Improving sentence compression by learning to predict gaze

Sigrid Klerke
University of Copenhagen
skl@hum.ku.dk

Yoav Goldberg
Bar-Ilan University
yoav.goldberg@gmail.com

Anders Søgaard
University of Copenhagen
soegaard@hum.ku.dk

Abstract

We show how eye-tracking corpora can be used to improve sentence compression models, presenting a novel multi-task learning algorithm based on multi-layer LSTMs. We obtain performance competitive with or better than state-of-the-art approaches.

1 Introduction

Sentence compression is a basic operation in text simplification which has the potential to improve statistical machine translation and automatic summarization (Berg-Kirkpatrick et al., 2011; Klerke et al., 2015), as well as helping poor readers in need of assistive technologies (Canning et al., 2000). This work suggests using eye-tracking recordings for improving sentence compression for text simplification systems and is motivated by two observations: (i) *Sentence compression is the task of automatically making sentences easier to process by shortening them.* (ii) *Eye-tracking measures* such as first-pass reading time and time spent on regressions, i.e., during second and later passes over the text, *are known to correlate with perceived text difficulty* (Rayner et al., 2012).

These two observations recently lead Klerke et al. (2015) to suggest using eye-tracking measures as metrics in text simplification. We go beyond this by suggesting that eye-tracking recordings can be used to induce better models for sentence compression for text simplification. Specifically, we show how to use existing eye-tracking recordings to improve the induction of Long Short-Term Memory models (LSTMs) for sentence compression.

Our proposed model *does not require* that the gaze data and the compression data come from the same source. Indeed, in this work we use gaze data from readers of the Dundee Corpus to improve sentence compression results on several datasets. While not explored here, an intriguing potential of this work is in deriving sentence simplification models that are personalized for individual users, based on their reading behavior.

Several approaches to sentence compression have been proposed, from noisy channel models (Knight and Marcu, 2002) over conditional random fields (Elming et al., 2013) to tree-to-tree machine translation models (Woodsend and Lapata, 2011). More recently, Filippova et al. (2015) successfully used LSTMs for sentence compression on a large scale parallel dataset. We do not review the literature here, and only compare to Filippova et al. (2015).

Our contributions

- We present a novel multi-task learning approach to sentence compression using labelled data for sentence compression and a disjoint eye-tracking corpus.
- Our method is fully competitive with state-of-the-art across three corpora.
- Our code is made publicly available at `https://bitbucket.org/soegaard/gaze-mtl16`.

2 Gaze during reading

Readers fixate longer at rare words, words that are semantically ambiguous, and words that are mor-

Proceedings of NAACL-HLT 2016, pages 1528–1533,
San Diego, California, June 12-17, 2016.

phologically complex (Rayner et al., 2012). These are also words that are likely to be replaced with simpler ones in sentence simplification, but it is not clear that they are words that would necessarily be removed in the context of sentence compression.

Demberg and Keller (2008) show that syntactic complexity (measured as dependency locality) is also an important predictor of reading time. Phrases that are often removed in sentence compression—like fronted phrases, parentheticals, floating quantifiers, etc.—are often associated with non-local dependencies. Also, there is evidence that people are more likely to fixate on the first word in a constituent than on its second word (Hyönä and Pollatsek, 2000). Being able to identify constituent borders is important for sentence compression, and reading fixation data may help our model learn a representation of our data that makes it easy to identify constituent boundaries.

In the experiments below, we learn models to predict the first pass duration of word fixations and the total duration of regressions to a word. These two measures constitute a perfect separation of the total reading time of each word split between the first pass and subsequent passes. Both measures are described below. They are both discretized into six bins as follows with only non-zero values contributing to the calculation of the standard deviation (SD):

0: measure = 0 or
1: measure $<$ 1 SD below reader's average or
2: measure $<$ .5 SD below reader's average or
3: measure $<$ .5 above reader's average or
4: measure $>$ .5 SD above reader's average or
5: measure $>$ 1 SD above reader's average

First pass duration measures the total time spent reading a word first time it is fixated, including any immediately following re-fixations of the same word. This measure correlates with word length, frequency and ambiguity because long words are likely to attract several fixations in a row unless they are particularly easily predicted or recognized. This effect arises because long words are less likely to fit inside the fovea of the eye. Note that for this measure the value 0 indicates that the word was not fixated by this reader.

Words	FIRST PASS	REGRESSIONS
Are	4	4
tourists	2	0
enticed	3	0
by	4	0
these	2	0
attractions	3	0
threatening	3	3
their	5	0
very	3	3
existence	3	5
?	3	5

Figure 1: Example sentence from the Dundee Corpus

Regression duration measures the total time spent fixating a word after the gaze has already left it once. This measure belongs to the group of late measures, i.e., measures that are sensitive to the later cognitive processing stages including interpretation and integration of already decoded words. Since the reader by definition has already had a chance to recognize the word, regressions are associated with semantic confusion and contradiction, incongruence and syntactic complexity, as famously experienced in garden path sentences. For this measure the value 0 indicates that the word was read at most once by this reader.

See Table 1 for an example of first pass duration and regression duration annotations for one reader and sentence.

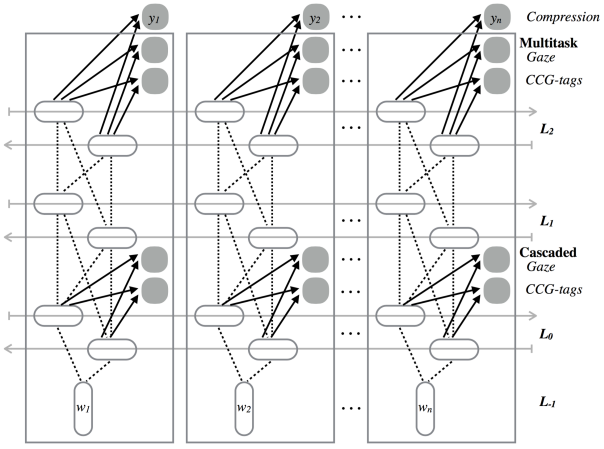

Figure 2: Multitask and cascaded bi-LSTMs for sentence compression. Layer L_{-1} contain pre-trained embeddings. Gaze prediction and CCG-tag prediction are auxiliary training tasks, and loss on all tasks are propagated back to layer L_0.

3 Sentence compression using multi-task deep bi-LSTMs

Most recent approaches to sentence compression make use of syntactic analysis, either by operating directly on trees (Riezler et al., 2003; Nomoto, 2007; Filippova and Strube, 2008; Cohn and Lapata, 2008; Cohn and Lapata, 2009) or by incorporating syntactic information in their model (McDonald, 2006; Clarke and Lapata, 2008). Recently, however, Filippova et al. (2015) presented an approach to sentence compression using LSTMs with word embeddings, but without syntactic features. We introduce a third way of using syntactic annotation by jointly learning a sequence model for predicting CCG supertags, in addition to our gaze and compression models.

Bi-directional recurrent neural networks (bi-RNNs) read in sequences in both regular and reversed order, enabling conditioning predictions on both left and right context. In the forward pass, we run the input data through an embedding layer and compute the predictions of the forward and backward states at layers $0, 1, \ldots$, until we compute the softmax predictions for word i based on a linear transformation of the concatenation of the of standard and reverse RNN outputs for location i. We then calculate the objective function derivative for the sequence using cross-entropy (logistic loss) and use backpropagation to calculate gradients and update the weights accordingly. A deep bi-RNN or k-layered bi-RNN is composed of k bi-RNNs that feed into each other such that the output of the ith RNN is the input of the $i + 1$th RNN. LSTMs (Hochreiter and Schmidhuber, 1997) replace the cells of RNNs with LSTM cells, in which multiplicative gate units learn to open and close access to the error signal.

Bi-LSTMs have already been used for fine-grained sentiment analysis (Liu et al., 2015), syntactic chunking (Huang et al., 2015), and semantic role labeling (Zhou and Xu, 2015). These and other recent applications of bi-LSTMs were constructed for solving a single task in isolation, however. We instead train deep bi-LSTMs to solve additional tasks to sentence compression, namely CCG-tagging and gaze prediction, using the additional tasks to regularize our sentence compression model.

Specifically, we use bi-LSTMs with three layers. Our baseline model is simply this three-layered model trained to predict compressions (encoded as label sequences), and we consider two extensions thereof as illustrated in Figure 2. Our first extension, MULTI-TASK-LSTM, includes the gaze prediction task during training, with a separate logistic regression classifier for this purpose; and the other, CASCADED-LSTM, predicts gaze measures from the inner layer. Our second extension, which is superior to our first, is basically a one-layer bi-LSTM for predicting reading fixations with a two-layer bi-LSTM on top for predicting sentence compressions.

At each step in the training process of MULTI-TASK-LSTMand CASCADED-LSTM, we choose a random task, followed by a random training instance of this task. We use the deep LSTM to predict a label sequence, suffer a loss with respect to the true labels, and update the model parameters. In CASCADED-LSTM, the update for an instance of CCG super tagging or gaze prediction only affects the parameters of the inner LSTM layer.

Both MULTI-TASK-LSTM and CASCADED-LSTM do multi-task learning (Caruana, 1993). In multi-task learning, the induction of a model for one task is used as a regularizer on the induction of a model for another task. Caruana (1993) did multi-task learning by doing parameter sharing across several deep networks, letting them share hidden layers; a technique also used by Collobert et al. (2011) for various NLP tasks. These models train task-specific classifiers on the output of deep networks (informed by the task-specific losses). We extend their models by moving to sequence prediction and allowing the task-specific sequence models to also be deep models.

4 Experiments

4.1 Gaze data

We use the Dundee Corpus (Kennedy et al., 2003) as our eye-tracking corpus with tokenization and measures similar to the Dundee Treebank (Barrett et al., 2015). The corpus contains eye-tracking recordings of ten native English-speaking subjects reading 20 newspaper articles from *The Independent*. We use data from nine subjects for training and one subject for development. We do not evaluate the gaze prediction because the task is only included as a way of regularizing the compression model.

S:	Regulators Friday shut down a small Florida bank, bringing to 119 the number of US bank failures this year amid mounting loan defaults.
T:	Regulators shut down a small Florida bank
S:	Intel would be building car batteries, expanding its business beyond its core strength, the company said in a statement.
T:	Intel would be building car batteries

Table 1: Example compressions from the GOOGLE dataset. S is the source sentence, and T is the target compression.

	Sents	Sent.len	Type/token	Del.rate
TRAINING				
ZIFF-DAVIS	1000	20	0.22	0.59
BROADCAST	880	20	0.21	0.27
GOOGLE	8000	24	0.17	0.87
TEST				
ZIFF-DAVIS	32	21	0.55	0.47
BROADCAST	412	19	0.27	0.29
GOOGLE	1000	25	0.42	0.87

Table 2: Dataset characteristics. Sentence length is for source sentences.

4.2 Compression data

We use three different sentence compression datasets, ZIFF-DAVIS (Knight and Marcu, 2002), BROADCAST (Clarke and Lapata, 2006), and the publically available subset of GOOGLE (Filippova et al., 2015). The first two consist of manually compressed newswire text in English, while the third is built heuristically from pairs of headlines and first sentences from newswire, resulting in the most aggressive compressions, as exemplified in Table 1. We present the dataset characteristics in Table 2. We use the datasets as released by the authors and do not apply any additional pre-processing. The CCG supertagging data comes from CCGbank,[1] and we use sections 0-18 for training and section 19 for development.

4.3 Baselines and system

Both the baseline and our systems are three-layer bi-LSTM models trained for 30 iterations with pre-trained (SENNA) embeddings. The input and hidden layers are 50 dimensions, and at the output layer we predict sequences of two labels, indicating whether to delete the labeled word or not. Our baseline (BASELINE-LSTM) is a multi-task learning bi-LSTM predicting both CCG supertags and sentence compression (word deletion) at the outer layer. Our first extension is MULTITASK-LSTM predicting CCG supertags, sentence compression, and reading measures from the outer layer. CASCADED-LSTM, on the other hand, predicts CCG supertags and reading measures from the initial layer, and sentence compression at the outer layer.

4.4 Results and discussion

Our results are presented in Table 3. We observe that across all three datasets, including all three annotations of BROADCAST, gaze features lead to improvements over our baseline 3-layer bi-LSTM. Also, CASCADED-LSTM is consistently better than MULTITASK-LSTM. Our models are fully competitive with state-of-the-art models. For example, the best model in Elming et al. (2013) achieves 0.7207 on ZIFF-DAVIS, Clarke and Lapata (2008) achieves 0.7509 on BROADCAST,[2] and the LSTM model in Filippova et al. (2015) achieves 0.80 on GOOGLE with much more training data. The high numbers on the small subset of GOOGLE reflects that newswire headlines tend to have a fairly predictable relation to

[1] http://groups.inf.ed.ac.uk/ccg/

[2] On a "randomly selected" annotator; unfortunately, they do not say which. James Clarke (p.c) does not remember which annotator they used.

LSTM	Gaze	ZIFF-DAVIS	BROADCAST			GOOGLE
Baseline		0.5668	0.7386	0.7980	0.6802	0.7980
Multitask	FP	0.6416	0.7413	0.8050	0.6878	0.8028
	REGR.	0.7025	0.7368	0.7979	0.6708	0.8016
Cascaded	FP	0.6732	**0.7519**	0.8189	**0.7012**	**0.8097**
	REGR.	**0.7418**	0.7477	**0.8217**	0.6944	0.8048

Table 3: Results (F_1). For all three datasets, the inclusion of gaze measures (first pass duration (FP) and regression duration (Regr.)) leads to improvements over the baseline. All models include CCG-supertagging as an auxiliary task. Note that BROADCASTwas annotated by three annotators. The three columns are, from left to right, results on annotators 1–3.

the first sentence. With the harder datasets, the impact of the gaze information becomes stronger, consistently favouring the cascaded architecture, and with improvements using both first pass duration and regression duration, the late measure associated with interpretation of content. Our results indicate that multi-task learning can help us take advantage of inherently noisy human processing data across tasks and thereby maybe reduce the need for task-specific data collection.

Acknowledgments

Yoav Goldberg was supported by the Israeli Science Foundation Grant No. 1555/15. Anders Søgaard was supported by ERC Starting Grant No. 313695. Thanks to Joachim Bingel and Maria Barrett for preparing data and for helpful discussions, and to the anonymous reviewers for their suggestions for improving the paper.

References

Maria Barrett, Željko Agić, and Anders Søgaard. 2015. The dundee treebank. In *The 14th International Workshop on Treebanks and Linguistic Theories (TLT 14)*.

Taylor Berg-Kirkpatrick, Dan Gillick, and Dan Klein. 2011. Jointly learning to extract and compress. In *Proceedings of ACL*.

Y. Canning, J. Tait, J. Archibald, and R. Crawley. 2000. *Cohesive generation of syntactically simplified newspaper text*. Springer.

Rich Caruana. 1993. Multitask learning: a knowledge-based source of inductive bias. In *ICML*.

James Clarke and Mirella Lapata. 2006. Constraint-based sentence compression an integer programming approach. In *COLING*.

James Clarke and Mirella Lapata. 2008. Global inference for sentence compression: An integer linear programming approach. *Journal of Artificial Intelligence Research*, pages 399–429.

Trevor Cohn and Mirella Lapata. 2008. Sentence compression beyond word deletion. In *COLING*.

Trevor Cohn and Mirella Lapata. 2009. Sentence compression as tree transduction. *Journal of Artificial Intelligence Research*, pages 637–674.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Vera Demberg and Frank Keller. 2008. Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, 109:193–210.

Jakob Elming, Anders Johannsen, Sigrid Klerke, Emanuele Lapponi, Héctor Martínez Alonso, and Anders Søgaard. 2013. Down-stream effects of tree-to-dependency conversions. In *NAACL*.

Katja Filippova and Michael Strube. 2008. Dependency tree based sentence compression. In *Proceedings of the Fifth International Natural Language Generation Conference*.

Katja Filippova, Enrique Alfonseca, Carlos A Colmenares, Lukasz Kaiser, and Oriol Vinyals. 2015. Sentence compression by deletion with LSTMs. In *EMNLP*.

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

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Jukka Hyönä and Alexander Pollatsek. 2000. Processing of finnish compound words in reading. *Reading as a perceptual process*, pages 65–87.

Alan Kennedy, Robin Hill, and Joël Pynte. 2003. The dundee corpus. In *ECEM*.

Sigrid Klerke, Sheila Castilho, Maria Barrett, and Anders Søgaard. 2015. Reading metrics for estimating task efficiency with mt output. In *EMNLP Workshop on Cognitive Aspects of Computational Language Learning*.

Kevin Knight and Daniel Marcu. 2002. Summarization beyond sentence extraction: a probabilistic approach to sentence compression. *Artificial Intelligence*, 139:91–107.

Pengfei Liu, Shafiq Joty, and Helen Meng. 2015. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *EMNLP*.

Ryan T McDonald. 2006. Discriminative sentence compression with soft syntactic evidence. In *EACL*.

Tadashi Nomoto. 2007. Discriminative sentence compression with conditional random fields. *Information Processing and Management: an International Journal*, 43(6):1571–1587.

Keith Rayner, Alexander Pollatsek, Jane Ashby, and Charles Clifton Jr. 2012. *Psychology of reading*. Psychology Press.

Stefan Riezler, Tracy H King, Richard Crouch, and Annie Zaenen. 2003. Statistical sentence condensation using ambiguity packing and stochastic disambiguation methods for lexical-functional grammar. In *NAACL*.

Kristian Woodsend and Mirella Lapata. 2011. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *EMNLP*.

Jie Zhou and Wei Xu. 2015. End-to-end learning of semantic role labeling using recurrent neural networks. *ACL*.