Instruction-guided deidentification with synthetic test cases for Norwegian clinical text

Jørgen Aarmo Lund*[1,2], Joel Burman[3], Ashenafi Zebene Woldaregay[3], Robert Jenssen[1], and Karl Øyvind Mikalsen[1,3]

[1]UiT The Arctic University of Norway
[2]DIPS AS
[3]University Hospital of North Norway

Abstract

Deidentification methods, which remove directly identifying information, can be useful tools to mitigate the privacy risks associated with sharing healthcare data. However, benchmarks to evaluate deidentification methods are themselves often derived from real clinical data, making them sensitive themselves and therefore harder to share and apply. Given the rapid advances in generative language modelling, we would like to leverage large language models to construct freely available deidentification benchmarks, and to assist in the deidentification process. We apply the GPT-4 language model to, for the first time, construct a synthetic and publicly available dataset of synthetic Norwegian discharge summaries with annotated identifying details, consisting of 1200 summaries averaging 100 words each. In our sample of documents, we find that the generated annotations highly agree with human annotations, with an F_1 score of 0.983. We then examine whether large language models can be applied directly to perform deidentification themselves, proposing methods where an instruction-tuned language model is prompted to either annotate or redact identifying details. Comparing the methods on our synthetic dataset and the NorSynthClinical-PHI dataset, we find that GPT-4 underperforms the baseline method proposed by Bråthen et al. [1], suggesting that named entity recognition problems are still challenging for instruction-tuned language models.

1 Introduction

Unstructured free text makes up a large portion of healthcare data. Many applications of Natural Language Processing (NLP) methods in healthcare involve extracting or condensing information from this text, such as inferring procedure and diagnosis codes, identifying adverse drug events, and detecting postoperative complications.

Clinical notes have distinct differences in syntax and vocabulary from the texts which make up most NLP training and evaluation corpora: abbreviations, domain-specific terms and terse syntax are common features of patient records [2]. Sharing data and establishing benchmark datasets [3] are important steps in ensuring NLP methods and models are robust to these linguistic differences, but carry significant privacy risks [4].

Deidentification methods attempt to mitigate these risks by detecting personally identifying details, so as to either remove them outright, tag them or substitute them with realistic pseudonyms [5]. These details are typically referred to as *Protected Health Information* (PHI). Deidentification of clinical text can then be treated as a Named Entity Recognition (NER) problem, where the goal is to identify terms which belong to a PHI category. The majority of recent text deidentification methods [6] build on supervised NER methods. The drawback of doing this is that the definition of PHI must be explicitly outlined and annotated for a training set, which the method is trained on. If the user's definition of PHI deviates significantly from the model's assumptions, due to differences in regulation or project restrictions, new annotations must be collected and the method must be retrained. However, zero- or few-shot NER methods could potentially decouple this bond, and allow the deidentification process to reuse the underlying model.

Liu et al. [7] phrase the de-identification task as a set of instructions for GPT-4, and report better recall on the n2c2 2014 deidentification benchmark [8] than their fine-tuned model based on ClinicalBERT [9]. In this article, we test whether a similar approach works for Norwegian clinical text, where fewer annotated text resources are publicly available.

2 Method

We compare two different ways of framing the NER problem as a sequence-to-sequence modelling task for a causal instruction-tuned [10] language model. In the *redaction* setting, we prompt the model to

*Available at: jorgen.a.lund@uit.no

Proceedings of the 5th Northern Lights Deep Learning Conference (NLDL), PMLR 233, 2024.

substitute PHI terms with entity *tags*:

```
"She was over 90 years old."
              ↓
   "She was over <Age> old."
```

In the second *annotation* setting, we prompt the model to keep PHI terms, but add surrounding entity markers:

```
      "She was over 90 years old."
                    ↓
"She was over <Age>90 years</Age> old."
```

We pick these settings mainly to follow the method suggested by Liu et al. [7], but they have additional advantages: they let the models discriminate between different uses of the same term, allow inference to be done in a single pass, and the syntax for the correct solution is known ahead of time. We hypothesize that redaction is an easier task, since annotation requires the model to attend to the previously generated PHI markers in addition to the original text. However, evaluating the results requires aligning the annotated ground truth with the response.

3 Dataset

3.1 Synthetic dataset

To establish a corpus for comparing and training deidentification methods for Norwegian text, we use GPT-4 to synthesize Norwegian clinical notes with annotated surrogate PHI.

The first version of the dataset consists of 1200 synthetic summaries, divided into a training set with 1000 summaries, a validation set of 100 summaries and a holdout test set of 100 summaries, where each summary is 100 words on average. An example summary is shown in Figure 3. To get a diverse set of PHI in the documents, we do not prompt the model to generate the notes from scratch, but randomly sample surrogate PHI which is inserted into prompt templates. These templates prompt the language model to generate an admission note or a discharge summary for the specified patient, where the PHI should be annotated.

Following the format in [1], we prompt the model to generate 8 categories of PHI, shown in Table 1. We source lists of Norwegian names, towns, and healthcare providers from SSB and Helsenorge respectively, setting aside 20% of the entries in each list for the synthetic test set. The main limitation of this approach is that it relies on the model to reproduce the language used in discharge summaries. The instruction tuning process rewards longer and clearer responses from the model [10], discouraging the abbreviations and terse syntax that we expect in clinical notes.

Class	Instances (orig.)	Instances (final)
First Name	208	139
Last Name	123	55
Age	103	99
SSN	100	37
Location	100	64
Health Care Unit	121	109
Date	200	131
Phone Number	101	39
Total	**1056**	**673**

Table 1. PHI entities in our synthetic test set, before and after removing trivial examples

Class	Precision	Recall	F1
First_Name	1.0	0.892	0.943
Last_Name	1.0	0.926	0.962
Age	1.0	1.0	1.0
Date	1.0	0.975	0.987
Location	1.0	1.0	1.0
Phone_Number	1.0	1.0	1.0
SSN	1.0	1.0	1.0
Health_Care_Unit	1.0	1.0	1.0
All	**1.0**	**0.967**	**0.983**

Table 2. Comparing GPT-4's annotations on a sample of 20 generated documents to the gold standard annotations by the lead and second coauthor.

The method also introduces a possible bias in favor of GPT-4 in the deidentification experiment, as the notes' structure and content reflect its training data. The GPT-4 performance on the synthetic dataset should therefore be treated as a best case. In the current implementation, the surrogate PHI is also chosen independently with uniform probability for each category, with no regard for a diagnosis' likelihood or which patients a hospital ward sees. This leads to less realistic notes, but should not be a problem for evaluating deidentification, where it is arguably an advantage if the method does not ”second-guess” notes describing rare situations.

To estimate the annotator reliability, a sample of 20 summaries was generated, with 2122 words and 210 PHI entities total. The sample was then annotated by the first author and the first coauthor, both machine learning researchers specializing in healthcare applications. Converting the annotations to per-word labels and comparing them, as in [1], yielded a Cohen's kappa of $\kappa = 0.938$, and an accuracy of $a = 0.982$ in predicting the other annotator's annotation. The annotators then unified the annotations into a gold standard, and the model-generated annotations were compared against this standard, as seen in Table 2.

However, many of the PHI entities are placed on separate lines with distinct formatting (e.g. `"Phone Number: 770 12345"`). While many clinical notes feature preset formatting and form fields [2], using

n	Cos. sim.
1	0.359
2	0.066
3	0.017
4	0.004
5	0.001
Mean	0.089

Table 3. The average cosine similarity between document n-grams in our synthetic test set

Category	**Total**
`First_Name`	70
`Last_Name`	49
`Age`	162
`Health_Care_Unit`	42
`Phone_Number`	9
`Social_Security_Number`	5
`Date_Full`	18
`Date_Part`	45
`Location`	9
(All PHI)	409

Table 4. The PHI categories represented in the NorSynthClinical-PHI [1] dataset.

these examples for training would reward methods for recognizing these structures instead of recognizing term context. Therefore, we set up a heuristic which removes lines with PHI entities if they feature no other text than the title. This removes 380 of the 1056 examples in our synthetic test set, as shown in Table 1.

To estimate the diversity of the final synthetic dataset, we compute the average cosine similarity between the n-gram representations of the documents, calculating it separately for $n \in (1, 5)$ and taking the mean similarity as an overall score, shown in Table 3.

3.2 Validation

As our external validation, we use the NorSynthClinical-PHI [1] dataset. The dataset builds on the NorSynthClinical [11] corpus of synthetic notes on patients' family history with cardiac disease, extending it with surrogate PHI. The dataset consists of 486 sentences, with 409 entities belonging to 9 categories, as shown in Table 4. As in [1], when evaluating the methods we merge the `Date_Part` and `Date_Full` classes, as well as the `Location` and `Health_Care_Unit` classes.

4 Experiment

We now compare our deidentification method using OpenAI's GPT models to the method by Bråthen et al. [1]. They use Conditional Random Fields (CRF) [12] trained on the NorNE [13] dataset to detect names, locations and healthcare providers, and regular expressions to detect dates and numeric identifiers. Since their trained CRF model is not available, we instead implement their method with the pretrained `nb_core_news_lg` NER model by Honnibal et al. [14] in its place. We refer to this implementation as the baseline, and also report the results from [1] where appropriate. We then prompt GPT-3 [15] (`text-davinci-edit-001`) through OpenAI's Edits interface, and prompt GPT-3.5 (`gpt-3.5-turbo-0613`) and GPT-4 [16] (`gpt-4-0613`) through their Chat Completion interface. Since there is no obvious benefit to stochastic sampling when using the models for deidentification, we set the temperature parameter to 0, and only sample the most probable answer from the models.

We test the models in three different settings: multi-class annotation, single-class annotation and redaction. In the first setting, we prompt the models to do NER and measure overall class precision and recall. In the latter settings, we consider all PHI part of one entity class, simplifying the problem to distinguishing PHI from non-PHI. We do this by assigning all entities and predictions to one class post-hoc, since prompting the models to assign them to a single class directly leads to worse overall precision and recall.

5 Evaluation

NER models are conventionally evaluated on overall and per-entity precision and recall with the assumption that the model assigns entity probabilities for each token. In our setting, where we treat deidentification as a sequence-to-sequence task, we cannot get these probabilities directly, since there isn't a one-to-one correspondence between the input and output tokens.

In other works applying pretrained language models to NER tasks, this is solved by querying the model for the entity types of individual words, as in [17], or by generating an unordered list of entities which is then parsed and related back to the original input, as in [18].

While [7] report their results as overall accuracy, the associated code only checks which proportion of the PHI terms are left after deidentification, which suggests that they are more likely measuring recall.

When comparing multi-class annotations, we report per-entity accuracy with exact matches only, as [1] do. However, we report per-token accuracy when looking at single-class annotation and redaction, since our redaction evaluation method counts positives and negatives on the token level.

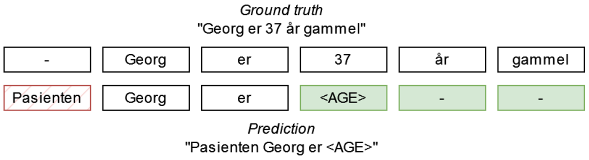

Figure 1. Using Needleman-Wunsch alignment to map the ground truth to the prediction

5.1 Estimating precision-recall in the redaction setting

To estimate precision-recall when the model redacts PHI, we propose using Needleman-Wunsch [19] alignment to align the ground truth with the redacted text, as shown in Figure 1. Currently, we only consider the single-class case, where we distinguish PHI from non-PHI – we leave the extension to general NER and pseudonymization for future work.

We use a basic scoring scheme where each match is scored with value 1, and mismatches and gaps have a penalty weight of -1. We consider any words replaced with anything other than a tag to be potential PHI, and therefore count them as false negatives.

Precision-recall estimation

Let s be the ground truth annotated text, and let d be the deidentified text where PHI should be replaced with a corresponding class tag p_c. We now count the number of true/false positives and negatives TP, FP, TN, and FP.

1. Split the strings s and d on a word level, producing the word lists $S = [w_1, \cdots, w_n] \in \mathcal{V}^*$ and $D = [d_1, \cdots, d_m] \in \mathcal{V}^*$.
2. Align S and D with Needleman-Wunsch alignment, producing the aligned lists of words and gaps $S^* \in \{\mathcal{V} \vee \texttt{<gap>}\}^*$ and $D^* \in \{\mathcal{V} \vee \texttt{<gap>}\}^*$. Then, for each pair of aligned tokens (w_i, d_i) in S^* and D^*:
    - If w_i is the gap character, count it as an *insertion.*
    - If w_i is a PHI token,
        - and d_i is a gap, count it as a TP;
        - and d_i is a PHI tag p_c, count it as a TP;
        - and d_i is w_i, count it as an FN;
        - Otherwise, count it as an FN and a *rewrite.*
    - If w_i is not a PHI token,
        - and d_i is a gap, count it as an FP and a *removal*;
        - and d_i is a PHI tag p_c, count it as an FP;
        - and d_i is w_i, count it as a TN;
        - otherwise, count it as an FN and a *rewrite.*

Method	Precision	Recall	F1
Baseline	0.585	0.713	0.643
GPT-4	0.789	0.822	0.805
GPT-3.5	0.701	0.647	0.673
GPT-3	0.620	0.653	0.636

Table 5. The results of annotation/NER on our synthetic test set, for exact entity matches.

Method	Precision	Recall	F1
Baseline	0.882	0.756	0.815
GPT-4	0.818	0.892	0.854
GPT-3.5	0.730	0.882	0.799
GPT-3	0.647	0.891	0.745

Table 6. The results of single-class annotation on our synthetic test set, counted per token.

This method gives us a way of estimating precision-recall in a sequence-to-sequence deidentification, and indicates how much the model rewrites or erroneously inserts text.

6 Results

Table 5 shows the results of annotating our synthetic test set with the GPT models. Here, GPT-4 manages to recover the annotations on the synthetic notes it generates, achieving the best overall $F_1 = 0.805$. Table 6 and 7 compare single-class annotation and redaction on the synthetic test set. Surprisingly, the GPT models perform worse when redacting terms rather than adding annotations, with all models underperforming the baseline's $F_1 = 0.695$. We then prompt the GPT models to perform NER on NorSynthClinical-PHI, listing the results in Table 8. Here, GPT-4 was unable to outperform the baseline $F_1 = 0.604$ and the $F_1 = 0.731$ reported in [1]. The per-class metrics for GPT-4 are listed in Table 9.

We also compare single-class annotation and redaction on NorSynthClinical-PHI in Table 10 and 11, respectively. Here, GPT-4 achieves a better F_1 score than our baseline, attaining $F_1 = 0.859$ when redacting PHI and $F_1 = 0.841$ when annotating it.

7 Discussion

When generating synthetic discharge summaries, GPT-4 is able to coordinate multiple requirements, annotating PHI while generating the summary. In our sample in Table 2, the generated annotations

Method	Precision	Recall	F1
Baseline	0.633	0.771	0.695
GPT-4	0.467	0.461	0.464
GPT-3.5	0.429	0.401	0.414
GPT-3	0.410	0.400	0.405

Table 7. The results of redaction on our synthetic test set.

Method	Precision	Recall	F1
Bråthen [1]	0.893	0.619	0.731
Baseline	0.721	0.520	0.604
GPT-4	0.448	0.615	0.518
GPT-3.5	0.453	0.359	0.401
GPT-3	0.380	0.285	0.326

Table 8. The results of annotation/NER on NorSynthClinical-PHI for exact entity matches.

Class	Precision	Recall	F1
First_Name	0.886	0.861	0.873
Last_Name	0.878	0.956	0.915
Age	0.206	0.352	0.260
SSN	0.600	0.750	0.666
Location/ Health_Care_Unit	0.370	0.558	0.444
Date	0.633	0.746	0.685
Phone_Number	0.200	0.333	0.250

Table 9. The per-class precision and recall of GPT-4 on NorSynthClinical-PHI.

Method	Precision	Recall	F1
Baseline	0.805	0.593	0.683
GPT-4	0.851	0.830	0.841
GPT-3.5	0.826	0.552	0.662
GPT-3	0.683	0.462	0.551

Table 10. The results of single-class annotation on NorSynthClinical-PHI, counted per token.

Method	Precision	Recall	F1
Baseline	0.807	0.594	0.684
GPT-4	0.808	0.916	0.859
GPT-3.5	0.662	0.795	0.723
GPT-3	0.221	0.117	0.153

Table 11. The results of redaction on NorSynthClinical-PHI, counted per token.

highly agree with the human annotations: none of the generated annotations were mislabelled, but a small number of generated entities were not labelled. Simultaneously, there is room to improve the diversity of the generated notes: the notes end up following a similar format with certain headlines and recurring words, as indicated by the unigram cosine similarity of 0.359. We also note that low n-gram similarity does not guarantee semantic diversity [20].

When evaluating the models on multi-class deidentification with exact entity matching, we find that the GPT models do not beat the classifier in [1] or our baseline implementation of it. However, when the requirements are relaxed to recognizing PHI on a token level, GPT-4 achieves a better F_1 score than our baseline on both datasets. We also found that prompting the GPT models to directly assign terms to a common PHI class led to lower precision and recall than prompting them with multiple classes, as in Figure 2, and assigning the entities to a single class afterwards. This suggests that the models' performance does not entirely depend on in-context learning, but also depends on similar NER tasks being present during pre-training or instruction tuning.

We were not able to clearly show whether redaction or annotation is the better framing of the task: while GPT-3.5 and GPT-4 both see improvement in redacting over annotating PHI on NorSynthClinical-PHI, they do worse when redacting the synthetic test set. These results support that zero-shot named entity recognition is a challenging task for current language models, especially decoder-only transformer models [21]. Table 9 suggests that GPT-4 performs well when recognizing names and dates, but does worse on tasks which require pure pattern recognition (e.g. phone numbers) and specific domain knowledge (names of healthcare providers).

In every experiment, the GPT models' performance on the task were consistent with their size and general capabilities. This suggests that deidentification could be a good task to evaluate language models, especially those tuned for healthcare applications. Future work includes evaluating language models with openly available weights (e.g. Llama 2 [22]) on the n2c2 benchmark datasets [8][23], and investigating whether guided generation methods [24] improve their performance. We will also investigate the performance of supervised models trained on our synthetic dataset.

8 Conclusion

We use GPT-4 to generate a synthetic Norwegian deidentification dataset, consisting of 1200 synthetic summaries. Taking a sample of 20 generated documents, we found that the generated annotations highly agree with human annotations,

Anonymize the following clinical note with tags. Replace first names with <First_Name> tags. Replace last names with <Last_Name> tags. Replace any strings that might be a location or address, such as ”Åssiden 31” with <Location> tags. Replace clinical and hospital names with <Location> tags. Replace the patient’s age and any texts that look like ”X år gammel” with <Age> tags. Replace phone numbers with <Phone_Number> tags. Replace 8 digit long numbers with <Phone_Number> tags. Replace social security numbers with <Social_Security_Number> tags. Replace 11 digit long numbers with <Social_Security_Number> tags. Replace dates and times with <Date> tags. Do not use any tags which were not specified above.
Example:
Georg Nordmann er 47 år gammel og innlagt på Haukeland siden 3. april . Georgs kone Åshild ønsker at vi ringer henne på telefon 770 12345 når vi vet mer .
Result:
<First_Name> <Last_Name> er <Age> og innlagt på <Location> siden <Date> . <First_Name> kone <First_Name> ønsker at vi ringer henne på telefon <Phone_Number> når vi vet mer .

Figure 2. The prompt used to redact text. A similar prompt is used to generate annotations, with instructions to enclose the terms with corresponding class tags.

with $F_1 = 0.983$. We make the dataset and the code to generate it available at https://github.com/UNN-SPKI/Nor-DeID-SynthData, and the code for evaluations available at https://github.com/UNN-SPKI/Nor-DeID-Evaluation.

Then, we apply the GPT models to the deidentification problem itself, comparing their performance to the baseline method by Bråthen et al. [1] over our synthetic dataset and the NorSynthClinical-PHI set. We find that the GPT models underperform the baseline method on multi-class deidentification, suggesting that zero-shot named entity recognition is still a challenging problem for large language models. However, for single-class anonymization, GPT-4 achieves better precision and recall than our CRF- and rule-based baseline method.

Acknowledgements

This work was funded by The Research Council of Norway (RCN) through its industrial Ph.D. program [project number 327520] and by DIPS AS.

Utskrivningsnotat
Alder: <Age>75</Age> år
<First_Name>Steinar</First_Name> ble innlagt på <Health_Care_Unit>UNN, Narvik Sykehus</Health_Care_Unit> den <Date>15. april 2015</Date> med hoveddiagnosekode S82425C, uforflyttet tverrbrudd i venstre fibula. Ved innleggelse hadde pasienten senket hjertefrekvens, klare tegn på dehydrering, begrenset mobilitet, moderat smerte, lav kjernetemperatur, overfladisk pusting, høyt blodtrykk.
Utskrivningsstatus: Pasienten har forbedret seg i løpet av oppholdet. Hydrering er oppnådd, og hjertefrekvensen har stabilisert seg. Pusten er fortsatt noe grunn, og blodtrykket er høyt, men under bedre kontroll. Pasientens temperatur er fortsatt lav, men stabilt. Pasienten har fortsatt begrenset mobilitet på grunn av den uforflyttede transversale frakturen i skaftet av venstre fibula. Smerter er moderate, men kontrollerbare med smertestillende medisin.

Figure 3. Generated Norwegian discharge summary describing a fictional patient who has been treated for a fracture. Personally identifying information – e.g. the patient’s age, name and admission date – is annotated by the model at generation time.

References

[1] S. Bråthen, W. Wie, and H. Dalianis. “Creating and Evaluating a Synthetic Norwegian Clinical Corpus for De-Identification”. In: *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*. Reykjavik, Iceland (Online): Linköping University Electronic Press, Sweden, May 2021, pp. 222–230. URL: https://aclanthology.org/2021.nodalida-main.22.

[2] H. Dalianis. *Clinical Text Mining: Secondary Use of Electronic Patient Records*. Springer International Publishing, 2018. ISBN: 978-3-319-78502-8. DOI: 10.1007/978-3-319-78503-5. URL: https://www.springer.com/gp/book/9783319785028 (visited on 04/20/2021).

[3] A. E. W. Johnson, L. Bulgarelli, L. Shen, A. Gayles, A. Shammout, S. Horng, T. J. Pollard, S. Hao, B. Moody, B. Gow, L.-w. H. Lehman, L. A. Celi, and R. G. Mark. “MIMIC-IV, a freely accessible electronic health record dataset”. In: *Scientific Data* 10.1 (Jan. 3, 2023). Number: 1 Publisher: Nature Publishing Group, p. 1. ISSN: 2052-4463. DOI: 10.1038/s41597-022-01899-x. URL: https://www.nature.com/articles/s41597-022-01899-x (visited on 09/15/2023).

[4] V. Pezoulas, T. Exarchos, and D. I. Fotiadis. *Medical data sharing, harmonization and analytics*. Academic Press, 2020. URL: https://shop.elsevier.com/books/medical-data-sharing-harmonization-and-analytics/pezoulas/978-0-12-816507-2.

[5] H. Berg, A. Henriksson, U. Fors, and H. Dalianis. "De-identification of Clinical Text for Secondary Use: Research Issues." In: *HEALTHINF* (2021), pp. 592–599. URL: https://www.scitepress.org/Papers/2021/103187/103187.pdf.

[6] T. Chomutare. "Clinical Notes De-Identification: Scoping Recent Benchmarks for n2c2 Datasets". In: *Studies in Health Technology and Informatics* 289 (Jan. 14, 2022), pp. 293–296. ISSN: 1879-8365. DOI: 10.3233/SHTI210917.

[7] Z. Liu, X. Yu, L. Zhang, Z. Wu, C. Cao, H. Dai, L. Zhao, W. Liu, D. Shen, Q. Li, T. Liu, D. Zhu, and X. Li. *DeID-GPT: Zero-shot Medical Text De-Identification by GPT-4*. Mar. 20, 2023. arXiv: 2303.11032[cs]. URL: http://arxiv.org/abs/2303.11032 (visited on 03/21/2023).

[8] A. Stubbs, C. Kotfila, and Ö. Uzuner. "Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/UTHealth shared task Track 1". In: *Journal of Biomedical Informatics*. Supplement: Proceedings of the 2014 i2b2/UTHealth Shared-Tasks and Workshop on Challenges in Natural Language Processing for Clinical Data 58 (Dec. 1, 2015), S11–S19. ISSN: 1532-0464. DOI: 10.1016/j.jbi.2015.06.007. URL: https://www.sciencedirect.com/science/article/pii/S1532046415001173 (visited on 07/10/2023).

[9] K. Huang, J. Altosaar, and R. Ranganath. *ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission*. arXiv:1904.05342. type: article. arXiv, Nov. 28, 2020. DOI: 10.48550/arXiv.1904.05342. arXiv: 1904.05342[cs]. URL: http://arxiv.org/abs/1904.05342 (visited on 09/15/2023).

[10] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Christiano, J. Leike, and R. Lowe. "Training language models to follow instructions with human feedback". In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh. Vol. 35. Curran Associates, Inc., 2022, pp. 27730–27744. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

[11] T. Rama, P. Brekke, Ø. Nytrø, and L. Øvrelid. "Iterative development of family history annotation guidelines using a synthetic corpus of clinical text". In: *Proceedings of the Ninth International Workshop on Health Text Mining and Information Analysis*. Louhi 2018. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 111–121. DOI: 10.18653/v1/W18-5613. URL: https://aclanthology.org/W18-5613 (visited on 09/15/2023).

[12] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data". In: *Proceedings of the Eighteenth International Conference on Machine Learning*. ICML '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 282–289. ISBN: 1558607781.

[13] F. Jørgensen, T. Aasmoe, A.-S. Ruud Husevåg, L. Øvrelid, and E. Velldal. "NorNE: Annotating Named Entities for Norwegian". English. In: *Proceedings of the Twelfth Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, May 2020, pp. 4547–4556. ISBN: 979-10-95546-34-4. URL: https://aclanthology.org/2020.lrec-1.559.

[14] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd. "spaCy: Industrial-strength Natural Language Processing in Python". In: (2020). DOI: 10.5281/zenodo.1212303.

[15] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. URL: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html (visited on 03/06/2023).

[16] OpenAI. *GPT-4 Technical Report*. arXiv:2303.08774. type: article. arXiv, Mar. 27, 2023. DOI: 10.48550/arXiv.2303.08774. arXiv: 2303.08774[cs]. URL: http:

//arxiv.org/abs/2303.08774 (visited on 09/15/2023).

[17] A. Davody, D. I. Adelani, T. Kleinbauer, and D. Klakow. “TOKEN Is a MASK: Few-shot Named Entity Recognition with Pre-trained Language Models”. In: *Text, Speech, and Dialogue: 25th International Conference, TSD 2022, Brno, Czech Republic, September 6-9, 2022, Proceedings.* Berlin, Heidelberg: Springer-Verlag, Sept. 5, 2022, pp. 138–150. ISBN: 978-3-031-16269-5. DOI: 10.1007/978-3-031-16270-1_12. URL: https://doi.org/10.1007/978-3-031-16270-1_12 (visited on 07/10/2023).

[18] L. Wang, R. Li, Y. Yan, Y. Yan, S. Wang, W. Wu, and W. Xu. *InstructionNER: A Multi-Task Instruction-Based Generative Framework for Few-shot NER.* Mar. 8, 2022. arXiv: 2203.03903[cs]. URL: http://arxiv.org/abs/2203.03903 (visited on 04/04/2023).

[19] S. B. Needleman and C. D. Wunsch. “A general method applicable to the search for similarities in the amino acid sequence of two proteins”. In: *Journal of Molecular Biology* 48.3 (Mar. 28, 1970), pp. 443–453. ISSN: 0022-2836. DOI: 10.1016/0022-2836(70)90057-4. URL: https://www.sciencedirect.com/science/article/pii/0022283670900574 (visited on 09/15/2023).

[20] G. Tevet and J. Berant. “Evaluating the Evaluation of Diversity in Natural Language Generation”. In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume.* Online: Association for Computational Linguistics, Apr. 2021, pp. 326–346. DOI: 10.18653/v1/2021.eacl-main.25. URL: https://aclanthology.org/2021.eacl-main.25.

[21] S. Wu, O. Irsoy, S. Lu, V. Dabravolski, M. Dredze, S. Gehrmann, P. Kambadur, D. Rosenberg, and G. Mann. *BloombergGPT: A Large Language Model for Finance.* arXiv:2303.17564. type: article. arXiv, May 9, 2023. DOI: 10.48550/arXiv.2303.17564. arXiv: 2303.17564[cs,q-fin]. URL: http://arxiv.org/abs/2303.17564 (visited on 09/15/2023).

[22] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom. *Llama 2: Open Foundation and Fine-Tuned Chat Models.* arXiv:2307.09288. type: article. arXiv, July 19, 2023. DOI: 10.48550/arXiv.2307.09288. arXiv: 2307.09288[cs]. URL: http://arxiv.org/abs/2307.09288 (visited on 09/15/2023).

[23] Ö. Uzuner, Y. Luo, and P. Szolovits. “Evaluating the State-of-the-Art in Automatic De-identification”. In: *Journal of the American Medical Informatics Association* 14.5 (Sept. 1, 2007), pp. 550–563. ISSN: 1067-5027. DOI: 10.1197/jamia.M2444. URL: https://doi.org/10.1197/jamia.M2444 (visited on 07/10/2023).

[24] B. T. Willard and R. Louf. *Efficient Guided Generation for Large Language Models.* arXiv:2307.09702. version: 1 type: article. arXiv, July 18, 2023. DOI: 10.48550/arXiv.2307.09702. arXiv: 2307.09702[cs]. URL: http://arxiv.org/abs/2307.09702 (visited on 09/15/2023).