Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference “Dialogue 2023”

Moscow, June 14–17, 2023

Scaled Down Lean BERT-like Language Models for Anaphora Resolution and Beyond

Vladislav Bolshakov
NTR Labs and
Bauman Moscow State
Technical University
Moscow, Russia
vbolshakov@ntr.ai

Rostislav Kolobov
NTR Labs
Tomsk, Russia
rkolobov@ntr.ai

Eugene Borisov
NTR Labs and
Higher IT School of
Tomsk State University
Tomsk, Russia
eborisov@ntr.ai

Nikolay Mikhaylovskiy
NTR Labs and
Higher IT School of
Tomsk State University
Moscow, Russia
nickm@ntr.ai

Gyuli Mukhtarova
NTR Labs
Moscow, Russia
gmukhtarova@ntr.ai

Abstract

We study performance of BERT-like distributive semantic language models on anaphora resolution and related tasks with the purpose of selecting a model for on-device inference. We have found that lean (narrow and deep) language models provide the best balance of speed and quality for word-level tasks, and opensource[1] RuLUKE-tiny and RuLUKE-slim models we have trained. Both are significantly (over 27%) faster than models with comparable accuracy. We hypothesise that the model depth may play a critical role for performance as, according to recent findings each layer behaves as a gradient descent step in autoregressive setting.

Keywords: BERT, LUKE, Anaphora Resolution
DOI: 10.28995/2075-7182-2023-22-1032-1040

Поджарые BERT-подобные модели для разрешения анафоры и не только

Владислав Большаков
ООО “НТР”
МГТУ им. Баумана
Москва
vbolshakov@ntr.ai

Ростислав Колобов
ООО “НТР”
Томск
rkolobov@ntr.ai

Евгений Борисов
ООО “НТР”
Высшая ИТ-Школа ТГУ
Томск
eborisov@ntr.ai

Николай Михайловский
ООО “НТР”
Высшая ИТ-Школа ТГУ
Москва
nickm@ntr.ai

Гюли Мухтарова
ООО “НТР”
Москва
gmukhtarova@ntr.ai

Аннотация

Изучена эффективность BERT-подобных моделей на задачах разрешения анафоры и смежных задачах, чтобы выбрать модели для использования на оконечном устройстве. Выяснено, что поджарые (узкие и длинные) языковые модели дают оптимальное соотношение скорости и качества. Представлены модели RuLUKE-tiny и RuLUKE-slim с открытым исходным кодом. Обе заметно (более чем на 27%) быстрее, чем модели со сравнимой точностью. Предположено, что глубина модели может играть решающую роль для ее эффективности, поскольку, согласно недавним исследованиям, каждый слой ведет себя как шаг градиентного спуска в условиях авторегрессии.

Ключевые слова: BERT, LUKE, разрешение анафоры

[1] https://huggingface.co/vbolshakov/RuLUKE-tiny
https://huggingface.co/vbolshakov/RuLUKE-lean

1 Introduction

1.1 Anaphora Resolution

Anaphora is the use of an expression (a pronoun or a noun phrase) whose interpretation depends upon a preceding expression in context (its antecedent). Anaphora and cataphora (which is the use of an expression that depends upon a postcedent expression) both are special cases of coreference, which occurs when two or more expressions in a text refer to the same person or thing.

Anaphora resolution is the problem of resolving what a pronoun, or a noun phrase refers to. We are specifically interested in resolving pronoun anaphora. It is a challenging task because it requires good understanding of the context and the ability to recognize complex relationships between words and phrases (Bolshakov and Mikhaylovskiy, 2023). However, this task is crucial in many applications of NLP, such as information retrieval (Schmolz, 2015), question answering (Castagnola, 2002), opinion mining (Jakob and Gurevych, 2010), and natural language understanding (Kilicoglu et al., 2016). In addition, anaphora resolution can be used to improve the readability of a text, by replacing repeated mentions of the same entity with a pronoun or other reference.

Recent anaphora and coreference resolution approaches typically use some fine-tuned pretrained language model. As coreference resolution approaches are reviewed in detail recently by (Bolshakov and Mikhaylovskiy, 2023), here we only list some specifically anaphora resolution work. The use of BERT-like models for anaphora resolution was likely first suggested by (Joshi et al., 2019). At about the same time (Mohan and Nair, 2019) suggested resolving ambiguous pronoun anaphorae using BERT and SVM, and (Wang, 2019) suggested a BERT-based approach for gendered pronoun resolution. (Hou, 2020) suggests an approach to bridging anaphora resolution via question ansvering based on SpanBERT (Joshi et al., 2020).

1.2 Downscaling Transformers

A lot of recent research have focused on laws of and approaches to scaling transformer (Vaswani et al., 2017) language models up (Hoffmann et al., 2022; Kaplan et al., 2020; Rae et al., 2021; Shoeybi et al., 2019). Significantly less effort is being devoted to building smaller and more compute-efficient models (Geiping and Goldstein, 2022). In this work we continue the latter line of research, with a focus on the use of transformers in anaphora resolution.

1.3 Our contribution

Our contribution in this paper is threefold:

- We cast the anaphora resolution problem in a form similar to named entity recognition and linking
- We empirically study the performance of varied language model architectures and training approaches and found that lean (narrow and deep) language models provide the best balance of speed and quality for word-level tasks,
- Finally, we opensource RuLUKE-tiny and RuLUKE-slim models we have trained that have better performance on our downstream tasks than comparable models, and the larger of two models we present performs on par with significantly larger models.

2 Anaphora Resolution Approach

For the anaphora resolution problem, we suggest an approach inspired by tagging named entities using embeddings extracted from the transformer model (see, for example, (Arkhipov et al., 2019)). Instead of named entity BIO tags (introduced by (Ramshaw and Marcus, 1995), see also (Nadeau and Sekine, 2007)) we suggest the following four tags:
0) 0 - the tag of all the words that are not in an anaphoric connection with the target pronoun;
1) AT_B - the tag of the first token included in the antecedent;
2) AT_C - the tag of subsequent tokens included in the antecedent;
3) AF_B - the tag of the first token of the anaphora.

We measure the accuracy of models with F1 metric applied to each token type, producing 4 separate metrics. This approach allows to analyze the models' performance in detail.

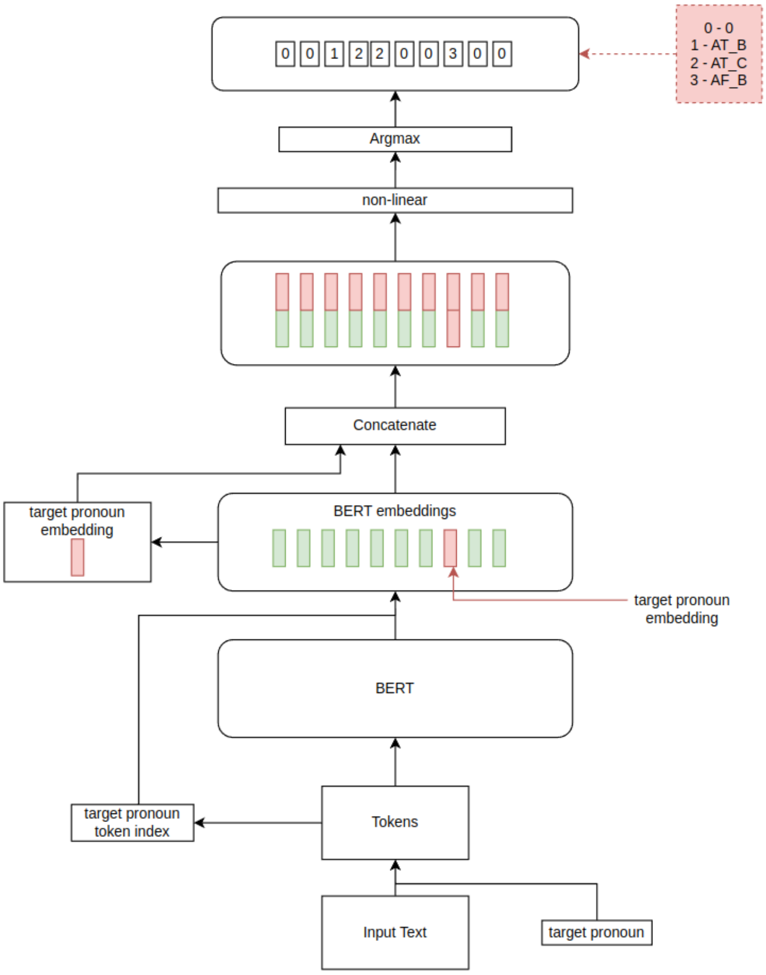

Figure 1: Architecture of the Anaphora Resolution model

Figure 1 depicts the suggested architecture for the anaphora resolution. Embeddings are generated for the tokens of the source text. The embedding of the token of the anaphora pronoun is concatenated with the embeddings of all tokens of the input text, and the result is passed to a fully-connected 2-layer network. At the output of the model, the argmax layer returns the indices of the most likely tags for each token. These tags are mapped to the words of the source text. In one pass over a window, the model finds antecedents for only one pronoun.

The largest (in terms of the number of words) antecedent is selected for each continuous span detected: "**Elizaveta Petrovna Kalinina** is the CEO of the company. **Liza** is responsible for a huge number of employees. Every day **she**...". The case and number of the antecedent are agreed with the pronoun. In cases where there were opening brackets/quotes in the antecedent, but their closing versions were not included, they are added.

3 Training approaches

We have benchmarked several approaches to training the models:

- Distillation
- Pretraining using SpanBERT approach (Joshi et al., 2020)
- Pretraining using LUKE approach (Yamada et al., 2020)

3.1 Distillation

We use knowledge distillation (Hinton et al., 2015) as a basic approach to training the models, following (Dale, 2021b). Knowledge distillation is the process of transferring knowledge from a large model (teacher) to a smaller one (student) (Gou et al., 2021).

We train bilingual English+Russian models and use two training sets:

- 2.5 million parallel English-Russian sentences collected from Yandex.Translate (Yan, 2022), OPUS-100 (Zhang et al., 2020) and Tatoeba (Tat, 2022; Tiedemann, 2012) corpora.
- 6.5 million sentences in Russian from ruswiki 2021, rusnews 2020 and rusweb 2019 collections from Leipzig corpus (Goldhahn et al., 2012) and Russian sentence pairs from XNLI (Conneau et al., 2018).

We use several losses and teachers. For the parallel corpus we, similarly to Dale (Dale, 2021b):

- distill CLS tokens, bringing their different projections closer to RuBERT (Kuratov and Arkhipov, 2019), LaBSE (Feng et al., 2022) and Laser (Artetxe and Schwenk, 2019) embeddings;
- distill the probability distribution of LaBSE (Feng et al., 2022) output tokens with MLM distillation loss, using the Kullback-Leibler divergence loss between mapped vocabularies of student and teacher models;
- minimize the whole-word MLM loss (Devlin et al., 2019) for English and Russian languages;
- minimize the translation ranking loss, as in LaBSE (Feng et al., 2022);

For the Russian corpus we

- minimize the per-token MLM loss with rubert-base-cased-sentence (Kuratov and Arkhipov, 2019);
- minimize the whole-word MLM loss to LABSE (Feng et al., 2022);
- minimize NLI loss.

We train models in three stages using Cosine Annealing with Warm Restarts on the first two stages. The details of the parameters on these stages are listed in Table 1. We have trained two models using this approach - one with rubert-tiny architecture but with extended dictionary (we call it distilRuBert-tiny) and the other with twice as much layers (we call it distilRuBert-lean). Input and output embeddings weights for these new models were partially copied from cointegrated/rubert-tiny2 and cointegrated/LaBSE-en-ru respectively. For distilRuBert-tiny we used cointegrated/rubert-tiny2 as a starting checkpoint.

	Stage 1	**Stage 2**	**Stage 3**
teachers	all	all	all but rubert-base-cased-sentence
steps	400000	800000	1100000
batch size for bilingual pairs	16	32	36
batch size for Russian texts	8	16	12
batch size for NLI	8	16	24
accumulation steps	4	4	1
learning rate	1e-5 to 0	1e-5 to 1e-6	1e-5

Table 1: Distillation stages description

3.2 Pretraining using SpanBERT approach

SpanBERT (Joshi et al., 2020) extends BERT by

- masking contiguous random spans, rather than random tokens, and
- training the span boundary representations to predict the entire content of the masked span, without relying on the individual token representations within it.

We only use the first option of these two. We have trained a model with rubert-tiny architecture using this approach, and refer to it distilRuSpanBert-tiny in this paper. We used cointegrated/rubert-tiny2 as a starting checkpoint.

Model	V	E	H	L	N
DeepPavlov/distilrubert-small-cased-conversational	119547	768	3072	2	106.4M
DeepPavlov/distilrubert-tiny-cased-conversational-v1	30522	264	792	3	10.3M
DeepPavlov/distilrubert-tiny-cased-conversational-5k	5031	264	792	3	3.6M
cointegrated/LaBSE-en-ru	55083	768	3072	12	127M
cointegrated/rubert-tiny2	83828	312	600	3	29.1M
cointegrated/rubert-tiny	29564	312	600	3	11.8M
(ours) distilRuBert-lean	55083	312	936	6	23.3M
(ours) distilRuBert-tiny	101520	312	600	3	34.3M
(ours) RuLUKE-tiny	83828	312	600	3	158.8M
(ours) RuLUKE-lean	55083	312	936	6	153.3M
(ours) distilRuSpanBert-tiny	101525	312	600	3	34.3M

Table 2: Parameters of models

3.3 Pretraining using LUKE approach

LUKE (Yamada et al., 2020) extends BERT by introducing:

- a new pretraining task that involves predicting randomly masked words and entities in a large entity-annotated corpus retrieved from Wikipedia
- an entity-aware self-attention mechanism that is an extension of the self-attention mechanism of the transformer, and considers the types of tokens (words or entities) when computing attention scores

We have trained two models using this approach - one with rubert-tiny2 architecture (we call it RuLUKE-tiny) and the other with distilRuBert-lean architecture (we call it RuLUKE-lean). For RuLUKE-tiny and RuLUKE-lean we used cointegrated/rubert-tiny2 and our distilRuBert-lean respectively as backbone transformers and starting checkpoints for further training. According to LUKE (Yamada et al., 2020), each model has additional entity vocabulary with top 500k entities from dump of Russian Wikipedia, that is why the disk size and the number of parameters of RuLUKE-tiny and RuLUKE-lean are larger compared to other models.

4 Experiments and results

4.1 Tasks and Datasets

We test the efficiency of the anaphora resolution approach overall and of each model in particular on the anaphora resolution subset of RuCoCo dataset (Dobrovolskii et al., 2022). To produce this subset we have sampled examples where one of the coreferences is a pronoun.

4.2 BERT-like Models

For our study, we selected small and medium sized BERT-like models that showed promising results in NLP tasks for the Russian language (Kolesnikova et al., 2022), (Dale, 2021b), based on the rating from Dale (Dale, 2022) and integrated well with spaCy (Honnibal and Montani, 2017). The size, performance and efficiency of BERT-like models depends on model architecture parameters and training approach. We treat the latter in Section 3. The key architectural parameters for BERT are:

- L - the number of hidden layers;
- H - the size of intermediate layer embeddings;
- E - the size of the output embedding;
- V - the size of vocabulary;
- N - the number of parameters (which is a function of the above parameters)

The Table 2 lists the architectural parameters of the key models we compare to and our models.

4.3 Inference Speed

We have benchmarked the performance of CPU inference of typical and potential architectures. The tests were run on the entire dataset. Time was measured in ms/sentence, mean of 3 runs, 1 loop each on an

Intel(R) Core(TM) i5-10400, 2.90GHz processor based computer with 6 cores. Batch size was set to 1, and torch.utils.data.DataLoader used $num\_workers = 0$. The results are listed in Table 3. It is easy to see that the performance only slightly depends on the vocabulary size and intermediate embedding dimension, grows linearly with the number of layers and slower - with the embedding dimension.

		$E = 264$				$E = 528$				$E = 768$			
	L	3	6	9	12	3	6	9	12	3	6	9	12
V =	$H = E * 2$	10.7	19.4	28.5	37.2	15.5	29.5	43.0	56.7	23.9	45.9	67.9	90.0
29	$H = E * 3$	11.3	20.6	30.2	39.3	17.1	32.3	47.7	62.8	28.5	54.9	82.7	108.9
564	$H = E * 4$	11.4	21.1	31.0	40.7	20.3	39.1	61.8	77.6	33.8	65.0	96.4	127.7
V =	$H = E * 2$	11.8	21.5	30.9	40.6	16.2	30.6	45.2	59.6	24.6	47.1	69.8	95.1
83	$H = E * 3$	11.9	22.2	32.5	42.9	17.9	33.9	50.7	67.2	29.7	57.6	85.5	113.4
828	$H = E * 4$	12.3	22.8	33.3	43.8	21.2	40.7	60.2	79.9	35.0	68.1	101.1	133.7
V =	$H = E * 2$	11.8	21.8	32.1	41.9	16.5	31.9	49.1	61.7	24.5	46.8	69.1	92.1
119	$H = E * 3$	12.5	23.1	33.4	43.8	18.1	34.3	51.6	66.8	28.8	55.7	82.7	109.8
547	$H = E * 4$	12.6	23.3	35.2	44.5	21.2	40.8	60.1	79.6	34.0	66.0	98.2	130.3

Table 3: Dependence of performance (ms/sentence) on model architecture parameters L - the number of hidden layers, H - the size of intermediate layer embeddings, E - the size of the output embedding, V - the size of vocabulary

Table 4 shows CPU inference speed of discussed models on two benchmarks:

- $Benchmark1$ - is the CPU speed task from (Dale, 2021a)
- $Benchmark2$ - shows the performance of models on anaphora resolution task when running on CPU using the same data as in Section 4.1 and architecture as in Section 2

For both benchmarks we report mean inference time in milliseconds per sentence and standard deviation, collected on 3 runs.

Model	$Benchmark1$	$Benchmark2$
DeepPavlov/distilrubert-small-cased-conversational	5.22 ± 0.13	28.15 ± 0.67
DeepPavlov/distilrubert-tiny-cased-conversational-v1	2.87 ± 0.06	15.92 ± 0.76
DeepPavlov/distilrubert-tiny-cased-conversational-5k	3.18 ± 0.07	15.36 ± 0.63
cointegrated/LaBSE-en-ru	28.10 ± 0.55	134.07 ± 1.20
cointegrated/rubert-tiny2	3.09 ± 0.12	15.70 ± 0.08
cointegrated/rubert-tiny	3.23 ± 0.09	14.86 ± 0.07
(ours) distilRuBert-lean	6.12 ± 0.09	20.73 ± 0.53
(ours) distilRuBert-tiny	3.23 ± 0.07	12.40 ± 0.47
(ours) RuLUKE-tiny	3.31 ± 0.13	11.79 ± 0.57
(ours) RuLUKE-lean	6.12 ± 0.09	22.01 ± 1.18
(ours) distilRuSpanBert-tiny	3.38 ± 0.18	12.29 ± 0.17

Table 4: CPU inference speed of models

4.4 Accuracy

We list the results of accuracy evaluation in two groups - tiny (Table **??**) and larger (Table 5)models. The results for LaBSE-en-ru are listed with larger models for comparison. The best results in each category are highlighted in bold.

5 Conclusion

Popular tiny Russian BERT models are trained primarily with sentence-related tasks in mind. Thus their accuracy on word-related tasks is significantly worse than on sentence-related tasks. It is hard to fine-

Model	AF_B F1	AT_B F1	AT_C F1	0 F1	Avg F1	Time
(ours) RuLUKE-tiny	0.966	0.372	0.484	0.509	0.583	11.79
(ours) distilRuSpanBert-tiny	0.973	0.290	0.438	0.471	0.543	12.29
(ours) distilRuBert-tiny	0.974	0.315	0.430	0.458	0.544	12.40
cointegrated/rubert-tiny	0.974	0.327	0.416	0.447	0.541	14.86
DeepPavlov/distilrubert-tiny-cased-conversational-5k	0.972	0.333	0.425	0.461	0.548	15.36
cointegrated/rubert-tiny2	0.964	0.318	0.437	0.468	0.547	15.70
DeepPavlov/distilrubert-tiny-cased-conversational-v1	0.972	0.380	0.473	0.511	0.584	15.92
distilRuBert-lean	0.975	0.422	0.490	0.521	0.602	20.73
RuLUKE-lean	0.975	0.411	0.500	0.528	0.604	22.01
DeepPavlov/distilrubert-small-cased-conversational	0.972	0.382	0.499	0.541	0.599	28.15
cointegrated/LaBSE-en-ru	*0.987*	*0.713*	*0.770*	*0.822*	*0.823*	134.07

Table 5: Accuracy and speed of the models

tune/distill such models to achieve better accuracy on word-related tasks than model trained from scratch with word-related tasks in mind. LUKE improves performance on word-related tasks to be on par with the best similarly-sized models, but but is much faster so RuLUKE-tiny is 35% faster than DeepPavlov/ distilrubert-tiny-cased-conversational-v1 that has about the same accuracy. SpanBERT training does not improve performance on anaphora resolution task.

For lean models, the accuracy improvement achieved by LUKE training is more noticeable, and the speedup compared to DeepPavlov/ distilrubert-small-cased-conversational is 28%. Still, 6 layers seems to be an inadequate number to match the performance of full-fledged, 12-layer models such as LaBSE. We believe this might be connected with the recent finding that transformers learn in-context by gradient descent in the domain of autoregressive problems (von Oswald et al., 2022). In the latter setting each layer behaves as a gradient descent step. While our formulation of anaphora resolution task is not autoregressive, a similar mechanism may also be present. This is a matter of the future research.

Acknowledgements

The authors are grateful to their colleagues at NTR Labs ML division for the discussions and support, Prof. Sergey Orlov and Prof. Oleg Zmeev for the computing facilities provided, and anonymous referees who have provided valuable feedback that allowed to significantly improve the paper.

References

Mikhail Arkhipov, Maria Trofimova, Yuri Kuratov, and Alexey Sorokin. 2019. Tuning multilingual transformers for language-specific named entity recognition. // *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, P 89–93, Florence, Italy, August. Association for Computational Linguistics.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Vladislav Bolshakov and Nikolay Mikhaylovskiy. 2023. Pseudo-labelling for autoregressive structured prediction in coreference resolution. // *Dialogue-2023*.

Luciano Castagnola. 2002. Anaphora resolution for question answering.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. // *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

David Dale. 2021a. encodechka-eval. `https://github.com/avidale/encodechka/`.

David Dale. 2021b. Small and fast BERT for russian language, in russian. `https://habr.com/ru/post/562064/`.

David Dale. 2022. Rating of russian-language sentence encoders, in russian. `https://habr.com/ru/post/669674/`.

Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. // *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, volume 1, P 4171–4186.

Vladimir Dobrovolskii, Mariia Michurina, and Alexandra Ivoylova. 2022. Rucoco: a new russian corpus with coreference annotation.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. // *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, P 878–891, Dublin, Ireland, May. Association for Computational Linguistics.

Jonas Geiping and Tom Goldstein. 2022. Cramming: Training a language model on a single gpu in one day. *arXiv preprint arXiv:2212.14034*.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at the Leipzig corpora collection: From 100 to 200 languages. // *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, P 759–765, Istanbul, Turkey, May. European Language Resources Association (ELRA).

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack William Rae, and Laurent Sifre. 2022. An empirical analysis of compute-optimal large language model training. // Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, *Advances in Neural Information Processing Systems*.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Yufang Hou. 2020. Bridging anaphora resolution as question answering. // *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, P 1428–1438, Online, July. Association for Computational Linguistics.

Niklas Jakob and Iryna Gurevych. 2010. Using anaphora resolution to improve opinion target identification in movie reviews. // *Proceedings of the ACL 2010 Conference Short Papers*, P 263–268, Uppsala, Sweden, July. Association for Computational Linguistics.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. 2019. Bert for coreference resolution: Baselines and analysis. // *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, P 5807–5812, 01.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Halil Kilicoglu, Graciela Rosemblat, Marcelo Fiszman, and Thomas C. Rindflesch. 2016. Sortal anaphora resolution to enhance relation extraction from biomedical literature. *BMC Bioinformatics*, 17(1), apr.

Alina Kolesnikova, Yuri Kuratov, Vasily Konovalov, and Mikhail Burtsev. 2022. Knowledge distillation of russian language models with reduction of vocabulary.

Yuri Kuratov and Mikhail Arkhipov. 2019. Adaptation of deep bidirectional multilingual transformers for russian language. *arXiv preprint arXiv:1905.07213*.

Monisha Mohan and Jyothisha J. Nair. 2019. Coreference resolution in ambiguous pronouns using bert and svm. *// 2019 9th International Symposium on Embedded Computing and System Design (ISED)*, P 1–5.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30:3 – 26, 08.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Lance Ramshaw and Mitch Marcus. 1995. Text chunking using transformation-based learning. *// Third Workshop on Very Large Corpora*.

Helene Schmolz. 2015. *Anaphora Resolution and Text Retrieval, A Linguistic Analysis of Hypertexts*. De Gruyter, Berlin, München, Boston.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*.

2022. Tatoeba, a collection of sentences and translations. `https://tatoeba.org/`.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. *Proceedings of the 8th International Conference on Language Resources and Evaluation, LREC 2012*, P 2214–2218.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *// Advances in Neural Information Processing Systems*, P 5999–6009.

Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. 2022. Transformers learn in-context by gradient descent.

Zili Wang. 2019. MSnet: A BERT-based network for gendered pronoun resolution. *// Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, P 89–95, Florence, Italy, August. Association for Computational Linguistics.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. *// Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, P 6442–6454, Online, November. Association for Computational Linguistics.

2022. English-russian parallel corpus (version 1.3), in russian. `https://translate.yandex.ru/corpus`.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation.