for English and Chinese in CoNLL-2016 Shared Task. In *Proceedings of the CoNLL-16 shared task*, pages 33–40. Association for Computational Linguistics.

Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Bonnie Webber, Attapol Rutherford, Chuan Wang, and Hongmin Wang. 2016. The CoNLL-2016 Shared Task on Shallow Discourse Parsing. In *Proceedings of the Twentieth Conference on Computational Natural Language Learning - Shared Task*, Berlin, Germany, August. Association for Computational Linguistics.