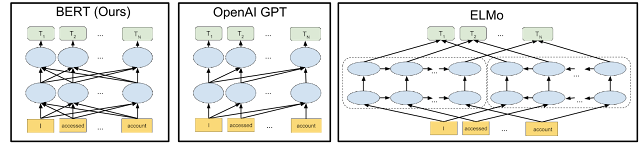
# BERT

* Published 2018 by google AI, led to a small revolution
* Huge neural network based on transformer architecture and pretrained on 3.3 Billion words
* Builds upon previous approaches on pretraining contextual representations: ELMO, GPT; Semi-supervised sequence learning, ULMFit

### Pretrained contextual representations

* + One of the biggest challenges in NLP is shortage of training data. Most task-specific datasets contain only a few hundred or thousand human-labeled training examples
  + NN brauchen aber teilweise riesige Datenmenge, um etwas gutes zu lernen
  + Um diese Lücke zu schließen, haben researcher eine Reihe an Approaches entwickelt, um general purpose language representation models mithilfe der riesigen Menge an unannotierten Text im Web, vorzutrainieren, known as pre-training.
  + Diese vortrainierten Models können dann fine-tuned werden auf small-data NLP tasks like question answering or sentiment analysis, resulting in substantial accuracy improvements compared to training these datasets from scratch
  + Pre-trained representations können entweder context-free oder contextual sein. Contextual können wiederum unidirectional or bidirectional sein.
  + Context-free: Word2Vec or GloVe erzeugen word embeddings for each word in the vocab. “bank” (Sparkasse) und “bank” (Sitzbank) hätten hier dieselbe representation, unabhängig von ihrem Kontext
  + Contextual Models: generate a representation of each word based on the context of the text/ based on the other words in the sentence
  + Unidirectional Model: “I accessed the bank account” repräsentiert bank anhand des Kontextes “I accessed the”, aber NICHT “account”
  + Bidirectional Model: BERT; represents “bank” using both, its previous and next context – making it deeply bidirectional



* BERT ist deeply bidirectional, OpenAI GPT is unidirectional and ELMO is shallowly bidirectional (Google Blogpost2) 🡪 auch in offiziellem BERT Paper

Pretraining Objectives:

* + If bidirectionality is so powerful, why hasn’t it been done before?
  + Unidirectional models are trained by predicting each word conditioned on the previous words in a sentence (classic language modelling). Ein bidirectionales Model kann allerdings nicht auf diese Art und weise trainiert werden, since it would allow the word that’s being predicted to indirectly see itself in a multi-layer model. Deshalb werden folgende Objectives verwendet

MLM

* + Helps it learn the context in a sentence
  + Maskieren von 15% der Tokens einer Input sequenz and then condition each word bidirectionally to predict the masked words.
  + Example: “The man went to the [MASK]1 . He bougt a [MASK]2 of milk. 🡪 MASK1 = store, MASK2 = gallon
  + In der LIteratur schon länger bekannt unter dem namen “Cloze task” (cloze paper)
  + Sfasd

NSP

* + Helps it learn the relationship between two sentences, nachdem das nicht directly von dem MLM gecaptured wird.
  + Für Satz A und B muss predicted werden, ob B der tatsächlich nächste Satz von A ist (50% ja, 50% einfach ein random Satz aus dem Korpus)
* Ermöglicht Transfer-Learning

### Transfer Learning:

* + The technique of transferring knowledge gained from performing one task to another, similar one
  + Beneficial compared to the resource intensive process of training networks from scratch

### Architecture

* Based on transformer architecture
* BERT decomposes input sentence(s) into WordPiece tokens (Wu et al., 2016) 🡪 wordpiece tokenization helps improve the representation of the input vocab and reduce its size by segmenting complex words into subwords. It therefore also tackles the OOV Problem by segmenting unknown vocabs in smaller units.
* Fundamentally, BERT is a stack of Transformer encoder layers (Vaswani et al., 2017) that consist of multiple self-attention ‘‘heads’’. For every input token in a sequence, each head computes key, value, and query vectors, used to create a weighted representation. The outputs of all heads in the same layer are combined and run through a fully connected layer. Each layer is wrapped with a skip connection and followed by layer normalization.
* The conventional workflow for BERT consists of two stages: pre-training and fine-tuning. Pretraining uses two self-supervised tasks: masked languagemodeling(MLM, prediction ofrandomly masked input tokens) and next sentence prediction (NSP, predicting if two input sentences are adjacent to each other). In fine-tuning for downstream applications, one or more fully connected layers are typically added on top of the final encoder layer.
* The input embedding layers (token, position, and segment) are combined to obtain a fixed-length vector. Special token [CLS] is used for classification predictions, and [SEP] separates input segments.
* Google1 and HuggingFace (Wolf et al., 2020) provide many variants of BERT, including the original ‘‘base’’ and ‘‘large’’ versions. They vary in the number of heads, layers, and hidden state size.
* Von A Primer BERTology

Finetuning

Fine-tuning is straightforward since the selfattention mechanism in the Transformer allows BERT to model many downstream tasks— whether they involve single text or text pairs—by swapping out the appropriate inputs and outputs. For applications involving text pairs, a common pattern is to independently encode text pairs before applying bidirectional cross attention, such as Parikh et al. (2016); Seo et al. (2017). BERT instead uses the self-attention mechanism to unify these two stages, as encoding a concatenated text pair with self-attention effectively includes bidirectional cross attention between two sentences.

For each task, we simply plug in the taskspecific inputs and outputs into BERT and finetune all the parameters end-to-end. At the input, sentence A and sentence B from pre-training are analogous to (1) sentence pairs in paraphrasing, (2) hypothesis-premise pairs in entailment, (3) question-passage pairs in question answering, and

(4) a degenerate text-∅ pair in text classification or sequence tagging. At the output, the token representations are fed into an output layer for tokenlevel tasks, such as sequence tagging or question answering, and the [CLS] representation is fed into an output layer for classification, such as entailment or sentiment analysis.

Compared to pre-training, fine-tuning is relatively inexpensive. All of the results in the paper can be replicated in at most 1 hour on a single Cloud TPU, or a few hours on a GPU, starting from the exact same pre-trained model.7 We describe the task-specific details in the corresponding subsections of Section 4. More details can be found in Appendix A.5.

* Due to open sourcing BERT, many researchers could reconstruct the results and finetune BERT for their own downstream tasks (machine translation, ….)

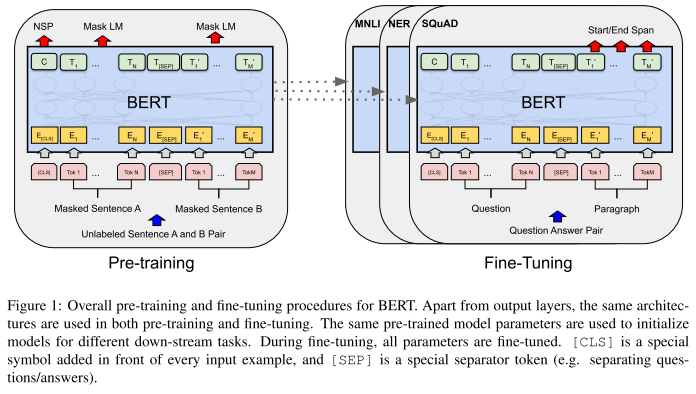


Figure : aus offiziellem BERT Paper

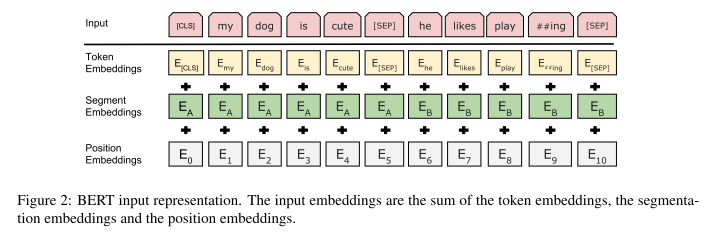


Figure : aus offiziellem BERT paper

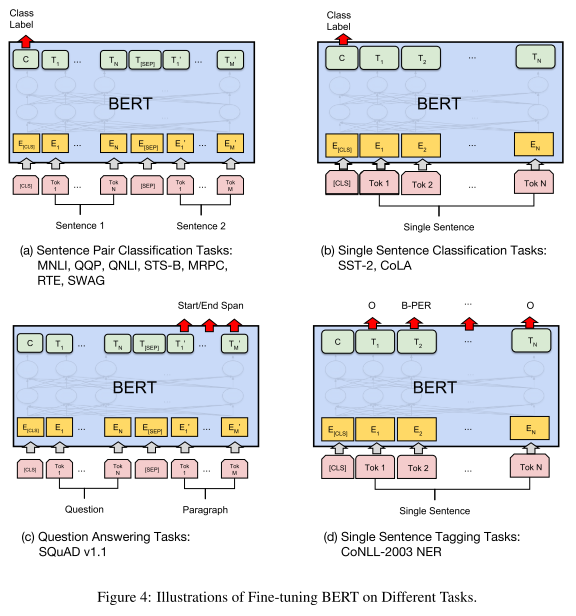


Figure : aus offiziellem BERT paper

# BERT and conversational agents

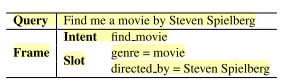
* Aufgrund der promising results of BERT for many downstream tasks, wurde es dann auch für verschiedene Aufgaben von conversational agents eingesetzt
* NLU tasks often suffer from small-scale human-labeled training data, resulting in poor generalization capability, especially for rare words. 🡪 BERT behebt das

### BERT-DST

* Dialogue state tracking: consists of determining at each turn of a dialogue the full representation of what the user wants at that point in the dialogue, which contains a goal constraint, a set of requested slots, and the user's dialogue act. The dialogue states predicted by DST are used by the downstream dialogue management component to produce API calls to a backend database and generate responses to the user. A dialogue state is often expressed as a collection of slot-value pairs (dafür ist eine ontology nötig)
* BERT-DST is an end-to-end dialogue state tracker which directly extracts slot values from the dialogue context. BERT is used as a dialogue context encoder whose contextualized language representations are suitable for scalable DST to identify slot values from their semantic contet.
* In diesem paper wird DST mit einer unknown ontology and unseen slot values behandelt
* Not requiring candidate value generation, BERTDST directly predicts slot values from the dialogue context

### BERT for joint intent classification and slot filling

* Inent class. Focuses on predicting the intent of the query, and slot filling extracts semantic concepts (slot filling = sequence labelling task, tags the input words)



* They show that BERT achieves significant improvement on intent classification accuracy, slot filling F1, and sentence-level semantic frame accuracy on several public benchmark datasets, compared to attention-based RNN models and slot-gated models.

### Fine-grained post-training for improving retrieval-based dialogue systems

* They use a fine-grained post-training method to perform better in the response selection task on three benchmark datasets (they also use a new training objective for that)
* “During the multi-turn response selection, BERT focuses on training the relationship between the context with multiple utterances and the response. However, this method of training is insufficient when considering the relations between each utterance in the context. This leads to a problem of not completely understanding the context flow that is required to select a response.”
* New Training objectives:
* Train the model by dividing the entire dialogue into multiple short context-response pairs
* Train the model with the new objective of utterance relevance classification (URC), which classifies the relation between given utterances and the target utterance into more fine-grained labels. Ähnlich zum NSP task, nurd ass hier das model unterscheidet, ob die target utterance is random or the next.

### Task-specific objectives of pre-trained language models for dialogue adaptiaton

* Pre-training auf task-indepent data enables the model zwar to learn (to some extent) universal language representations, but fails to capture crucial task-specific features
* Focus der Arbeit liegt auf Dialogue-related Natural Language Processing (DrNLP) tasks und auf dem Erstellen eines Dialogue-Adaptive Pre-training objectives (DAPO)
* Das DAPO model ist dann finetuned auf dialogue based question answering, response selection and dialogue quality evaluation 🡪 Outperformed auf allen Tasks
* Sie verwenden allerdings ELECTRA als ausgangsmodell!

### BERT with history answer embedding for conversational question answering

* One of the major challenges in multi-turn conversational search is to model the conversational history to answer the current question. Existing methods either prepend history turns to the current question or use complicated attention mechanisms to model the history. This paper proposes history answer embedding: it enables seamless itegration of conversational history into a conversational question answering model built on BERT
* Why Conversational QA? ConvQA is a simplified setting of conversational search since ConvQA systems do not focus on asking proactively, but are concrete enough for IR researchers to work on modelling the change of information needs between cycles.
* ConvQA is closely related to machine comprehension (SQUAD), nurd ass questions are organized in conversations.

### Dialogue Breakdown detection

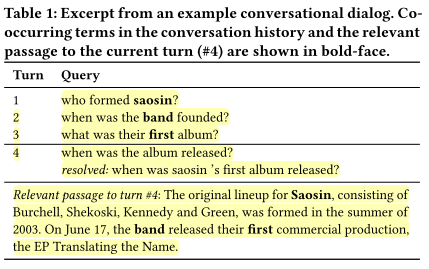
* BERT für dialogue breakdown detection
* The difficulty of developing chat-oriented dialogue systems is that such systems are required to respond to a very wide range of topics expressed by user utterances. Since it is still difficult for the current dialogue systems to continue outputting appropriate responses, utterances that cause the dialogue to collapse are often generated. It is assumed that the continuation of dialogue becomes easy when we can detect and suppress such problematic utterances

### Question rewriting for conversational question answering

* ConvQA requires the ability to correctly interpret a question in the context of previous conv turns
* Question rewriting: is specifically designed to reformulate ambiguous questions, which depend on the conversational context, into unambiguous questions that can be correctly interpreted outside of the conversational context.
* We introduce a conversational QA architecture that sets the new state of the art on the TREC CAsT 2019 passage retrieval dataset. Moreover, we show that the same QR model improves QA performance on the QuAC dataset with respect to answer span extraction, which is the next step in QA after passage retrieval. Our evaluation results indicate that the QR model we proposed achieves near human-level performance on both datasets and the gap in performance on the end-to-end conversational QA task is attributed mostly to the errors in QA
* Architektur:

Vorheriger conversational context + aktuelle implicit question = Input 🡪 Query Rewriting (GPT) 🡪 Passage Retrieval: produce ranked list of text passages from a collection, ordered by their relevance to a given natural language question. Besteht aus candidate selection (traditional retrieval algorithm BM25) und passage re-ranking (BERT). 🡪 extractive QA: given a question and a passage find the answer as a contiguous text span within the given passage (BERT)

### Query resolution for conversational search



* Query rewriting 🡪 in diesem fall: add missing context from the conversational history to the current turn query, if needed; bsp siehe Bild
* Query resolution wird in diesem Paper als binary term classification task formuliert/ vorgeschlagen (QuReTeC = Query Resolution by Term Classification): für jeden term in the previous turns of the conversation wird entschieden, ob er zu der aktuellen Query hinzugefügt werden soll oder nicht
* Das vorgeschlagene QuReTec Model basiert auf BERT: The model encodes the conversational history and the current turn query and uses a term classification layer to predict a binary label for each term in the conversational history
* When QURETEC is integrated in multi-stage ranking architecture, it significantly outperforms baseline models of different nature and is robust across conversation turns.
* Also, we found that our distant supervision method can substantially reduce the required amount of gold standard query resolutions required for training QuReTeC, using only query-passage relevance labels. This result is especially important in low resource scenarios, where gold standard query resolutions might not be readily available.

### A short survey of pretrained language models for conversational AI

* Most datasets available for NLP tasks are rather small. This data scarcity problem makes it difficult to train the deep neural networks, as they would result in an over-fitted model and not generalize well on these small datasets. The concept of pre-training a model on ImageNet corpus has been employed for quite a few years in the field of computer vision [7, 10]. The idea is to make the model learn the general features of an image and this learning can then be utilized in any vision task such as image captioning etc. to achieve the state-of-the-art results.
* The first BERT based model for QuAC was based on history answer embeddings to provide extra information to input tokens [12]. Later, [9] improved accuracy by introducing the last two contexts when answering the current question. [19] introduced the reasoning process in BERT-based architecture that improved the accuracy on the leader board drastically as compared to the previous models.
* The top positions on CoQA leaderboard 3 are occupied by pre-trained language models.
* A traditional task-oriented
* model consists of four modules namely: i) natural language understanding, ii) dialogue state tracking (DST), iii) policy learning, and iv) natural language generation.
* Lane [3] recently utilize the strengths of BERT in improving the scalability of DST module. The DST module is use to maintain the state of user’s intentions through out the dialogue. The key component ofthe model is BERT dialogue context encoding module which generates contextualized representations of the words which are very effective for mining slot values from the contextual patterns.
* In recent years, the promising notion of pre-trained language models has gained widespread attention by researchers. It is an emerging paradigm aimed to generate better contextualized representations of the words so that the dialogue systems have a better understanding of the context

### Towards topic guided conversational recommender systems (CRS)

* Conversational recommender system aims to provide high quality recommendations through conversations with users. Early CRS mainly asked questions about user preferences over predefined slots to make recommendations. Neuere Ansätze interagieren mit dem Nutzer durch natural language conversations, emphasizing fluent response generation and precise recommendation
* They use BERT for the recommendation Module: 🡪 we utilize the pretrained language model BERT to encode the historical utterancest, (+ SASRec to encode user interaction sequence). Aus diese zwei Repräsentationen kann dann die probability berechnet werden, mit der ein Item vom Item set dem user recommendet wird
* They also use BERT for the dialogue module: it predicts the next topic that guides the user tot he target topic.

Also zusammengefasst setzen viele Researchers BERT schon für conversational agents für die unterschiedilchsten Aufgaben ein. Dabei haben sie immer das standard BERT modell eingesetzt (manchmal auch angepassen BERT, der unterschieldiche pretrianing objectives hatte, bsp. History embeddings, …) was natürlich auch meistens Sinn macht, da viele der verwendeten Datensätze (wie Squad oder CoQA) auch der generellen Domäne angehört hatten. Allerdings gibt es in der Literatur sehr viele Arbeiten, die beweisen, dass BERT domänenspezifisches Wissen mangelt und daher downstream tasks gewisser Domänen limitiert sind. Das wird im nächsten Kapitel bearbeitet.

Struktur

* Conversational Agents
  + Welche Arten von Conv agents gibt es 🡪 Chatbot, Dialogue system , …
  + Conversational Agents entwicklung 🡪 von handcrafted zu neuronalen Netzen
  + Conversational Agents für welche Kontexte: Health, elderly care, …, sogar Cooking
  + Conv. Agents für die Küche:
    - Siehe <https://link.springer.com/chapter/10.1007/978-3-642-40942-4_19> fpr Mobile speech cooking assistant
    - <http://dspace.library.uvic.ca/handle/1828/9583> Caprecipes: food recommendation🡪 proof of concept für food recommender system
    - Wieso könnte das Relevant sein?
    - Es gibt mehrere Paper, die annehmen, dass das sinnvoll ist (Paper von Elsweiler usw.)
    - Frummets Paper: untersucht information needs, die beim Kochen auftreten
    - Sabrina hat Arbeit gemacht bzgl. Recommender systems
  + Conv. Agents funktionsweise/ welche NLP tasks werden benötigt/ gelöst 🡪 um dann sagen zu können, dass meine drei test-tasks relevant für conversational agents sind
    - Hier vielleicht eher allgemein sagen, dass es keine einheitliche Pipeline gibt, sondern es ganz davon abhängt, wie das System eingesetzt wird (generell gibt’s eine Komponente für Sprach-recognition, eine für Natural Language understanding, und eine für text-generation 🡪 CookBERT wäre hier für die NLU komponente sinnvoll)
* BERT Prerequisites
  + Word embeddings
  + Encoder Decoder
  + Attention
  + Transformer
  + Transfer Learning
* BERT
  + Architektur
    - Was ist anders als bei Transfomrern
    - Input darstellung
  + BERT
  + Static vs. Contextualized word embeddings
    - Was macht BERT so besonders im Vergleich zu ähnlichen Ansätzen
  + BERT Architektur
  + Pretraining
    - MLM
    - NSP
  + Finetuning
    - Für unterschiedliche Tasks, z.B. NER, Question Answering Text classification, …
    - Aufgrund der guten Performance hats auch nicht lange gedauert, bis BERT für conversational agents eingesetzt wurde:
    - Ist auch ein Konzept von Transfer Learning
  + BERT und Conversational Agents
    - Hier dann korrespondierend zum Punkt in conversational agents sagen, für welche Tasks BERT beispielsweise eingesetzt wird (und damit auch meine Tasks motivieren)
      * Intent classification
      * Answer span extraction
    - Das Problem allerdings: fehlendes Domänenspezifisches Wissen
  + Fehlendes Domänenwissen/ Domänenadaption
    - SciBERT
    - DAPT
    - Das führt dazu, dass BERT schon für unterschiedlichste Domänen verfügbar ist
  + BERT für unterschiedliche Domänen
    - BERT in der Kochdomäne
    - Hier auch schon die verschiedenen Datensätze mit reinbringen, in denen BERT angewandt wird und gute Performance erreicht (Cookversational search, DoQA, FoodBase)
* Cooking datasets relevant for conversational agents
  + Auch wenn Kochen sehr relevant ist, ist die Anzahl verfügbarer Datensätze überschaubar. Im folgenden werden 5 Datensätze vorgestellt, die in der Arbeit relevant sind
  + Die meisten Datensätze sind zudem sparse, was Transfer Learning zusätzlich motiviert
  + Die fünf Datensätze vorstellen
    - Cookversational search
    - DoQa
    - RecipeNLG
    - Recipe1M
    - FoodBase
* Summary and Key differences
  + Research Question hier reinpacken

# Conversational Agents

### History und Einordnung

* History
* Wie werden Cas eingeordnet/ klassifiziert
* Welche Arten von Cas gibt’s
* In welchen Kontexten werden sie angewandt
* Conversational agents sind mittlerweile ubiquitous.
* Sie kommen in unterschiedlichen Ausführungen und die bezeichung in der Literatur und Media ist sehr inkonsistent. Nichtsdestotrz lassen sich derartige System einteilen: Chatbot, Question answering system, Dialogue System. In dieser Arbeit wird der Begriff Conversational Agent verwendet, und kombiniert damit alle solche systeme, die eine alternative zu traditionellen Methoden für menschen to seek for information bieten, indem sie die suche mehr conversational machen
* Conv. Agents existieren schon länger und haben ihren Anfang mit ELIZA.
* Frühere Ansätze basierten auf handgecrafteten Regeln, und heute haupsächlich neuronale Netze (wie bspw. BERT, was in Section … behandelt wird)
* Während Cas open domain sein können, können sie aber auch in einer spezifischen Domäne eingesetzt werden, beispiele sind …

### Conversational Agents in the kitchen

* Auch wenn die Kochdomäne arguably ein sinnvoller Kontext für Cas ist, gibt es relativ wenig Arbeit in diesem Bereich und keinen bekannten kommerziellen Küchenassistenten. Lediglich open domain Cas wie Google Assitant können für kochspezifische Aufgaben/ Needs verwendet werden, wobei hier auch die Nachfrage relativ groß zu sein scheint. Derartige Systeme können beispielsweise Timer stellen, oder Rezepte durchgehen und teilweise auch rezepte vorschlagen, allerdings alles in einem begrenzten Rahmen.

Folgender Research in der Literatur wurde betrieben:

* <https://www.tandfonline.com/doi/abs/10.1080/10400435.2012.659834> - the design of an interactive assistive kitchen system
* <https://www.mdpi.com/1424-8220/14/1/1629> a smart kitchen for ambient assisted living
* Foodie fooderson
* recipeBot
* Frummet Paper
* Sabrina Paper
* Elsweiler Paper

### Cas Tasks/ Funktionsweise

* Wie funktionieren Cas, welche NLP tasks werden benötigt
* GUS:
* Framework für task-based dialogue
* First introduced in 1977 by Bobrow et al.
* Most if not all modern commercial digital asisstants are based on this
* Slot filling (= sequence labelling task und damit sehr ähnlich zu NER), Domain classification (wenn das dialogue system multi-domain ist, ansonsten nicht nötig) und intent determination (what general task or goal does the user want to accomplish?) sind für GUS Systeme relevant
* Question Answering Systems

### Neural Language Modeling

* Was ist language modeling/ was ist das Ziel davon

Transfomer



* = a new simple network architecture, based solely (einzig und allein) on attention mechanismus. Dadurch sind keine Wiederholungsschleifen wie bei bisherigen Architekturen RNN und LSTM mehr nötig
* Neben Verbesserung bei Machine translation sehr guten Biggest benefit comes from how the transformer lends itself to parallelization and requires significantly boosts the speed with which deep learning models can be trained
* Proposed in paper “Attention is all you need”
* The Transformer follows the overall architecture of the encoder-decoder structure: Es gibt einen encoding component (wiederum bestehend aus einem stack von 6 Encodern) und einen decoding component (wiederum bestehend aus einem stack von 6 Decoder)
* Alle Encoders bestehen aus zwei Layern: Self-Attention layer (a layer that helpts the encoder look at other words in the input sentence as it encodes a specific word) und Feed-Forward Network
* Die Decoder haben zusätzlich noch eine attention layer that helps the decoder focus on relevant parts of the input sentence

Self-Attention

* ”The animal didn't cross the street because it was too tired”
* Self-Attention allows the model to associate the word “it” with “animal”
* As the model processes each word (each position in the input sequence), self attention allows it to look at other positions in the input sequence for clues that can help lead to a better encoding for this word

Detailed

1. Create three vectors from each of the encoders input vectors: Query vector, key and value vector

## Vorheriges Related Work

## BERT

With the publication of Bidirectional Encoder Representations from Transformers (BERT) by the Google AI team (Devlin et al., 2018), a small revolution in the field of natural language processing was triggered. BERT is a huge neural network model based on the transformer architecture (Vaswani et al., 2017) and was pretrained on 3.3 billion words from the general text domain. While it builds upon recent approaches of pretraining contextual representations, particularly Semi-supervised Sequence Learning (Dai & Le, 2015), GPT (Radford et al., 2018), ELMo (Peters et al., 2018) and ULMFit (Howard & Ruder, 2018), and thus shares many similarities with them, BERT is arguably the most popular and best performing model[[1]](#footnote-1). In order to justify this popularity and outstanding performance, and thus also the decision to use BERT as the basis for this work, essential properties and design choices of BERT are summarized in the following subsections.

### Pretraining Contextual Representations

The pretraining of general-purpose language representation models overcomes the sparsity of training data that many NLP tasks suffer from, since pretraining can be performed on a vast amount of unannotated text data, of which there is plenty on the internet. The pretrained model can then be finetuned via supervised training for small-data downstream tasks, which generally results in substantial performance improvements compared to training on these datasets from scratch. This concept of transferring the knowledge learned while performing one tasks (i.e., the task performed during pretraining), to another, similar task (i.e., the target/ downstream task), is known as transfer learning and will be of importance when finetuning in section … The conventional BERT framework consists of pretraining and finetuning. 🡪 nach hinten verschieben

Approaches for such pretrained language representations are either context-free or contextual and contextual ones can in turn be unidirectional or bidirectional. When considering the word “tie” in the sentences “the game ended in a tie” and “I tie my hair back”, context-free representations like GloVe or word2vec

Die Ansätze für derartig vortrainierte Modelle lassen sich in context-freie und contextual models einteilen, wobei contextual models wiederum unidirectional or bidirectional sein können.

* When considering the sentences … and …, the word “tie” has the same

While context-free models like GloVe or word2vec have a fixed representation for each word in the vocabulary, and the word embeddings for “tie” in the sentences “the game ended in a tie” and “I tie my hair back” are therefore identical, contextual models create word embeddings that are based on the context in which the word is placed.

* Beispiele: tie
* “The game ended in a **tie**.”
* “I **tie** my hair back.”
* “I wear a **tie** with my suit.”

BERT, on the other hand, is the first deeply bidirectional model, as it looks at the content before and behind the word. This leads to an improved grasp of word meaning and context compared to previous approaches, also reflected in its state-of-the-art performance on eleven downstream tasks (Zitat).

### Pretraining Objectives

In order to learn the contextual representations mentioned in the previous section, BERT uses the masked language modelling pretraining objective

* Auch unter dem namen Cloze task bekannt
* 15% der Tokens in der Inputsequenz werden ausgewählt
  + 80% of the time: replace token with [MASK]
  + 10%: replace word with random word
  + 10%: keep word unchanged (to bias the representation towards the actual observed word
  + Das Model versucht dann to predict the originally masked token based on the left and right context of the sequence.
    - Vorteil davon (von dem gesamten drei verfahren): the transformer encoder does not know which words it will be asked to predict or which have been replaced by random words, so it is forced to keep a distributional contextual representation of every input

BERT additionally makes use of the NSP tasks,

* To understand the relationship between two sentences
* This is not directly captured by the language modelling
* Predicting if two input sentences are adjacent to each other
* During training sind 50% der Inputs tatsächlich nachfolgende Sätze und 50% sind random sentences from the pretraining corpus

### BERT workflow

* Input/ Output representations
* Ende: Durch open sourcing konnten viele Researchers die Ergebnisse reproduzieren und BERT für eigene Aufgaben nutzen
* BERT and conversational agents

## BERT and Conversational Agents

* Passage re-ranking (Vakulenko et al., 2021)
* Query rewriting (Voskarides et al., 2020)
* Sequence labelling (slot filling) (Chen et al., 2019)
* Answer span extraction (Vakulenko et al., 2021)
* Intent classification (Chen et al., 2019)
* Dialog breakdown detection (Sugiyama, 2021)
* Response selection (Han et al., 2021; Wang et al., 2021)
* Dialogue state tracking (Chao & Lane, 2019)
* (Zhou et al., 2020) used BERT for a conversational recommender system for predicting items (probability von Items aus Liste bestimmen, die dem Nutzer vorgeschlagen werden) and predicting topic (next topic that guides the user to the target topic)
* (Yang, W. et al., 2019) BERT is used as a reader in their BERTserini approach (BERT macht answer extraction) 🡪 was also integrated into a chatbot
  + BERTs open sourcing allowed many researchers to apply BERT for their own tasks
  + Kurzer Hintergrund zu conversational agents und BERT
  + Conversational agents in der Küche
  + Verbnidung zw. Den beiden herstellen
  + Problem von BERT anteasern
* Das Anpassen von BERT an eine bestimmte Domäne ist gut untersucht und es existieren unterschiedliche Ansätze.
* Die am häufigsten vertretenen Ansätze:
  + Pretrain from scratch
  + DAPT
  + TAPT

## Adapting BERT for Different Domains

* See <https://arxiv.org/abs/1812.11806> “An introduction to domain adaptation and transfer learning”
* Welche Ansätze existieren in der Literatur und was sind jeweils vor und nachteile
* Ich nehme DAPT, da …
* BERT wurde inzwischen für die verschiedensten Domänen angepasst, bspw. …
* BERT models for cooking domain sin dim nächsten Kapitel zusammengefasst
* BERT for cooking domain
* Es gibt schon arbeiten, die BERT für die Kochdomäne anpassen
* FoodBERT: weniger Daten, nur für sehr spezielle Aufgabe getestet und nicht auf CAs ausgerichtet
* MenuNER (eher die Restaurant domäne)
* Summary and Key Differentiators
* BERT ist toll und wird auch schon für CAs eingesetzt
* BERT fehlt domänenspez. Wissen
* BERT gibts für viele Domänen, allerdings noch nicht wirklich für die Kochdomäne
* FoodBERT und anderes cookingBERT paper + NER Paper zu FoodBase
* Datensätze enthalten häufig nur wenige Daten
* Beispiele, wann BERT im NLP angewandt wird/ werden kann:
* Foodie Fooderson
* <https://cseweb.ucsd.edu/~jmcauley/pdfs/emnlp19c.pdf>
* Conversational agents for the kitchen

Given the lack of exploration of neural embedding models for ingredient substitution and the challenges concerning evaluation, we propose several learningbased approaches for substitute generation and conduct both a ground truth based and a human evaluation. – FoodBERT

* Auswirkungen der Datenmenge auf DAPT kurz erläutern.

## BERT for the Cooking Domain

## Summary and Key Differentiators

Andere Arbeiten haben BERT zwar schon für die Kochdomäne angepasst, allerdings mach ich 2 Sachen anders:

* Größeren Datensatz für DAPT, was laut Literatur generell zu besseren Ergebnissen führen kann
* Bisherige Modelle (FoodBERT) nur auf sehr speziellen/ wenigen Aufgaben getestet. Ich will für mehrere Aufgaben gucken, ob sich CookBERT für conversational agents im Kochbereich eignet, indem für mehrere relevante Tasks ausgewertet wird.

Research question:

* **“What are the effects of domain adaptation in the performance of a pre-trained German BERT model on German legal downstream tasks?”. – Effects of inserting domain vocabulary …**
* ***How does cooking domain adaption of BERT affect the performance of CA-relevant tasks in/ from this domain?***

1. Note that recently published models RoBERTa Liu et al. (2019) and XLNet Yang, Z. et al. (2019). However, they are not considered in this thesis due to computational constraints. [↑](#footnote-ref-1)