In Proceedings of the Workshop on Uphill Battles in Language Processing (UBLP) at EMNLP 2016, Austin, TX, November 2016.

Statistical Script Learning with Recurrent Neural Networks

Karl Pichotta and **Raymond J. Mooney**
Department of Computer Science
The University of Texas at Austin
{pichotta,mooney}@cs.utexas.edu

Abstract

We describe some of our recent efforts in learning statistical models of co-occurring events from large text corpora using Recurrent Neural Networks.

1 Introduction

Natural language *scripts* are structured models of stereotypical sequences of events used for document understanding. For example, a script model may encode the information that from *Smith landed in Beijing*, one may presumably infer *Smith flew in an airplane to Beijing*, *Smith got off the plane at the Beijing airport*, etc. The world knowledge encoded in such event co-occurrence models is intuitively useful for a number of semantic tasks, including Question Answering, Coreference Resolution, Discourse Parsing, and Semantic Role Labeling.

Script learning and inference date back to AI research from the 1970s, in particular the seminal work of Schank and Abelson (1977). In this work, events are formalized as quite complex hand-encoded structures, and the structures encoding event co-occurrence are non-statistical and hand-crafted based on appeals to the intuitions of the knowledge engineer. Mooney and DeJong (1985) give an early non-statistical method of automatically inducing models of co-occurring events from documents, but their methods are non-statistical.

There is a growing body of more recent work investigating methods of learning statistical models of event sequences from large corpora of raw text. These methods admit scaling models up to be much larger than hand-engineered ones, while being more robust to noise than automatically learned non-statistical models. Chambers and Jurafsky (2008) describe a statistical co-occurrence model of (verb, dependency) pair events that is trained on a large corpus of documents and can be used to infer implicit events from text. A number of other systems following similar paradigm have also been proposed (Chambers and Jurafsky, 2009; Jans et al., 2012; Rudinger et al., 2015). These approaches achieve generalizability and computational tractability on large corpora, but do so at the expense of decreased representational complexity: in place of the rich event structures found in Schank and Abelson (1977), these systems model and infer structurally simpler events.

In this extended abstract, we will briefly summarize a number of statistical script-related systems we have described in previous publications (Pichotta and Mooney, 2016a; Pichotta and Mooney, 2016b), place them within the broader context of related research, and remark on future directions for research.

2 Methods and results

In Pichotta and Mooney (2016a), we present a system that uses Long Short-Term Memory (LSTM) Recurrent Neural Nets (RNNs) (Hochreiter and Schmidhuber, 1997) to model sequences of events. In this work, events are defined to be verbs with information about their syntactic arguments (either the noun identity of the head of an NP phrase relating to the verb, the entity identity according to a coreference resolution engine, or both). For example, the sentence *Smith got off the plane at the Beijing air-*

port would be represented as (get_off, smith, plane, (at, airport)). This event representation was investigated in Pichotta and Mooney (2014) in the context of count-based co-occurrence models. Balasubramanian et al. (2013), Modi and Titov (2014), and Granroth-Wilding and Clark (2016) describe systems for related tasks with similar event formulations.

In Pichotta and Mooney (2016a), we train an RNN sequence model by inputting one component of an event tuple at each timestep, representing sequences of events as sequences of event components. Standard methods for learning RNN sequence models are applied to learning statistical models of sequences of event components. To infer probable unobserved events from documents, we input observed document events in sequence, one event component per timestep, and then search over the components of a next event to be inferred using a beam search. That is, the structured prediction problem of event inference is reduced to searching over probable RNN output sequences. This is similar in spirit to a number of recent systems using RNN models for structured prediction (Vinyals et al., 2015; Luong et al., 2016; Dong and Lapata, 2016).

While the count-based event co-occurrence system we investigated in Pichotta and Mooney (2014) treats events as atomic—for example, *the plane flew* and *the plane flew over land* are unrelated events with completely independent statistics—this method decomposes events into components, and the two occurrences of the verb *flew* in the above examples have the same representation. Further, a low-dimensional embedding is learned for every event component, so *flew* and *soared* can get similar representations, allowing for generalization beyond the lexical level. Given the combinatorial number of event types,[1] decomposing structured events into components, rather than treating them as atomic, is crucial to scaling up the number of events a script system is capable of inferring. In fact, the system presented in Pichotta and Mooney (2014) does not use noun information about event arguments for this reason, instead using only coreference-based entity information.

System	Recall at 25	Human
Unigram	0.101	-
Bigram	0.124	2.21
LSTM	0.152	3.67

Table 1: Next event prediction results in Pichotta and Mooney (2016a). Partial credit is out of 1, and human evaluations are out of 5 (higher is better for both). More results can be found in the paper.

Table 1 gives results comparing a naive baseline ("Unigram," which always deterministically guesses the most common events), a co-occurrence based baseline ("Bigram," similar to the system of Pichotta and Mooney (2014)) and the LSTM system. The metric "Recall at 25" holds an event out from a test document and judges a system by its recall of the gold-standard event in its list of top 25 inferences. The "Human" metric is average crowdsourced judgments of inferences on a scale from 0 to 5, with some post hoc quality-control filtering applied. The LSTM system outperforms the other systems. More results and details can be found in Pichotta and Mooney (2016a).

These results indicate that RNN sequence models can be fruitfully applied to the task of predicting held-out events from text, by modeling and inferring events comprising a subset of the document's syntactic dependency structure. This naturally raises the question of to what extent, within the current regime of event-inferring systems trained on documents, explicit syntactic dependencies are necessary as a mediating representation. In Pichotta and Mooney (2016b), we compare event RNN models, of the sort described above, with RNN models that operate at the raw text level. In particular, we investigate the performance of a text-level sentence encoder/decoder similar to the skip-thought system of Kiros et al. (2015) on the task. In this setup, during inference, instead of encoding events and decoding events, we encode raw text, decode raw text, and then parse inferred text to get its dependency structure.[2] This system does not obviously encode event co-occurrence structure in the way that the

[1] With a vocabulary of V verb types, N noun types, P preposition types, and event tuples of arity k, there are about VPN^{k-1} event types. For $V = N = 10000$, $P = 50$, and $k = 4$, this is 5×10^{17}.

[2] We use the Stanford dependency parser (Socher et al., 2013).

previous one does, but can still in principle infer implicit events from text, and does not require a parser (and can be therefore be used for low-resource languages).

System	Accuracy	BLEU	1G P
Unigram	0.002	-	-
Copy/paste	-	1.88	22.6
Event LSTM	0.023	0.34	19.9
Text LSTM	0.020	5.20	30.9

Table 2: Prediction results in Pichotta and Mooney (2016b). More results can be found in the paper.

Table 2 gives a subset of results from Pichotta and Mooney (2016b), comparing an event LSTM with a text LSTM. The "Copy/paste" baseline deterministically predicts a sentence as its own successor. The "Accuracy" metric measures what percentage of argmax inferences were equal to the gold-standard held-out event. The "BLEU" column gives BLEU scores (Papineni et al., 2002) for raw text inferred by systems (either directly, or via an intermediate text-generation step in the case of the Event LSTM output). The "1G P" column gives unigram precision against the gold standard, which is one of the components of BLEU. Figure 1, reproduced from Pichotta and Mooney (2016b), gives some example next-sentence predictions. Despite the fact that it is very difficult to predict the next sentence in natural text, the text-level encoder/decoder system is capable of learning learning some aspects of event co-occurrence structure in documents.

These results indicate that modeling text directly does not appear to appreciably harm the ability to infer held-out events, and greatly helps in inferring held-out text describing those events.

3 Related Work

There are a number of related lines of research investigating different approaches to statistically modeling event co-occurrence. There is, first of all, a body of work investigating systems which infer events from text (including the above work). Chambers and Jurafsky (2008) give a method of modeling and inferring simple (verb, dependency) pair-events. Jans et al. (2012) describe a model of the same sorts of events which gives superior performance on the task of held-out event prediction; Rudinger et al. (2015) follow this line of inquiry, concluding that the task of inferring held-out (verb, dependency) pairs from documents is best handled as a language modeling task.

Second, there is a body of work focusing on automatically inducing structured collections of events (Chambers, 2013; Cheung et al., 2013; Nguyen et al., 2015; Ferraro and Van Durme, 2016), typically motivated by Information Extraction tasks.

Third, there is a body of work investigating high-precision models of situations as they occur in the world (as opposed to how they are described in text) from smaller corpora of event sequences (Regneri et al., 2010; Li et al., 2012; Frermann et al., 2014; Orr et al., 2014).

Fourth, there is a recent body of work investigating the automatic induction of event structure in different modalities. Kim and Xing (2014) give a method of modeling sequences of images from ordered photo collections on the web, allowing them to perform, among other things, sequential image prediction. Huang et al. (2016) describe a new dataset of photos in temporal sequence scraped from web albums, along with crowdsourced story-like descriptions of the sequences (and methods for automatically generating the latter from the former). Bosselut et al. (2016) describe a system which learns a model of prototypical event co-occurrence from online photo albums with their natural language captions. Incorporating learned event co-occurrence structure from large-scale natural datasets of different modalities could be an exciting line of future research.

Finally, there are a number of alternative ways of evaluating learned script models that have been proposed. Motivated by the shortcomings of evaluation via held-out event inference, Mostafazadeh et al. (2016) recently introduced a corpus of crowdsourced short stories with plausible "impostor" endings alongside the real endings; script systems can be evaluated on this corpus by their ability to discriminate the real ending from the impostor one. This corpus is not large enough to train a script system, but can be used to evaluate a pretrained one. Hard coreference resolution problems (so-called "Winograd schema challenge" problems (Rahman and Ng, 2012)) provide another possible

Input:	As of October 1 , 2008 , ⟨OOV⟩ changed its company name to Panasonic Corporation.
Gold:	⟨OOV⟩ products that were branded "National" in Japan are currently marketed under the "Panasonic" brand.
Predicted:	The company's name is now ⟨OOV⟩.
Input:	White died two days after Curly Bill shot him.
Gold:	Before dying, White testified that he thought the pistol had accidentally discharged and that he did not believe that Curly Bill shot him on purpose.
Predicted:	He was buried at ⟨OOV⟩ Cemetery.
Input:	The foundation stone was laid in 1867.
Gold:	The members of the predominantly Irish working class parish managed to save £700 towards construction, a large sum at the time.
Predicted:	The ⟨OOV⟩ was founded in the early 20th century.
Input:	Soldiers arrive to tell him that ⟨OOV⟩ has been seen in camp and they call for his capture and death.
Gold:	⟨OOV⟩ agrees .
Predicted:	⟨OOV⟩ is killed by the ⟨OOV⟩.

Figure 1: Examples of next-sentence text predictions, reproduced from Pichotta and Mooney (2016b). ⟨OOV⟩ is the out-of-vocabulary pseudo-token, which frequently replaces proper names.

alternative evaluation for script systems.

4 Future Work and Conclusion

The methods described above were motivated by the utility of event inferences based on world knowledge, but, in order to leverage large text corpora, actually model documents rather than scenarios in the world *per se*. That is, this work operates under the assumption that modeling event sequences in documents is a useful proxy for modeling event sequences in the world. As mentioned in Section 3, incorporating information from multiple modalities is one possible approach to bridging this gap. Incorporating learned script systems into other useful extrinsic evaluations, for example coreference resolution or question-answering, is another.

For the task of inferring verbs and arguments explicitly present in documents, as presented above, we have described some evidence that, in the context of standard RNN training setups, modeling raw text yields fairly comparable performance to explicitly modeling syntactically mediated events. The extent to which this is true for other extrinsic tasks is an empirical issue that we are currently exploring. Further, the extent to which representations of more complex event properties (such as those hand-encoded in Schank and Abelson (1977)) can be learned automatically (or happen to be encoded in the learned embeddings and dynamics of neural script models) is an open question.

Acknowledgments

This research was supported in part by the DARPA DEFT program under AFRL grant FA8750-13-2-0026.

References

Niranjan Balasubramanian, Stephen Soderland, Mausam, and Oren Etzioni. 2013. Generating coherent event schemas at scale. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP-2013)*.

Antoine Bosselut, Jianfu Chen, David Warren, Hannaneh Hajishirzi, and Yejin Choi. 2016. Learning prototypical event structure from photo albums. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL-16)*.

Nathanael Chambers and Daniel Jurafsky. 2008. Unsupervised learning of narrative event chains. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL-08)*, pages 789–797.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics (ACL-09)*, pages 602–610.

Nathanael Chambers. 2013. Event schema induction with a probabilistic entity-driven model. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP-2013)*.

Jackie Chi Kit Cheung, Hoifung Poon, and Lucy Vanderwende. 2013. Probabilistic frame induction. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-13)*.

Li Dong and Mirella Lapata. 2016. Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL-16)*.

Francis Ferraro and Benjamin Van Durme. 2016. A unified Bayesian model of scripts, frames and language. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-16)*.

Lea Frermann, Ivan Titov, and Manfred Pinkal. 2014. A hierarchical Bayesian model for unsupervised induction of script knowledge. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL-14)*, pages 49–57.

Mark Granroth-Wilding and Stephen Clark. 2016. What happens next? Event prediction using a compositional neural network model. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-16)*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Ting-Hao K. Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Aishwarya Agrawal, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, Larry Zitnick, Devi Parikh, Lucy Vanderwende, Michel Galley, and Margaret Mitchell. 2016. Visual storytelling. In *Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-16)*.

Bram Jans, Steven Bethard, Ivan Vulić, and Marie Francine Moens. 2012. Skip n-grams and ranking functions for predicting script events. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL-12)*, pages 336–344.

Gunhee Kim and Eric P. Xing. 2014. Reconstructing storyline graphs for image recommendation from web community photos. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR-14)*.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. In *Proceedings of the 29th Annual Conference on Neural Information Processing Systems (NIPS-15)*.

Boyang Li, Stephen Lee-Urban, Darren Scott Appling, and Mark O Riedl. 2012. Crowdsourcing narrative intelligence. *Advances in Cognitive Systems*, 2:25–42.

Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. In *Proceedings of the 4th International Conference on Learning Representations (ICLR-16)*.

Ashutosh Modi and Ivan Titov. 2014. Inducing neural models of script knowledge. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning (CoNLL-2014)*, Baltimore, MD, USA.

Raymond J. Mooney and Gerald F. DeJong. 1985. Learning schemata for natural language processing. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85)*, pages 681–687.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. 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 (NAACL-16)*.

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 (ACL-15)*.

J Walker Orr, Prasad Tadepalli, Janardhan Rao Doppa, Xiaoli Fern, and Thomas G Dietterich. 2014. Learning scripts as Hidden Markov Models. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI-14)*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-02)*, pages 311–318.

Karl Pichotta and Raymond J. Mooney. 2014. Statistical script learning with multi-argument events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*, pages 220–229.

Karl Pichotta and Raymond J. Mooney. 2016a. Learning statistical scripts with LSTM recurrent neural networks. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-16)*.

Karl Pichotta and Raymond J. Mooney. 2016b. Using sentence-level LSTM language models for script inference. In *Proceedings of the 54th Annual Meeting of*

the Association for Computational Linguistics (ACL-16), Berlin, Germany.

Altaf Rahman and Vincent Ng. 2012. Resolving complex cases of definite pronouns: the Winograd schema challenge. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL-12)*, pages 777–789.

Michaela Regneri, Alexander Koller, and Manfred Pinkal. 2010. Learning script knowledge with web experiments. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL-10)*, Uppsala, Sweden, July.

Rachel Rudinger, Pushpendre Rastogi, Francis Ferraro, and Benjamin Van Durme. 2015. Script induction as language modeling. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP-15)*.

Roger C. Schank and Robert P. Abelson. 1977. *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures*. Lawrence Erlbaum and Associates.

Richard Socher, John Bauer, Christopher D. Manning, and Andrew Y. Ng. 2013. Parsing with compositional vector grammars. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL-13)*.

Oriol Vinyals, Łukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. 2015. Grammar as a foreign language. In *Proceedings of the 29th Annual Conference on Neural Information Processing Systems (NIPS-15)*, pages 2755–2763.