Abstractive Text Summarization with Application to Bulgarian News Articles

Nikola Taushanov
Faculty of Mathematics and Informatics
Sofia University St. Kliment Ohridski
nktaushanov@gmail.com

Ivan Koychev
Faculty of Mathematics and Informatics
Sofia University St. Kliment Ohridski
koychev@fmi.uni-sofia.bg

Preslav Nakov
Qatar Computing Research Institute
HBKU
pnakov@qf.org.qa

Abstract

With the development of the Internet, a huge amount of information is available every day. Therefore, text summarization has become critical part of our first access to the information. There are two major approaches for automatic text summarization: abstractive and extractive. In this work, we apply abstractive summarization algorithms on a corpus of Bulgarian news articles. In particular, we compare selected algorithms of both techniques and we show results which provide evidence that the selected state-of-the-art algorithms for abstractive text summarization perform better than the extractive ones for articles in Bulgarian. For the purpose of our experiments we collected a new dataset consisting of around 70,000 news articles and their topics. For research purposes we are also sharing the tools to easily collect and process such datasets.

1. Introduction

Text summarization is the task of creating a shorter version of a given text that retains the most important pieces of information. There are two major approaches for automatic text summarization: abstractive and extractive. The latter selects different parts (sentences) of the text in order to construct the summary. On the other hand, the abstractive summarization is considered closer to the way people approach the problem: they first analyze and understand the input and then generate the content of the summary.

Recent studies (Nallapati et al., 2016) have shown that abstractive summarization methods perform better than extractive ones, but it is clear that the two approaches have different problems, which still need to be solved. Abstractive summaries are often unable to provide accurate factual details and they also tend to repeat words or sentences as shown in (See et al., 2017). Extractive summaries on the other hand have problems related to the fact that sentences cannot easily be separated from the context, especially when they contain references to others which are not extracted.

A great amount of work has been done on applying, evaluating, and improving these models in English, but not a lot of research exists for other languages. This document shows results of applying effective methods in both abstractive and extractive text summarization on a big corpus of news articles in Bulgarian. It should also serve as a starting point for future research in this language.

In the rest of this work, Section 2 provides more context on the related work and how ours fits in it. Section 3 has more details on the models which will be used. The experiments and the results of applying them are shown in Section 4, and in Section 5 we provide qualitative analysis on the produced output. Section 6 concludes our work and gives direction for future developments.

2. Related work

The majority of the research made in the past in the area of text summarization focuses on extractive methods. Their goal is to extract important sentences and use them to form summaries. Earlier research

Keywords: NLP, abstractive text summarization, encoder-decoder, deep recurrent encoder

is based on features of the sentences such as position in the text, frequency of the words or sentences mostly based on TF-IDF (Edmundson, 1969; Baxendale, 1958). Some of the best results were achieved when the text is represented as a graph, in which each sentence is a node and the edges and their weights are based on similarity metrics. The problem then shifts to finding the most important or most central node in the graph. The node degree is used in (Freeman, 1978) and eigenvector centrality in (Bonacich, 1972). Variation of the eigenvector centrality is used for PageRank in (Page et al., 1999). The same graph algorithm but a different distance function adapted for sentences is used in (Erkan and Radev, 2004) and in (Mihalcea and Tarau, 2004). An algorithm similar to PageRank, which uses absorbing Markov chains to encourage diversity among the top ranked nodes, is proposed in (Zhu et al., 2007).

With the recent development of large computing resources, the enormous amount of available data online, and the research and advancements made in the area of deep neural networks, the focus falls back to abstractive summarization. Initially, for the problem of machine translation, some of the first works which apply encoder-decoder networks are (Cho et al., 2014b; Cho et al., 2014a). In (Sutskever et al., 2014) they show promising results by using Long Short-Term Memory (LSTM) units (Hochreiter and Schmidhuber, 1997) and reversing the input sequence. In (Schuster and Paliwal, 1997) bidirectional recurrent neural networks (RNNs) are used for the first time, to the best of our knowledge. (Bahdanau et al., 2014) show that using attention in bidirectional RNNs improves the encoder-decoder performance even further.

In (Rush et al., 2015), inspired by the machine translation models, they use neural attention model for abstractive text summarization. In (Nallapati et al., 2016; See et al., 2017) switching generator-pointers are applied in order to solve the common problems of inaccurately reproducing details, inability to use out-of-vocabulary words and repetition of words or sentences.

In this work, we use a model architecture similar to (Bahdanau et al., 2014) but we also apply multi-layer bidirectional RNNs as in (Vinyals et al., 2014). The solution also addresses the very common problem of covariant shifting in deep neural networks using layer normalization (Ba et al., 2016). We apply this model on a novel corpus in Bulgarian and compare the results with extractive models which implement TextRank with a few different similarity metrics.

3. Implemented Methods for Summarization

In this section, we describe the selected extractive models and the proposed abstractive ones.

3.1. Extractive Summarization

The extractive summarization methods identify the most important parts of the text, extract them, and then use them to create a summary. Some of the best results in this area are observed in models which use a modification of the PageRank algorithm (Page et al., 1999). We have chosen to implement and apply TextRank (Mihalcea and Tarau, 2004) using two different similarity metrics.

It is a graph-based ranking algorithm for computing relative importance of vertices within a graph. Each vertex V_i is essentially a sentence from the input text. The weight w_{ij} of an edge between two nodes V_i and V_j in the graph is defined by the value of a similarity metric applied on their corresponding sentences. To build a weighted score WS for each node V_i, the algorithm uses the slightly modified PageRank formula (1), where d is the damping factor, with value between 0 and 1, used for modeling the probability of jumping from a given vertex to another random one. Each sentence is then ranked based on the score of its node.

$$WS(V_i) = (1-d) + d \sum_{j \in In(V_i)} \frac{w_{ij}}{\sum_{v_k \in Out(V_j)} w_{jk}} WS(V_j) \tag{1}$$

In the first model - *TR*, we use a similarity metric which measures the content overlap of two sentences and it is the one proposed in the original work (Mihalcea and Tarau, 2004). Given two sentences S_i and S_j, each represented as $S_i = s_1^i, s_2^i, \ldots, s_{|S_i|}^i$, the similarity between them is defined in (2)

$$Similarity(S_i, S_j) = \frac{|\{s_k \mid s_k \in S_i \& s_k \in S_j\}|}{log(|S_i|) + log(|S_j|)} \tag{2}$$

Representing the sentences using the vector space model (Salton et al., 1975) creates TF-IDF weighted vectors for each of them. A cosine similarity on those vectors is used in the second model called TR-cosine. This is similar to the solution proposed in (Erkan and Radev, 2004) but without the binarization of the graph weights which they propose.

Regardless of the similarity metric, the algorithm scores each sentence, ranks them accordingly and picks the top scoring ones for the summary.

3.2. Abstractive Summarization

The state-of-the-art abstractive summarization models use sequence-to-sequence with attention networks in order to read the input text and then generate a summary word by word. The baseline model, which we have implemented and applied, is very similar to the one proposed in (Nallapati et al., 2016). It is the recurrent neural network with encoder-decoder architecture which is depicted in Figure 1. Each word of the text $X_0, X_1, \ldots, X_n$, is first transformed using a word embeddings layer. It is then fed to the multi-layered bidirectional encoder. In comparison (Nallapati et al., 2016) uses a single layer.

Also, each cell in both the encoder and the decoder is implemented with a LSTM unit instead of Gated Recurrent Unit (Cho et al., 2014b). The last layer of the encoder is connected to the decoder using attention as in (Bahdanau et al., 2014) and calculated with (3) and (4)

$$e_i^t = v^T tanh(W_h h_i + W_s s_t + b_{attn}) \tag{3}$$

$$a^t = softmax(e^t) \tag{4}$$

where v, W_h, W_s and b_{attn} are learnable parameters, h_i is the hidden state of the encoder at encoding step i, s_t is the decoder state and a^t is the attention vector at timestep t.

The result from the decoder is transformed back to a word using a projection layer. We will refer to this baseline model with *s2s-lstm*.

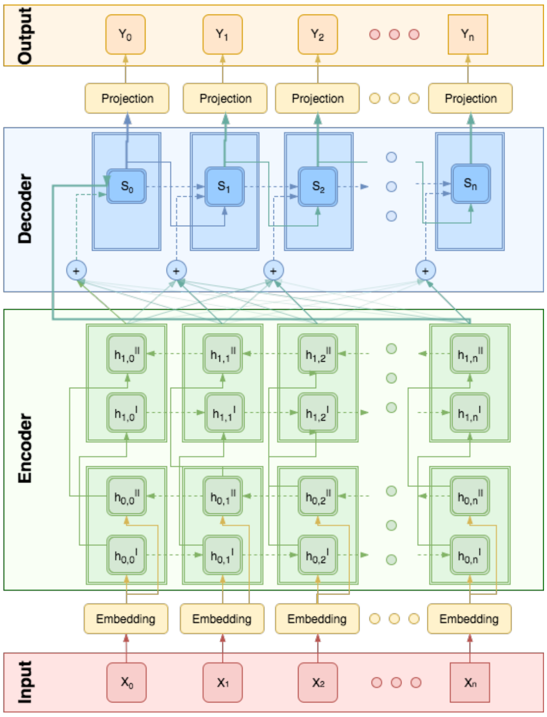

Figure 1: Encoder-decoder RNN with embeddings, multi-layered encoder and attention.

We will extend the baseline model with dropout (Srivastava et al., 2014) for better regularization of the network and call it *s2s-lstm+d*. The dropout factor is part of the hyper parameters of the model.

The final modification which we have made, which leads to model *s2s-lstm+d+ln*, is a network layer normalization (Ba et al., 2016). This technique is often used to solve the problem of covariant shifting in deep neural networks.

All of the selected models will use gradient clipping (Pascanu et al., 2012) to help with the exploding gradients, Xavier weights initialization (Glorot and Bengio, 2010) for an improved network initialization and Adagrad gradient descent (Duchi et al., 2011) for better learning. The loss function is a weighted cross-entropy for a sequence of logits with sampling (Jean et al., 2014) because the output of the network is a sequence of words and the size of the vocabulary is usually very big.

As usual for the sequence-to-sequence networks, the input fed into the decoder during the training starts with a special word for start of sentence and ends with a special word for end of sentence. During the decoding phase, each output of the decoder is used as input on the next step until an end of sentence is generated. The usual beam search approach for sequence-to-sequence networks is used, in order to find the best result (Cho et al., 2014a).

On the other hand, on each step of the training, the words from the actual summary are the input to the decoder. This approach showed better performance and faster training compared to the curriculum learning strategy proposed in (Bengio et al., 2015) which uses the word generated on the previous step for an input to the current one.

4. Experiments and results

4.1. Dataset

FocusNews: As of the time of writing, there is no big enough dataset in Bulgarian which could be used for abstractive text summarization. As part of this work, we created a big corpus which is suitable to run our experiments upon. An appropriate source for this was the FocusNews news agency website (FOCUS, 2018). It contains around 70000 articles at any given moment. Extracting articles for the period from January 2017 to September 2017 resulted in a corpus of 76300 news articles and their headlines. It contains texts with minimal length of 1 word, a maximum of 8854 and average of 20. The size of the vocabulary of words used is 200 000. This dataset will be split into a training set of size 65472, a validation set with 7271 and a test set with 3557 articles. The code for collecting and processing such datasets is available online[1].

Compared to some popular English corpora, where the data is anonymized and the names of towns, people, countries, etc. are replaced with tokens, in this corpus it is not. The articles are very close to those a person would read. Because all the articles and their headlines come from a reputable source, we could easily make the assumption that the headlines contain the most important information and could be used as short summaries of the articles.

DUC-2004: A dataset presented on the DUC 2004 (NIST, 2004) competition has been the default way to experiment and test automatic text summarization in various researches. It consists of a small number of English news articles on different topics with multiple human produced reference summaries for each of them. This dataset is small and unsuitable for training abstractive models but works well for extractive ones.

4.2. Evaluating the FocusNews dataset

FocusNews is a new corpus which has not been used in other research so far. We will compare the results of applying the same models on DUC-2004 and FocusNews using ROUGE-1, ROUGE-2 and ROUGE-L recall, precision and F-scores (Lin, 2004). In Table 1 we show that both datasets are of similar quality. As expected, the models show better scores when applied on DUC-2004, given the fact that its reference summaries are human prepared and well selected. When looking at ROUGE-1 and ROUGE-L, both models show better results for DUC-2004 than FocusNews, but for ROUGE-2 the results are very close to each other. In both datasets *TR* is better than *TR-cosine*. Despite the difference in the results, the numbers show that FocusNews is a suitable dataset for our experiments.

[1] `https://github.com/nktaushanov/focusnews`

	ROUGE-1			ROUGE-2			ROUGE-L		
	R	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**
DUC-2004: TR	*0.292*	*0.291*	*0.292*	*0.046*	*0.046*	*0.046*	*0.256*	*0.255*	*0.255*
DUC-2004: TR-cosine	0.256	0.255	0.255	0.038	0.038	0.038	0.247	0.247	0.247
FocusNews: TR	0.186	0.153	0.160	0.057	0.049	0.051	0.180	0.149	0.155
FocusNews: TR-cosine	0.147	0.121	0.126	0.044	0.037	0.038	0.144	0.118	0.122

Table 1: Comparing DUC-2004 and FocusNews with extractive summarization

4.2.1. Models evaluation and analysis

The three selected abstractive models have been tested with a variety of different hyper parameters. The best results from all tests were observed with the hyper parameters specified in Table 2.

	s2s-lstm	**s2s-lstm+d**	**s2s-lstm+d+ln**
Minimum learning rate	0.01	0.01	0.01
Batch size	50	200	50
Learning rate	0.15	0.15	0.15
Encoding layers	3	2	2
Encoding steps	120	120	120
Decoding steps	40	30	30
Minimum input length	2	2	2
Hidden state size	256	256	256
Embedding dimensions	128	128	128
Max gradient norm	2	2	2
Dropout keep probability	1.0	0.7	0.5
Num of loss samples	4096	4096	4096
Max article sentences	4	4	4
Min article sentences	2	2	2

Table 2: Hyper parameter of the abstractive models

The evaluation of the models on FocusNews presented in Table 3 shows a clear performance difference between the abstractive models *s2s-lstm* and *s2s-lstm+d* compared to *s2s-lstm+d+ln* which has layer normalization. The ROUGE-1, ROUGE-2 and ROUGE-L scores for the last one are almost twice higher compared to the first two models. It is clear that covariant shifting and better regularization make a huge difference in this task. In the case of extractive summarization, *TR* performs better than *TR-cosine* which means that the similarity metric of content overlap is better than the cosine distance of the vectorized sentences.

Comparing the best results from the extractive and abstractive algorithms, it looks like the latter perform better, although the numbers are not that far apart. In each of the recall metrics they produce almost the same results, but the extractive solution performs worse based on precision and F-score.

	ROUGE-1			ROUGE-2			ROUGE-L		
	R	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**
s2s-lstm	0.102	0.109	0.103	0.012	0.013	0.012	0.102	0.109	0.103
s2s-lstm+d	0.093	0.103	0.096	0.014	0.016	0.015	0.093	0.103	0.096
s2s-lstm+d+ln	*0.192*	*0.198*	*0.191*	*0.052*	*0.053*	*0.052*	*0.192*	*0.198*	*0.191*
TR	0.186	0.153	0.160	0.057	0.049	0.051	0.180	0.149	0.155
TR-cosine	0.147	0.121	0.126	0.044	0.037	0.038	0.144	0.118	0.122

Table 3: Evaluation of all the models on FocusNews

5. Qualitative Analysis

Table 4 shows a couple of good summary examples generated from our best model *s2s-lstm+d+ln*. All of them are correct and do not have any syntactical or semantical problems.

In the first example we can observe how the generated summary is different from any of the sentences in the text but it still has the same meaning as the original. Instead of having the exact same words as in the article, it contains their synonyms - such as ограничава (*ogranichava*, ”restricts”) instead of ”спира” (spira, ”stops”) and ”в двете посоки” (dvete posoki, ”in both directions”) instead of ”в двете платна” (v dvete platna, ”in both lanes”). Looking closely at the original text and the generated one, it looks like the original sentence was transformed by omitting some of the words which were not important and replacing others with their synonyms.

In the second example, the generated summary has less details but is a good paraphrase. In the original one, the text says that the landfill of the town is on fire whereas in the generated - there is a fire in the area of the town. The latter is a bit more generic, but still preserves the important information. What is more interesting in this case is the fact that the generated summary has a reference to an earlier part of the sentence - ”града” (grada, ”the town”) which refers to ”пазарджик” (Pazardzhik) - the name of the town.

Both examples show impossible to achieve with extractive summarization situations which could often be observed in human made summaries. Unfortunately, the widely used scoring metrics cannot measure very well those solutions.

Article	пловдив . спира се движението на превозни средства в двете платна на бул . ” александър стамболийски ” в пловдив заради ремонт на водопроводната мрежа . това съобщиха за радио ” фокус ” – пловдив от оп ” организация и контрол на транспорта ” . затворен ще бъде участъкут от ул . ” никола димков ” до ул . ” стефан стамболов ” на 16.08.2017г /сряда/ от 9:00 ч до 16:00 часа . маршрутите на автобусни линии # 16 , 20 , 27 и 36 от вътрешноградския транспорт също се променят . цветана тончева
Original	пловдив : спират движението по бул . ” александър стамболийски ” в града заради ремонт на 16 август
Generated	пловдив : ограничава се движението в двете посоки на бул . ” александър стамболийски ” заради ремонт на водопроводната мрежа
Article	пазарджик . гори сметището на пазарджик , предаде репортер на агенция ” фокус ” . виждат се пламъци от пътя . задимен е пътят за селата капитан димитриево и дебръщица по посока пътя пазарджик-пещера .
Original	гори сметището на пазарджик
Generated	пазарджик : пожарът в района на града
Article	78-годишна жена от град сливен е станала жертва на телефонна измама . това съобщиха от областната дирекция на мвр – сливен . потърпевшата е била въвлечена в заблуждение , че помага на органите на реда при залавянето на престъпна група , занимаваща се с телефонни измами ...
Original	възрастна жена е станала жертва на телефонна измама
Generated	възрастна жена от града е станала жертва на телефонна измама

Table 4: Examples of good summaries generated from model *s2s-lstm+d+ln*

There is an example of an average summary in Table 5. Its content is ambiguous and has wrong details - ”7-те задържани” (sedemte zadarzhani, ”the 7 detained”) instead of ”14 обвиняеми” (14 obvinyaemi, ”14 defendants”). It is still a correct sentence whose meaning is very close to the original one, but the detail mismatch is hard to ignore. This shows that abstractive approaches are generally better at paraphrasing and showing generic information but they are more error-prone when specific details are in place - numbers, places, people, etc.

Article	пазарджик . окръжният съд в пазарджик започна изслушването на компютърни и технически експертизи , извършени от две вещи лица по делото срещу 14 обвиняеми за разпространение идеите на идил , предаде репортер на радио ” фокус ” . . .
Original	пазарджик : окръжният съд започна изслушването на компютърни и технически експертизи по делото срещу 14 обвиняеми за разпространение идеите на идил
Generated	пазарджик : окръжният съд започна изслушването на 7-те задържани по делото за разпространение идеите на идил

Table 5: Example of an average summary generated from model *s2s-lstm+d+ln*

Regardless of the good examples, there are lot of bad summaries in the output as well. The one in Table 6 shows a completely erroneous summary which makes no real sense. Other common issues which we observed were incorrect people names, ambiguities, word repetitions, etc.

Article	галерия видин . паметникът на благодарността в центъра на града , пострадал от вандалски акт , е почистен . това съобщиха от пресцентъра на община видин . ” това е възмутително и недопустимо деяние ” , заяви кметът на община видин огнян ценков по повод оскверняването на паметника до стамбол капия , поставен в знак на благодарност...
Original	видин : паметникът на благодарността в центъра на града , пострадал от вандалски акт , е почистен
Generated	видин : паметникът на бензина в центъра на града , пострадали от войните , е почистен

Table 6: Example of a bad summary generated from model *s2s-lstm+d+ln*

6. Conclusions

In this work, we experiment with both extractive and abstractive automatic text summarization and show that the latter performs better on articles in Bulgarian. To the best of our knowledge, no other work so far has applied abstractive summarization in this language. We also propose a novel benchmarked dataset in Bulgarian which is suitable for training and evaluation of abstractive models. Future research would focus on resolving the issues of inaccurate factual details and unnecessary repetition.

References

Ba, L. J., Kiros, R., and Hinton, G. E. (2016). Layer normalization. *CoRR*, abs/1607.06450.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Baxendale, P. B. (1958). Machine-made index for technical literature: An experiment. *IBM J. Res. Dev.*, 2(4):354–361, October.

Bengio, S., Vinyals, O., Jaitly, N., and Shazeer, N. (2015). Scheduled sampling for sequence prediction with recurrent neural networks. *CoRR*, abs/1506.03099.

Bonacich, P. (1972). Factoring and weighting approaches to status scores and clique identification. *The Journal of Mathematical Sociology*, 2(1):113–120.

Cho, K., van Merrienboer, B., Bahdanau, D., and Bengio, Y. (2014a). On the properties of neural machine translation: Encoder-decoder approaches. *CoRR*, abs/1409.1259.

Cho, K., van Merrienboer, B., Gülçehre, Ç., Bougares, F., Schwenk, H., and Bengio, Y. (2014b). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, 12:2121–2159, July.

Edmundson, H. P. (1969). New methods in automatic extracting. *J. ACM*, 16(2):264–285, April.

Erkan, G. and Radev, D. R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.*, 22:457–479.

FOCUS. (2018). *FOCUS Information Agency*. `http://www.focus-news.net/`, Accessed: 2018-02-04.

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social Networks*, 1(3):215 – 239.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In Teh, Y. W. and Titterington, M., Eds., *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May. PMLR.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November.

Jean, S., Cho, K., Memisevic, R., and Bengio, Y. (2014). On using very large target vocabulary for neural machine translation. *CoRR*, abs/1412.2007.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Proc. ACL workshop on Text Summarization Branches Out*, page 10.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into texts. In Lin, D. and Wu, D., Eds., *Proceedings of EMNLP 2004*, pages 404–411, Barcelona, Spain, July. Association for Computational Linguistics.

Nallapati, R., Xiang, B., and Zhou, B. (2016). Sequence-to-sequence rnns for text summarization. *CoRR*, abs/1602.06023.

NIST. (2004). *Document Understanding Conference DUC-2004*. `http://duc.nist.gov/duc2004/`, Accessed: 2018-02-04.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). *The PageRank Citation Ranking: Bringing Order to the Web.* Technical Report 1999-66, Stanford InfoLab, November. Previous number = SIDL-WP-1999-0120.

Pascanu, R., Mikolov, T., and Bengio, Y. (2012). Understanding the exploding gradient problem. *CoRR*, abs/1211.5063.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. *CoRR*, abs/1509.00685.

Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, November.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, Nov.

See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. *CoRR*, abs/1704.04368.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, January.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. *CoRR*, abs/1409.3215.

Vinyals, O., Kaiser, L., Koo, T., Petrov, S., Sutskever, I., and Hinton, G. E. (2014). Grammar as a foreign language. *CoRR*, abs/1412.7449.

Zhu, X., Goldberg, A., Gael, J. V., and Andrzejewski, D. (2007). Improving diversity in ranking using absorbing random walks. *HLT-NAACL*, pages 97–104.