Including a contemporary NLP application within an introductory course: an example with student feedback from a University of Applied Sciences

Saurabh Kumar
Artificial Intelligence Technologies
FORVIA Clean Mobility
Augsburg, Germany
saurabh.kumar@forvia.com

Alessandra Zarcone
Technische Hochschule Augsburg
Fakultät für Informatik
Augsburg, Germany
alessandra.zarcone@hs-augsburg.de

Abstract

The recent extensive media coverage of Large Language Models and their applications like ChatGPT have created an unprecedented awareness and curiosity related to Natural Language Processing (NLP) amongst university students from different fields. However, students must understand and master a number of theoretical topics before they can understand how such models work and how they are applied to real-life examples. Within an introductory NLP course at a University of Applied Sciences, we asked ourselves how to best include teaching material related to contemporary applications in order to encourage students to experiment on their own, not only within the course but also later in their studies or in an industrial setting. What could be the major components and how could it be made accessible for students from different programs? What would be the added value compared to a plethora of existing online videos and tutorials? We present our experience with a step-by-step session on a contemporary applied topic, namely Semantic Textual Similarity. We share the examples, the visualization, the slides and the code samples used in the session. We discuss the students' feedback as well as further possibilities for similar future sessions.

1 Introduction

The extensive debates in the media and the publicity received by Large Language Models (LLMs) like GPT4 (OpenAI, 2023) and LaMDA (Thoppilan et al., 2022) that power dialogue-based applications like ChatGPT and Bard respectively has led to heightened curiosity about Natural Language Processing amongst university students. This is particularly the case for students at University of Applied Sciences (*Hochschulen* in the German system), who focus more on use-case-oriented applied research and are fascinated by the industrial applications of NLP and in particular of LLMs, as well as of dialog or assistance systems.

This focus on industrial applications makes it worthwhile to provide students at Universities of Applied Sciences tools to better understand these technologies, their capabilities and their shortcomings as well as opportunities to come up with their own use cases. However, this could constitute a challenge in introductory courses, as students must first understand a number of basic concepts and have the possibility to work with some realistic data, before being able to explore the possibilities and limitations of the most recent approaches. It is thus not trivial to choose a contemporary topic that can be used to explain basic theoretical concepts while being relevant for many practical applications at the same time. Additionally, there are constraints related to required computational power and ease of creating own data.

We chose the topic of Semantic Textual Similarity (STS, Agirre et al., 2012) to introduce the challenges of sentence embeddings as well as their relevance for real-world use cases. We present our experience, provide the materials. We include the feedback provided by the students and discuss the possibilities to further improve selection of topics, and measure student pro-activeness in using the provided code for their own experiments.

2 The Introductory NLP Course

The course, held at the Technische Hochschule Augsburg, consists of 12 units. Each unit consists of four consecutive 45 minute slots, with a short break after the first two. The teaching slots are organized with an alternation of frontal teaching with slides and practical exercises using jupyter notebooks and student presentations. During the current semester (summer semester 2023), 44 students from the Bachelor programs *Informatik*, *Wirtschaftsinformatik* and *Interaktive Medien* and from the Master programs *Informatik, Business Information Systems* and *Applied Research* enrolled in the course, choosing it as one of their electives.

Proceedings of the 1st Workshop on Teaching for NLP, pages 1–7

The Master students as well as the students of *Interaktive Medien* are required to integrate their written exam with either a presentation during class, or a report on relevant papers, or a report on a small, practical project.

The topics of the first 10 units roughly correspond to the content and materials in Chapters 2 to 11 of Jurafsky and Martin (2023), while the last two units are dedicated to Chatbots and Dialogue Systems (Chapter 15).

3 Choosing a contemporary application

Our aim was to introduce a contemporary application in the middle of the course, with the goal of making the significance of some basic, already-introduced theoretical concepts evident and of increasing the students' curiosity about the topics which would be introduced later in the course. We wanted to introduce a topic where the students themselves could explore the effectiveness and shortcomings of simple methods and follow it up with exploration of newer methods and under what conditions they would be useful. We thus picked the topic of Semantic Textual Similarity (STS), with a focus on popular semantic information retrieval systems that combine sentence embeddings (Reimers and Gurevych, 2019) and vector indices (Johnson et al., 2021) with existing methods like BM25 (Robertson and Zaragoza, 2009). The unit on Semantic Textual Similarity was presented during Unit 6, that is after the introducing language modeling (n-grams), text classification (Naïve Bayes and logistic regression) and vector semantics (word embeddings) in the previous units. During the same unit, feedforward networks were introduced as models for language modeling and textual classification. We introduced the limitation of their input to a fixed-length word window and the question of how word-level embeddings can be best combined to represent word sequences (for example, in order to represent a document to be categorized). Our goal was to make the students aware of the fact that they could think of extending the code samples provided in the session to achieve information retrieval quality similar to some of the state-of-the-art systems. The code and slides are available in the GitHub repository at: `https://github.com/saurabhkumar/lecture1_semantic_similarity`.

4 Datasets, result visualization and computational needs

It is common practice to request students to download standard datasets from the internet and work with it in the courses. This has the advantage that datasets can be reused, and the course material can be standardized. However, students from Universities of Applied Sciences may be more interested in use-case specific datasets than in standard benchmarks. Furthermore, it is also challenging in an introductory course to make students understand why algorithm performance on a benchmark dataset does not always translate to good performance on their own real-word examples and what characteristics of their example data are not covered in the benchmark dataset.

To overcome this limitation, we created a small set of examples at increasing levels of complexity, based on the contrast of different general-domain concepts such as countries, capitals, language, economics, demographics, and cuisine. The data can be obtained from Wikipedia and the links have been provided in the jupyter notebook for the session. This helped us to set out our goals for the topic and explain the performance of algorithms as the complexity of real-world sentences increases. Our main goal was to enable the students to easily expand the datasets themselves and see the change in performance. This is rarely seen in datasets like the STS benchmark (Cer et al., 2017). We believe this is critical to help students reflect on the effect of different data, learning the value of understanding the characteristics of their datasets and appreciate their impact on the performance of algorithms they use to achieve task specific goals. This approach also allowed us to increase the students' curiosity and encourage them to try things out with far more complex use cases, for example the possibility of automatically adding nodes based on these concepts to a knowledge graph.

We started with a simple set of sentences (***Sentence Set 1***) and demonstrating the cosine similarities between sentence vectors obtained by just averaging the individual normalized word vectors.

S1: *Paris is the capital of France*

S2: *Berlin is the capital of Germany*

S3: *French is a Romance language of the Indo-European family*

S4: *German is an Indo-European language which*

belongs to the West Germanic group of Germanic languages

The last two sentences are taken from Wikipedia[1].

It is important to be able to visualize the results during the experimentation. We tried to provide code to the students to be able to easily visualize the results for the examples. We found that using the visualization also made it easy to demonstrate the progressive improvements in the achieved results to the students as the methods were changed.

Figure 1 showed the students that the result is not very impressive.

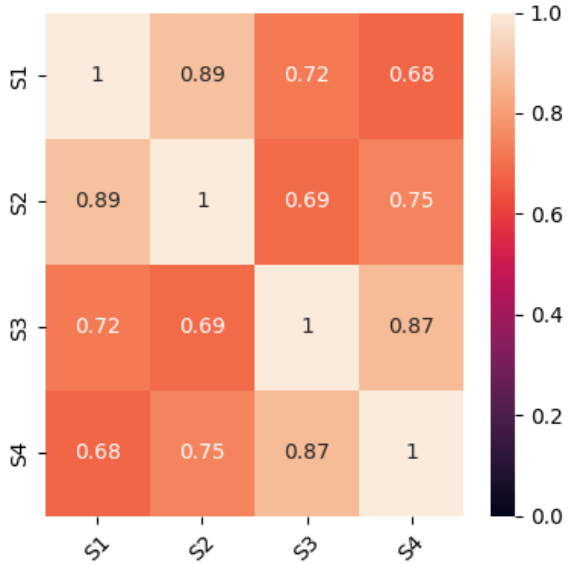

Figure 1: Cosine similarities for ***Sentence Set 1***.

We then suggested a trivial method, that is just using the 'important' words. We thus modified the sentences (***Sentence Set 2***) and demonstrated the improvement brought by this method, as shown in Figure 2.

S1: *Paris capital France*

S2: *Berlin capital Germany*

S3: *French language*

S4: *German language*

Between one step and the other, we activated the students by encouraging them to brainstorm and suggest what a possible next step could be, and we could notice that students were impressed at seeing

[1] https://en.wikipedia.org/wiki/French_language, https://en.wikipedia.org/wiki/German_language

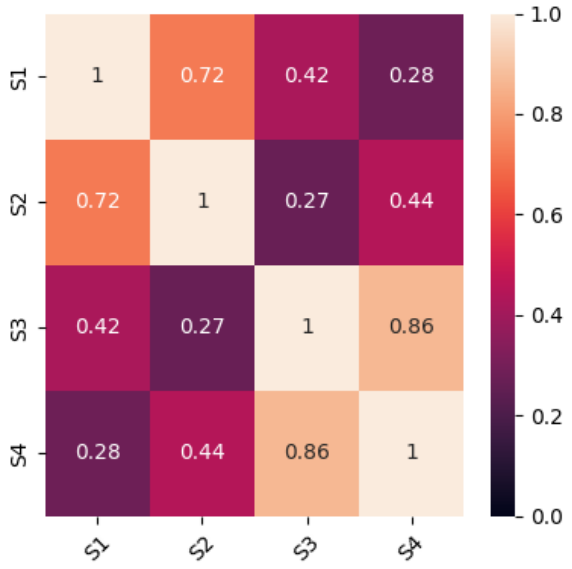

Figure 2: Cosine similarities for ***Sentence Set 2***.

the results of what they had already learnt in the previous units of the course. This also helped in pointing to additional reading material mentioned in the code to understand what could be done to find 'important words' in this context. The heat maps with overlayed similarity scores made it easy to showcase the improved performance.

Then we tried to explain the challenges posed by real-word data by using more complex and longer sentences (***Sentence Set 3***) taken again from the Wikipedia pages related to similar topics[2].

S1: *France has a developed high-income mixed economy characterised by sizeable government involvement economic diversity a skilled labour force and high innovation. For roughly two centuries the French economy has consistently ranked among the ten largest globally.*

S2: *Germany is a federal, parliamentary, representative democratic republic. Federal legislative power is vested in the parliament consisting of the Bundestag (Federal Diet) and Bundesrat (Federal Council), which together form the legislative body.*

S3: *With a population of 80.2 million according to the 2011 German Census, rising to 83.7 million as of 2022, Germany is the most populous*

[2] https://en.wikipedia.org/wiki/France, https://en.wikipedia.org/wiki/Germany. We intentionally chose and mentioned these sources to enable students to later pick their own 'real world' sentences instead of trivialized examples.

country in the European Union, the second-most populous country in Europe after Russia, and the nineteenth-most populous country in the world.

S4: *Each region of France has traditional specialties: cassoulet in the Southwest, choucroute in Alsace, quiche in the Lorraine region, beef bourguignon in Burgundy, provençal tapenade, etc.*

We trivially removed a set of stop words from these sentences (as in the script provided) and showed the results of our trivial method to generate sentence level embeddings and calculate cosine similarity. The students could see that the results as shown in Figure 3 were not what they expected, and they could experiment by replacing the sentences in the code.

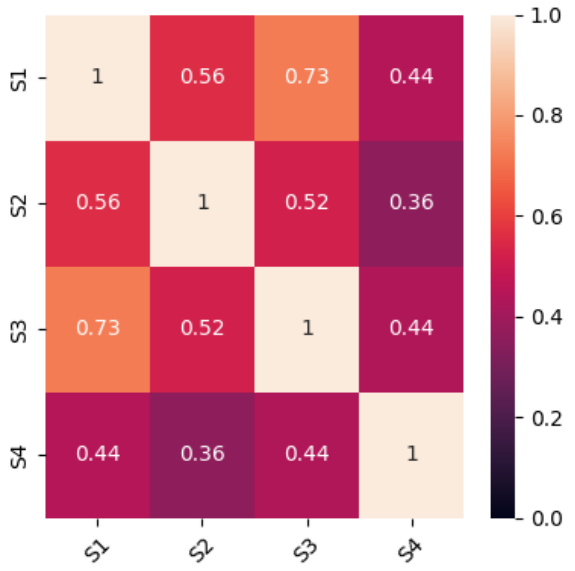

Figure 3: Cosine similarities for ***Sentence Set 3***.

In this way we were able to set up the stage for methods that are built on top of models like BERT (Devlin et al., 2019), to pique the students' interest for the next topics in the course and to show why those methods have high practical significance for many NLP applications. We did not go into the details of the methods, as they will be introduced in later units. Instead, we continued with increasing the complexity of our example sentences and demonstrating the effectiveness of the methods. We expected to create appreciation for the need for methods like attention (Bahdanau et al., 2014) and transformers (Vaswani et al., 2017) that would follow later in the course while still focusing on our chosen topic about sentence embeddings and Semantic Textual Similarity.

We modified the sentence set to have higher diversity in topics and create more complexity for the task by having sentences of very different lengths (***Sentence Set 4***). We believe it is important for students to understand factors such as length and complexity when dealing with real-world data.

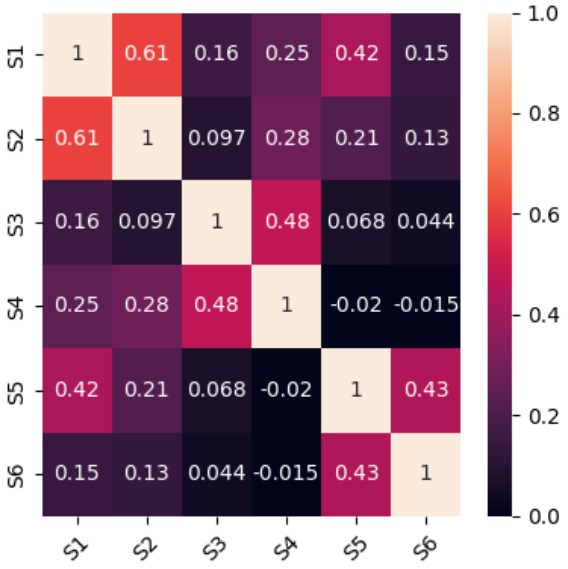

Figure 4: Cosine similarities for ***Sentence Set 4***.

S1: *France has a developed high-income mixed economy characterised by sizeable government involvement economic diversity a skilled labour force and high innovation. For roughly two centuries the French economy has consistently ranked among the ten largest globally.*

S2: *French economy is the world's seventh-largest economy by nominal GDP*

S3: *Germany is a federal, parliamentary, representative democratic republic. Federal legislative power is vested in the parliament consisting of the Bundestag (Federal Diet) and Bundesrat (Federal Council), which together form the legislative body.*

S4: *With a population of 80.2 million according to the 2011 German Census, rising to 83.7 million as of 2022, Germany is the most populous country in the European Union, the second-most populous country in Europe after Russia, and the nineteenth-most populous country in the world.*

S5: *Each region of France has traditional specialties: cassoulet in the Southwest, choucroute*

in Alsace, quiche in the Lorraine region, beef bourguignon in Burgundy, provençal tapenade, etc.

S6: *A typical French Christmas dish is turkey with chestnuts.*

Another aspect that cannot be neglected in an introductory course with participants from varied disciplines is the computational requirements. The code explicitly mentions the computational needs and how students could use different models based on the computational resources available to them. We used the Sentence Transformers library (`https://www.sbert.net/`) that is based on the methods presented in Reimers and Gurevych (2019) and selected a model that could be easily used on a standard laptop by all students. We additionally mentioned the other models that could be experimented with when more RAM and computational power was available to the students. The result in Figure 4 showed the students how such models could leverage attention mechanism to generate better sentence embeddings.

We ended our demonstration by providing the students with a method to generate data for model finetuning for a task, in order to further separate concepts like economy and cuisine and hinted at possible experiments, for example changing the finetuning dataset source and size and see the effects. The default model we selected in the sample code is important here because students would not be able to run the finetuning on standard laptops for larger models. Completing this task would prepare students for complex real-world applications like domain adaptation of the models for use in semantic information retrieval systems.

At the end we suggested the students to think about collecting the sentences in this form and automatically trying to add them to a graph with a hierarchy of conceptual nodes. To increase their curiosity and willingness to experiment, we provided a hint that this task could provide them a simple basis for more complex representations like Knowledge Graphs, as used for example by Google.

5 Instruction Language

We experimented with a combined usage of German and English. Since the course is taught in German, the slides were created in German and the teaching was also done in German. However, the code comments and additional reading mentioned in the code was in English. The need and intention behind this are twofold. First, most additional reading material related to the topic is available only in English and the documentation for the used libraries is also available only in English. Second, this opens the possibility for students to experiment themselves with creating sample data in German and experiment with multilingual models. Since the text examples are from Wikipedia, getting the data in German and extending it is relatively simple. Interestingly, in the anonymized feedback collected later, the majority of students mentioned that even though they appreciated that the slides and teaching was done in German, they did not consider it necessary. No respondent gave the feedback that having the code comments and additional reading material in English was of any concern to them.

6 Student feedback and possible takeaways

We asked the students to provide anonymized feedback for a set of questions/statements. Eleven students provided the feedback about the session. Figure 5 shows the responses to a subset of questions (translated to English) where students had to select one answer from the four available choices.

Twenty students participated in a more general evaluation of the whole introductory course and were also free to leave comments about the course. Three of them explicitly mentioned the guest lecture on semantic textual similarity as a positive aspect or expressed the wish to see more presentations from the applied domain. Even though the sample size for the results is not large and it is based on a single topic and session, there are some important takeaways for us from this. As most students managed to understand the topic and experimented with the code, we believe that it is feasible to introduce such topics in an introductory course. They also found the availability of code useful. The feedback that most students found the topic relevant for usage in their later careers and that the session increased their interest in the domain, points to the benefits of such an approach.

6.1 Further possibilities and challenges

We realized that one of the shortcomings of our current approach was that we could not evaluate how many students put in the effort to modify or extend the datasets or use different models to conduct their own experiments. We would like to explore how

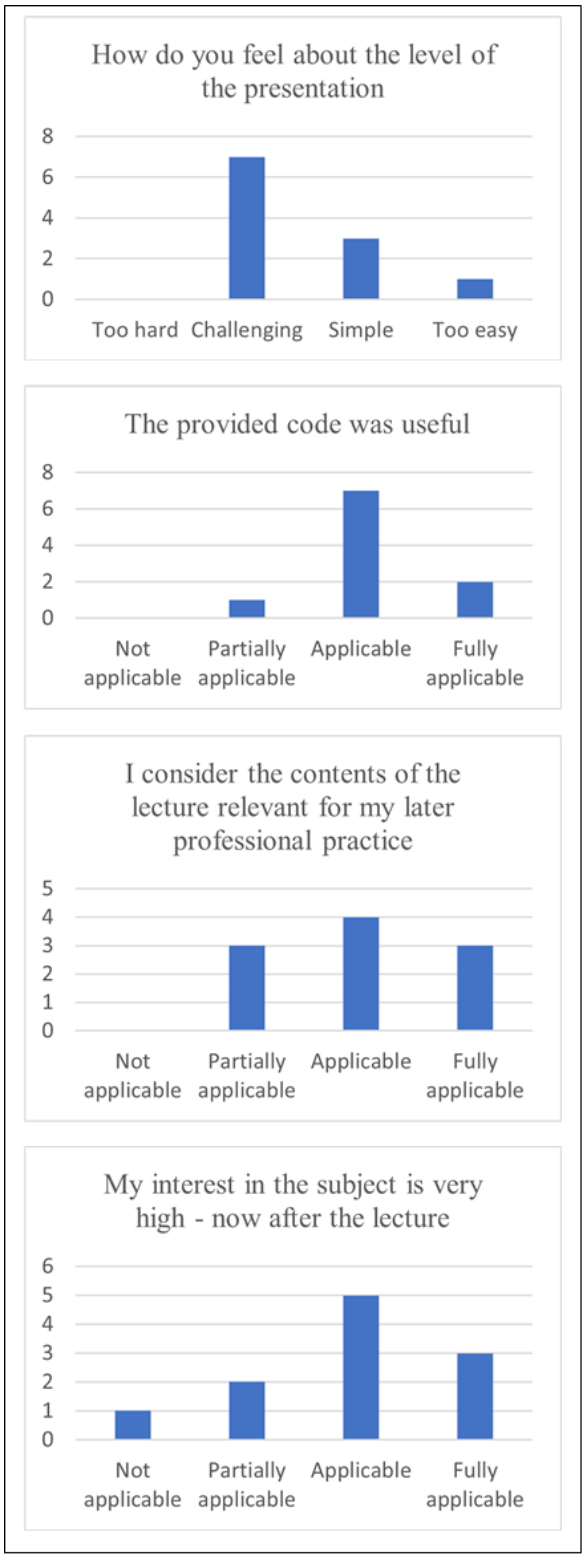

Figure 5: Feedback from the students about the session

we could do this in an introductory course – possibly using an e-learning platform such as Moodle[3] – and how could the students be rewarded for their effort. We believe that the selection of topics is important and not every contemporary topic can be introduced with equal ease in an introductory course.

We also attempted another such session on Instruction Finetuning for LLMs, with a similar goal to offer a demo within the existing constraints on computational power and data. The code and slides for the second unit are available in the GitHub repository at: https://github.com/saurabhkumar/instruction_tuned_llm. Additional complexities arise from the computational requirements of the most recent technologies – at least while the Technische Hochschule is in the process of acquiring GPU servers. Until then, we are faced with the challenge of introducing computationally-intensive topics while enabling all students to be able to use the code.

[3] https://moodle.org/

Acknowledgements

We would like to thank the University of Applied Sciences (Technische Hochschule) in Augsburg for providing us this opportunity. We would also like to thank Mr. Klaus Spindler at FORVIA Clean Mobility for the encouragement to conduct the session and in general engagement with students at the University of Applied Sciences (Technische Hochschule) in Augsburg. We are also extremely thankful to all the students who provided their valuable feedback.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, Montréal, Canada. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

J. Johnson, M. Douze, and H. Jegou. 2021. Billion-Scale Similarity Search with GPUs. *IEEE Transactions on Big Data*, 7(03):535–547.

Dan Jurafsky and James H Martin. 2023. *Speech and Language Processing*. Jan 7, 2023 draft, accessed here https://web.stanford.edu/ jurafsky/slp3/.

OpenAI. 2023. Gpt-4 technical report.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, YaGuang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Vincent Zhao, Yanqi Zhou, Chung-Ching Chang, Igor Krivokon, Will Rusch, Marc Pickett, Pranesh Srinivasan, Laichee Man, Kathleen Meier-Hellstern, Meredith Ringel Morris, Tulsee Doshi, Renelito Delos Santos, Toju Duke, Johnny Soraker, Ben Zevenbergen, Vinodkumar Prabhakaran, Mark Diaz, Ben Hutchinson, Kristen Olson, Alejandra Molina, Erin Hoffman-John, Josh Lee, Lora Aroyo, Ravi Rajakumar, Alena Butryna, Matthew Lamm, Viktoriya Kuzmina, Joe Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguera-Arcas, Claire Cui, Marian Croak, Ed Chi, and Quoc Le. 2022. Lamda: Language models for dialog applications.

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*, 30.