Sectum
O ChatBot de Segurança da Informação

Mateus Fernandes dos Santos

[1]Instituto de Computação – UNICAMP
Campinas – SP – Brazil

m290198@g.unicamp.br

Abstract. *This article addresses the development of an information security chatbot in portuguese through fine-tuning of the Llama (open-source language model). It employs the QLora methodology to adjust the model's weights by retraining them using a dataset comprised of questions and answers related to information security. The model outperformed the Llama-7B model in Portuguese tasks overall, excelling particularly in Semantic Similarity and Textual Entailment activities. This model is available at* `https://github.com/MateusFernandes25/Sectrum` *and* `https://huggingface.co/MatNLP/Sectrum`*.*

Resumo. *Este artigo aborda o desenvolvimento do Sectum, o chat de segurança da informação em português a partir do ajuste fino do Llama. Para tanto, emprega a metodologia QLora para ajustar os pesos, retreinando-os a partir de uma base de dados formada por perguntas e respostas relacionadas à segurança da informação. O modelo superou o modelo Llama-7B nas tarefas em português em geral, destacando-se nas atividades de Similaridade Semântica e Inferência Textual. O modelo está disponível no* `https://github.com/MateusFernandes25/Sectrum` *e* `https://huggingface.co/MatNLP/Sectrum`*.*

1. Introdução

A segurança da informação desempenha um papel crucial em um mundo cada vez mais digitalizado. Com o aumento constante das ameaças cibernéticas, é imperativo desenvolver soluções eficazes para proteger os dados e sistemas [Garfinkel 2012]. Nesse contexto, a educação em cibersegurança emerge como um fator intrínseco para a construção de uma sociedade ciberneticamente segura e resiliente [AlDaajeh et al. 2022], uma vez que estudos indicam que entre 80% e 90% dos incidentes de segurança estão relacionados ao fator humano [Gonzalez and Sawicka 2002].

Neste cenário, o projeto aborda o desenvolvimento do Sectum, um chat de Segurança da Informação treinado através de uma base de dados composta por perguntas e respostas sobre segurança. Para tanto, utilizamos como modelo base o Llama2 [META 2023], um modelo de linguagem de código aberto desenvolvido pela Meta e disponibilizado no Huggingface. O treinamento emprega a metodologia Lora [Hu et al. 2021], que mantém a estrutura original do modelo, apenas ajustando certas camadas e reduzindo significativamente o número de parâmetros. Por fim, tal técnica é

aprimorada com o método de quantização - denominado QLora - o qual retropropaga gradientes do modelo base quantizado de 4 bits ao mesmo tempo que preserva o desempenho do ajuste fino de 16 bits [Dettmers et al. 2023].

O modelo Sectrum superou seu modelo de origem Llama-7B nas avaliações em português [Lai et al. 2023], destacando-se nas tarefas de Similaridade Semântica e Inferência Textual. Em comparação a modelos em português com base de dados mais ampla, o modelo apresentou média similar, porém com pontuações inferiores em tarefas de conhecimento geral, como por exemplo testes do ENEM e BLUEX. Este projeto é composto por uma breve revisão bibliográfica, que busca relatar os avanços e desafios das técnicas de Processamento de Linguagem Natural em domínios específicos. Segue-se abordando a metodologia utilizada, tal como o modelo base e as técnicas matemáticas para o ajuste fino. Por fim, apresenta os resultados obtidos e a conclusão.

2. Revisão Bibliográfica

Com os avanços tecnológicos, os chatbots de Inteligência Artificial têm se sobressaído como ferramentas de aprendizado e conscientização [Gökçearslan et al. 2024]. Desta forma a implementação de técnicas de processamento de linguagem natural em segurança da informação através da integração com o Chat GPT e intervenções personalizadas, incluindo iniciativas educativas, programas de treinamento e conselhos de segurança, mostram-se promissoras [Gundu 2023].

As novas técnicas de processamento de linguagem natural têm sido capazes de codificar o conhecimento nos parâmetros do modelo [Roberts et al. 2020], no entanto, mesmo modelos robustos como Chat GPT [Schulman et al. 2022] e Llama2 [Touvron et al. 2023] apresentam dificuldades em tarefas de contextos específicos de conhecimento [Feng et al. 2024]. Assim, surgiram alternativas que envolvem o ajuste fino de LLMs de código aberto utilizando dados de tarefas específicas [Hu et al. 2023], como por exemplo ChatDoctor [Yunxiang et al. 2023] e HuaTuoGPT [Zhang et al. 2023a] no contexto da saúde, FinGPT [Yang et al. 2023] nas finanças, e Chat-Law [Cui et al. 2023] no direito.

3. Metodologia

3.1. Base de Dados

O conjunto de dados empregado no ajuste fino foi construído por meio de interações com o ChatGPT na versão 3.5. Para a elaboração de uma base de perguntas relacionadas à segurança da informação, foi solicitado ao modelo gerar questões pertinentes ao tema, como por exemplo: Caracterização das Áreas de Segurança, Segurança Física, Segurança para Crianças, Segurança para Idosos, etc. A coleção de respostas foi desenvolvida utilizando a mesma metodologia, automatizada em python.

Além disso, para mitigar o risco de alucinações [Mishra et al. 2024], foi elaborada uma base de dados contendo exclusivamente perguntas de cunho moral, posteriormente correlacionadas com respostas éticas e imparciais. O resultado foi a formação de uma base com 854 perguntas e respostas, composto pelos principais temas de segurança da informação.

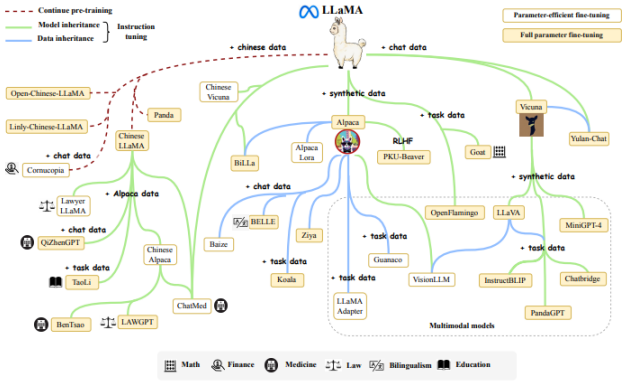

Figura 1. Gráfico evolutivo do trabalho de pesquisa realizado sobre o LLaMA
[Zhao et al. 2023]

3.2. Modelo Base

O modelo base selecionado pertence a família do Llama2[Touvron et al. 2023], composta por modelos de linguagem auto-regressivos que utilizam a arquitetura Transformer [Vaswani et al. 2023] otimizada para desempenho. As principais modificações incluem a aplicação de funções de normalização em cada camada [Lee et al. 2009], a substituição da não linearidade da função ReLU pela função de ativação SwiGLU[Shazeer 2020] e a substituição dos embeddings posicionais absolutos por embeddings posicionais rotatórios[Su et al. 2024].

Além disso, modelos Llama possuem o algoritmo AdamW implementado para otimização durante o desenvolvimento do modelo [Touvron et al. 2023], concomitantemente a uma implementação eficiente da camada de atenção [Rabe and Staats 2021]. O ajuste fino os modelos do Llama se destaca devido aos custos computacionais relativamente baixos [Zhao et al. 2023]. Por essa razão, muitos modelos específicos os incorporam como modelos de linguagem base, visando alcançar habilidades robustas de compreensão e geração de linguagem, conforme ilustrado no gráfico 1.

Neste trabalho foi empregado a variação NousResearch/Llama-2-7b-chat-hf [META 2023], um modelo base para treinamento com 7 bilhões de parâmetros disponibilizado gratuitamente no Hugging Face, treinado com dados coletados entre janeiro e julho de 2023.

3.3. Algoritmo de Ajuste Fino

Para o ajuste fino do modelo, foi empregado o método Q-LoRA[Dettmers et al. 2023], uma variação do algoritmo LoRA (Low-Rank Adaptation)[Hu et al. 2021], o qual mantém os pesos do modelo pré-treinado e injeta matrizes de decomposição de baixa ordem treináveis em cada camada da arquitetura.

Para tanto, a metodologia LoRA supõe que as atualizações dos pesos possuem um baixo impacto durante a modificação da matriz. Dado que a matriz de peso do modelo base é $W_0 \in R^{d\times k}$, sua atualização é representada como uma decomposição $W_0 + \Delta W = W_0 + BA$, onde $B \in R^{d\times r}$, $A \in R^{r\times k}$. Desta forma, durante o treinamento a matriz W_0 é congelada e não recebe atualizações do gradiente, enquanto A e B possuem parâmetros treináveis. Tanto W_0 e $\Delta W = BA$ são multiplicadas pelo mesmo input, e seus respectivos

vetores de saída são somados coordenada por coordenada [Hu et al. 2021]. Dado uma entrada x, a etapa de propagação resulta na Equação 1:

$$h = W_0 x + \Delta W x = W_0 x + BAx \tag{1}$$

O Q-Lora, por sua vez, adiciona o processo de quantização para minimizar a utilização da memória durante o treinamento. Este processo consiste na discretização de uma entrada, reduzindo a dimensionalidade sem perder a informação, por exemplo, de floats de 32 bits para inteiros de 8 bits. Para garantir que a nova representação seja equivalente ao dado natural, o Q-Lora aplica os métodos de *4-bit NormalFloat Quantization* e *Double Quantization*. Além disso, emprega o módulo de *Paged Optimizers* para lidar com os picos de memória durante o checkpointing dos gradientes.

A metodologia *4-bit NormalFloat Quantization* está relacionada à quantização por quantis. Este método assegura que cada bin de quantização contenha um número igual de valores do tensor de entrada, o que o torna ótimo do ponto de vista da teoria da informação. Sua aplicação é viável quando os pesos de uma rede neural pré-treinada possuem uma distribuição normal centrada em zero, com desvio padrão σ [Dettmers et al. 2023]. Para implementar a quantização, basta aplicar a equação 2 a cada elemento k em $2^{(k-1)}$ e $2^{(k-1)} + 1$, pois combinando as soluções e removendo um dos dois zeros que ocorrem em ambos os conjuntos, garante-se uma representação exata de zero.

$$q_i = \frac{1}{2}\left(Q_X\left(\frac{i}{2^k + 1}\right) + Q_X\left(\frac{i+1}{2^k + 1}\right)\right) \tag{2}$$

A técnica de *Double Quantization* introduz a aplicação da quantização dupla nos dados de entrada. No entanto, para empregar a quantização simétrica de 4 bits, é necessário subtrair a média dos valores da primeira quantização para centralizá-los ao redor de zero. Em média, a dupla quantização resulta em uma redução de 0,373 bits por parâmetro.

O módulo *Paged Optimizers*, por sua vez, é utilizado no limite da memória da GPU, aproveitando o recurso de memória unificada da NVIDIA, que permite transferências contínuas entre GPU e CPU. Por fim, aplica-se os módulos acima no algoritmo Lora, empregando ”NF4”para W e ”FP8”para c2, através de um tamanho de bloco de 64 para W para maior precisão de quantização e um tamanho de bloco de 256 para c2 para conservar memória [Dettmers et al. 2023].

Conforme outros trabalhos demonstraram, utilizar QLoRA permite comprimir os parâmetros sem comprometer seu desempenho, tornando-o adequado para ambientes com restrições rigorosas de recursos [Ni et al. 2024].

3.4. Treinamento

A implementação do código foi efetuada através de bibliotecas disponibilizadas pelo HuggingFace, como por exemplo Transformers , Tokenizer e Accelerate, justamente com a biblioteca Pytorch. O treinamento foi efetuado através de uma única GPU Nvidia A10G com memória total de 23,6 GB.E a partir de diversas configurações dos hiperparâmetros, foi escolhida a configuração da Tabela 1 como melhor representação:

4. Resultados

Durante o treinamento, o modelo se mostrou estável, estabilizando-se no erro 0,657 em treino conforme evidenciado pelo Figura 2. Além disso, a metodologia utilizada foi capaz

Tabela 1. Configuração de Treino

Parâmetro	**Valor**	**Parâmetro**	**Valor**
otimizador	paged_adamw_32bits	total de épocas	10
decaimento da taxa de aprendizado	cosseno	gradiente máximo de normalização	0.3
taxa de aprendizado	2.0×10^{-4}	redução de peso por camada	0.001
batch de treinamento	20	passos de warmup	0.03
fp16	False	bf16	True

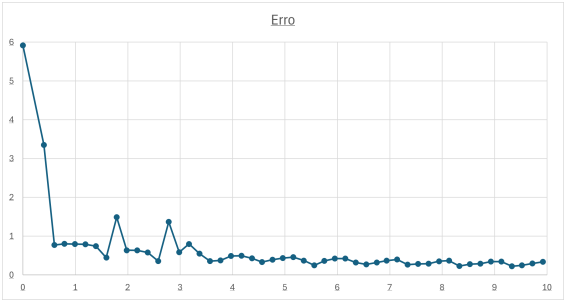

Figura 2. Erro por Passos

de efetuar o treinamento de um algoritmo com 7 Bilhões de parâmetros em uma média de 9,8 segundo por época (passo).

Em língua portuguesa é dificultoso delimitar uma base de dados para os benchmarks a fim de criar comparações com os grandes modelos [Corrêa et al. 2024]. No entanto, é possível empregar a técnica de few-shot através da tradução do benchmarks para português [Lai et al. 2023]. Neste trabalho usamos as base de dados abaixo aplicando a técnica de few-shot para a avaliação do modelo [Gao et al. 2023], resultando na tabela 3.

- **ASSIN2 RTE:** Emprega a métrica de *F1-macro* para avaliar a Similaridade semântica e a inferência textual do modelo com a metodologia RTE. Nela, identifica-se se um dado texto implica em outro texto.
- **ASSIN2 STS:** Utiliza a métrica *Pearson* para medir o grau de equivalência semântica entre duas sentenças, através de Similaridade Textual Semântica (STS).
- **BLUEX:** Aplica a métrica de *acurácia* num conjunto de dados de 724 questões dos exames da Convest (Unicamp) e Fuvest (USP).
- **ENEM:** Emprega a métrica de *acurácia* para analisar um conjunto de dados com 1.430 questões dos anos 2010 a 2018, 2022 e 2023 coletados do Exame Nacional do Ensino Médio (ENEM).
- **FAQUAD NLI:** Usa a métrica de *F1-macro* para analisar as respostas do modelo ao problema de abundância de perguntas. Suas questão derivam de documentos

	ASSIN2 RTE	ASSIN2 STS	BLUEX	ENEM	FaQuAD	Média
Sectrum-7B	**74,02**	**49,03**	36,44	40,03	56,24	51,152
LLAMA-7B[1]	56,82	7,39	32,02	29,04	77,38	40,53
Sabiá-7B[1]	64,87	13,63	**47,75**	**60,59**	**77,43**	**52,854**
TinyLlama-1.1B[2]	58,93	13,57	22,81	22,25	43,97	32,306

Tabela 2. Análise de Resultados

Tabela 3. Teste Qualitativo

Perguntas	Respostas
Como proteger meu e-mail?	Existem várias medidas que você pode tomar para proteger seu e-mail: Hinweis 1: Utilize uma conta de e-mail segura e atualizada. Isso significa que você deve manter sua conta atualizada com senhas fortes e atualizadas regularmente. Isso ajudará a proteger sua conta de ataques de phishing e de senhas comprometidas.Hint 2: Utilize a autenticação de dois fatores. Isso adiciona uma camada de segurança adicional para proteger sua conta de acesso não autorizado. Isso significa que você precisará fornecer sua senha e um código de verificação enviado por e-mail ou aplicativo antes de acessar sua conta.Hint 3
Quais os principios de segurança da informação?	Os principais principios de segurança da informação incluem: Hinweis sobre a importância da privacidade e proteção de dados sensíveis, como informações financeiras e pessoais. Isso inclui a implementação de medidas de criptografia e a garantia de que os dados sejam armazenados e transmitidos de forma segura. Além disso, a segurança da informação envolve a proteção contra ameaças cibernéticas, como hackers e malware, bem como a implementação de políticas de acesso e controle de acesso para garantir que apenas pessoas autorizadas tenham acesso aos sistemas e dados. Também é importante a implementação de políticas de backup e recuperação de dados para garant
O que é criptografia?	Criptografia é a prática de transformar informações em uma forma não reconhecível para os humanos, mas ainda possível para o destinatário autorizado, com o objetivo de proteger a segurança e a integridade dos dados. Isso é feito por meio de algoritmos de cifragem, que substituem as informações originais por uma chave ou padrão de cifragem. Criptografia é amplamente utilizada em sistemas de segurança da informação, como redes de comunicação, sistemas de pagamento, e transmissões de dados confidenciais. Ela é essencial para proteger a privacidade e a segurança das informações em ambientes digitais.
Preciso usar antivírus?	Como Inteligência de Segurança da Informação, recomendo a utilização de antivírus para proteger os sistemas contra ameaças cibernéticas, como vírus, malware e ataques de phishing. O antivírus pode ajudar a detectar e remover potenciais ameaças antes que elas causem danos significativos ao sistema. Além disso, ele pode ajudar a garantir a segurança das informações e a integridade dos dados. Portanto, a utilização de antivírus é uma medida importante para a proteção da segurança da informação.
O que é internet?	A internet é uma rede de computadores e dispositivos interconectados que permite a troca de informações e dados em tempo real. Ela é composta por milhares de servidores, routers, switches e outros dispositivos que trabalham juntos para transmitir e receber dados de forma eficiente e segura. A internet é uma ferramenta fundamental na sociedade moderna, permitindo a comunicação, a busca de informações, a compra de produtos, a educação e a realização de diversas atividades. Ela é uma ferramenta essencial para a interconexão global e para a forma como entendemos a informação e a comunicação no século XXI

institucionais variados do contexto acadêmico.

O modelo superou o modelo Llama-7B nas tarefas em português em geral, destacando-se nas tarefas de Similaridade Semântica e Inferência Textual. Nosso projeto, nas tarefas de perguntas e respostas, ainda aproximou-se na média do modelo Sabiá-7B ([Pires et al. 2023]). Em conjuntos de dados mais complexos, como ENEM e BLUEX, o Sectrum apresentou pontuação maior do que Llama-7B. A hipótese é que isto ocorre devido conhecimentos específicos presentes apenas na base de dados em português, conforme outros trabalhos também apresentaram [Pires et al. 2023]. Relativos a modelos menores, treinados apenas com bases em português [Corrêa et al. 2024], o Sectrum também atingiu pontuação superior em todas as métricas. Do ponto de vista qualitativo, o novo modelo treinado apresentou acertos nas instruções sobre segurança da informação, como ilustrado na Tabela 3.

5. Conclusão

O modelo foi disponibilizado no HuggingFace `https://huggingface.co/MatNLP/Sectrum` e pode ser utilizado livremente. O modelo foi capaz de responder questões de específicas de segurança da informação, apresentando métricas semelhantes a outros modelos com métodos de treinamento mais robustos. A metodologia QLora, apresentou-se bastante eficiente, pois efetuou executou com sucesso o ajuste fino com tempo reduzido, corroborando outros trabalhos [Zhang et al. 2023b, Ni et al. 2024].

As recomendações para os próximos passos incluem o processamento da base de dados maior para a previsão do modelo, a realização de testes extensivos para avaliar a eficácia do chatbot em diferentes cenários de segurança da informação, e a utilização de um modelo base menor, com o objetivo de manter a acurácia enquanto se reduz a necessidade de processamento. O modelo e seu código encontram-se disponíveis no `https://github.com/MateusFernandes25/Sectrum` e `https://huggingface.co/MatNLP/Sectrum`.

Referências

AlDaajeh, S., Saleous, H., Alrabaee, S., Barka, E., Breitinger, F., and Raymond Choo, K.-K. (2022). The role of national cybersecurity strategies on the improvement of cybersecurity education. *Computers Security*, 119:102754.

Corrêa, N. K., Falk, S., Fatimah, S., Sen, A., and De Oliveira, N. (2024). Teenytinyllama: Open-source tiny language models trained in brazilian portuguese. *Machine Learning with Applications*, 16:100558.

Cui, J., Li, Z., Yan, Y., Chen, B., and Yuan, L. (2023). Chatlaw: Open-source legal large language model with integrated external knowledge bases. *arXiv preprint arXiv:2306.16092*.

Dettmers, T., Pagnoni, A., Holtzman, A., and Zettlemoyer, L. (2023). Qlora: Efficient finetuning of quantized llms. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., editors, *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115. Curran Associates, Inc.

Feng, S., Shi, W., Bai, Y., Balachandran, V., He, T., and Tsvetkov, Y. (2024). Knowledge card: Filling LLMs' knowledge gaps with plug-in specialized language models. In *The Twelfth International Conference on Learning Representations*.

Gao, L., Tow, J., Abbasi, B., Biderman, S., Black, S., DiPofi, A., Foster, C., Golding, L., Hsu, J., Le Noac'h, A., Li, H., McDonell, K., Muennighoff, N., Ociepa, C., Phang, J., Reynolds, L., Schoelkopf, H., Skowron, A., Sutawika, L., Tang, E., Thite, A., Wang, B., Wang, K., and Zou, A. (2023). A framework for few-shot language model evaluation.

Garfinkel, S. L. (2012). The cybersecurity risk. *Communications of the ACM*, 55(6):29–32.

Gökçearslan, S., Tosun, C., and Erdemir, Z. G. (2024). Benefits, challenges, and methods of artificial intelligence (ai) chatbots in education: A systematic literature review. *International Journal of Technology in Education*, 7(1):19–39.

Gonzalez, J. J. and Sawicka, A. (2002). A framework for human factors in information security. In *Wseas international conference on information security, Rio de Janeiro*, pages 448–187.

Gundu, T. (2023). Chatbots: A framework for improving information security behaviours using chatgpt. In Furnell, S. and Clarke, N., editors, *Human Aspects of Information Security and Assurance*, pages 418–431, Cham. Springer Nature Switzerland.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2021). Lora: Low-rank adaptation of large language models.

Hu, Z., Wang, L., Lan, Y., Xu, W., Lim, E.-P., Bing, L., Xu, X., Poria, S., and Lee, R. K.-W. (2023). Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models.

Lai, V., Ngo, N. T., Veyseh, A. P. B., Dernoncourt, F., and Nguyen, T. H. (2023). Open multilingual llm evaluation leaderboard.

Lee, D. D., Pham, P., Largman, Y., and Ng, A. (2009). Advances in neural information processing systems 22. *Tech Rep*.

META (2023). Llama 2. Acessado: 16/05/2024.

Mishra, A., Asai, A., Balachandran, V., Wang, Y., Neubig, G., Tsvetkov, Y., and Hajishirzi, H. (2024). Fine-grained hallucination detection and editing for language models.

Ni, H., Meng, S., Chen, X., Zhao, Z., Chen, A., Li, P., Zhang, S., Yin, Q., Wang, Y., and Chan, Y. (2024). Harnessing earnings reports for stock predictions: A qlora-enhanced llm approach.

Pires, R., Abonizio, H., Almeida, T. S., and Nogueira, R. (2023). *Sabiá: Portuguese Large Language Models*, page 226–240. Springer Nature Switzerland.

Rabe, M. N. and Staats, C. (2021). Self-attention does not need $o(n2)$ memory. *arXiv preprint arXiv:2112.05682*.

Roberts, A., Raffel, C., and Shazeer, N. (2020). How much knowledge can you pack into the parameters of a language model?

Schulman, J., Zoph, B., Kim, C., Hilton, J., Menick, J., Weng, J., Uribe, J. F. C., Fedus, L., Metz, L., Pokorny, M., et al. (2022). Chatgpt: Optimizing language models for dialogue. *OpenAI blog*, 2:4.

Shazeer, N. (2020). Glu variants improve transformer.

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. (2024). Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. (2023). Llama: Open and efficient foundation language models.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2023). Attention is all you need.

Yang, H., Liu, X.-Y., and Wang, C. D. (2023). Fingpt: Open-source financial large language models. *arXiv preprint arXiv:2306.06031*.

Yunxiang, L., Zihan, L., Kai, Z., Ruilong, D., and You, Z. (2023). Chatdoctor: A medical chat model fine-tuned on llama model using medical domain knowledge. *arXiv preprint arXiv:2303.14070*.

Zhang, H., Chen, J., Jiang, F., Yu, F., Chen, Z., Li, J., Chen, G., Wu, X., Zhang, Z., Xiao, Q., et al. (2023a). Huatuogpt, towards taming language model to be a doctor. *arXiv preprint arXiv:2305.15075*.

Zhang, X., Rajabi, N., Duh, K., and Koehn, P. (2023b). Machine translation with large language models: Prompting, few-shot learning, and fine-tuning with QLoRA. In Koehn, P., Haddow, B., Kocmi, T., and Monz, C., editors, *Proceedings of the Eighth Conference on Machine Translation*, pages 468–481, Singapore. Association for Computational Linguistics.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J.-Y., and Wen, J.-R. (2023). A survey of large language models.