

**CookBERT – Adapting BERT for the Cooking Domain**

Bachelor thesis in Media Informatics at the Institute for Language, Literature and Cultural Studies (I:IMSK)

Submitted by: Pascal Strobel

Address: Gluckstraße 3, 93053 Regensburg

E-Mail (university): pascal.strobel@stud.uni-regensburg.de

E-Mail (private): paschistrobel@web.de

Matriculation number: 2106133

First reviewer: Prof. Dr. Udo Kruschwitz

Second reviewer: PD Dr. David Elsweiler

Supervisor: Alexander Frummet (M. Sc.)

Current semester: 7

Submitted on: 30.2.2016

Contents

[1 Introduction 8](#_Toc96205571)

[2 Background and Related Work 11](#_Toc96205572)

[2.1 BERT 11](#_Toc96205573)

[2.1.1 Pretraining Contextual Representations 11](#_Toc96205574)

[2.1.2 Pretraining Objectives 12](#_Toc96205575)

[2.1.3 General BERT workflow 12](#_Toc96205576)

[2.2 BERT and Conversational Agents 12](#_Toc96205577)

[2.3 Adapting BERT for Different Domains 13](#_Toc96205578)

[2.4 BERT for the Cooking Domain 14](#_Toc96205579)

[2.5 Summary and Key Differentiators 14](#_Toc96205580)

[3 Methodology 14](#_Toc96205581)

[3.1 Data for DAPT 15](#_Toc96205582)

[3.2 Analyzing Domain Similarity 16](#_Toc96205583)

[3.3 Domain Vocabulary Insertion 17](#_Toc96205584)

[3.4 DAPT 19](#_Toc96205585)

[3.5 Implementation Details 19](#_Toc96205586)

[3.6 Finetuning 20](#_Toc96205587)

[3.6.1 Intent Classification 20](#_Toc96205588)

[3.6.2 Named-entity recognition 21](#_Toc96205589)

[3.6.3 Question Answering 21](#_Toc96205590)

[4 Evaluation 21](#_Toc96205591)

[4.1 Multi-class classification 21](#_Toc96205592)

[4.2 Named-entity recognition 21](#_Toc96205593)

[4.3 Question Answering 21](#_Toc96205594)

[5 Discussion 21](#_Toc96205595)

[6 Limitations 21](#_Toc96205596)

[7 Conclusion 22](#_Toc96205597)

[Bibliography 23](#_Toc96205598)

[Anhang A: Bausteine wissenschaftlicher Arbeiten 27](#_Toc96205599)

[A1 Theoretische Arbeit 27](#_Toc96205600)

[A2 Konstruktive Arbeit 27](#_Toc96205601)

[A3 Empirische Arbeit 27](#_Toc96205602)

[Erklärung zur Urheberschaft 29](#_Toc96205603)

Abbildungsverzeichnis (optional, in der Regel nicht notwendig)

[Abbildung 1: Blumen (Quelle, Jahr, Seitenzahl) **Fehler! Textmarke nicht definiert.**](#_Toc359834290)

Tabellenverzeichnis (optional, in der Regel nicht notwendig)

[Tabelle 1: Empfohlener Textumfang **Fehler! Textmarke nicht definiert.**](#_Toc359834281)

**Zusammenfassung**

Abstract

* Recent Fortschritt in NLP hat sich auch auf CA ausgewirkt, welche zunehmend ubiquitous werden und in vielen Bereichen des Lebens zu finden sind
* Früher regelbasiert, heute neuronale Netzte
* Eines davon ist BERT

# Introduction

Conversational agents (CAs) like Amazon’s Alexa or Apple’s Siri become more and more pervasive and are applied in a broad range of contexts, including health (Ni et al., 2017; Xu et al., 2019), elderly care (Bickmore et al., 2005), education (Graesser et al., 1999; Winkler et al., 2020), customer service (Cui et al., 2017) and home cooking (Angara et al., 2017; Chu, 2021). Although there are various types of conversational agents, which are titled and categorized very inconsistently in literature and media, they all provide an alternative to traditional methods for humans to seek for information by making the search process more conversational, mainly via written or spoken natural language (McTear, 2020, pp. 12–13). Users benefit from this more natural interaction as it promises an increased ease of use and speed of user requests as well as a convenient usage (Brandtzaeg & Følstad, 2017). While early approaches to create CAs were mainly based on handcrafted rules (e. g. Weizenbaum’s ELIZA (1966)), this now shifted towards the usage of large amounts of data to train powerful deep neural networks which can then gain a superb grasp of human language.

One of the most promising models in recent development is Bidirectional Encoder Representations from Transformers (BERT) - a huge neural network proposed by the Google AI team (Devlin et al., 2018) which is pretrained on 3.3 billion words from BooksCorpus (Zhu et al., 2015) and English Wikipedia. It builds upon previous approaches on 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 also follows the new trend in NLP, that is transfer learning - a technique of transferring knowledge gained from performing one task to another, similar one, which is beneficial compared to the resource intensive process of training neural networks (sometimes with several hundred million parameters) from scratch. However, what really sets BERT apart from previous approaches is that it is “the first deeply bidirectional, unsupervised language representation, pre-trained using only a plain text corpus” (Devlin & Chang, 2018, What Makes BERT Different?).

While it builds upon previous approaches on pretraining contextual representations and adopts many concepts, it is the first … . 🡪 das ganze in einem Satz verkürzt

This bidirectionality, combined with the self-attention mechanism, provides a better grasp of word meanings and context, which is reflected in achieving state-of-the-art performance on eleven NLP tasks, as reported by Devlin et al. (2018). BERT’s outstanding performance, followed by its open sourcing, ensured that it was subsequently also integrated into CA pipelines to solve tasks like passage re-ranking (Vakulenko et al., 2021), query rewriting (Vakulenko et al., 2021; Voskarides et al., 2020), sequence labelling (Chen et al., 2019), answer span extraction (Yang, W. et al., 2019) and intent classification (Chen et al., 2019).

As mentioned before, many CAs are applied in specific domains and thus encounter domain specific user requests. However, a recurring limitation of BERT in literature is its lack of domain specific knowledge due to pretraining being performed only on general text data. This can lead to performance loss on downstream tasks (Gururangan et al., 2020), and thus emphasizes the need to modify BERT for the desired domain.

Proceeding from these latest developments and findings, this bachelor thesis deals with the adaptation of BERT for one particular domain, i.e., the domain of cooking, aiming to provide a domain specific model that can be applied for various natural language understanding (NLU) tasks of CAs for the kitchen. The cooking domain is believed to be a pertinent context for multiple reasons. First, cooking provides situations where traditional search is rather inconvenient, as users are multi-tasking, and their hands are occupied. Another reason is the arguably strong demand for CAs in the kitchen, as Google Home Devices were used for more than 16 million recipes in 2018th holiday season, passing one million on Christmas day alone (Huffman, 2019). Additionally, it has been argued in the past, that aiding in the kitchen (e. g. via recipe recommendations) could potentially lower the barriers to healthier cooking and thus improve the nutrition of people (Elsweiler et al., 2015; Elsweiler et al., 2017; Freyne & Berkovsky, 2010).

By adapting BERT to the specific domain of cooking, it is hoped to improve its natural language understanding of this domain. Therefore, the model is evaluated for several CA relevant downstream tasks, including named entity recognition (NER), intent classification and question answering (in the sense of answer span extraction). Being able to improve the performance of the BERT model for such tasks can be a crucial step towards building a sophisticated CA for the kitchen.

…

The adaption of BERT on cooking domain is aiming to create a model that is applicable for various NLU tasks of future potential conversational agents for the kitchen. The performance of the model is thus evaluated on/ against a variety of downstream tasks relevant to conversational agents. The thesis aims to answer the underlying research question that is:

***How does cooking domain adaption of BERT affect the performance of CA-relevant tasks in/ from this domain?***

* ***RQ am Ende der Literature review.***
* Slot filling und NER sind beides sequence labelling tasks und an sich sehr ähnlich

🡪 vielleicht hier die tasks noch gar nicht aufzählen, sondern dann erst im related work (und hier nur erwähnen, dass es in mehreren Pipelines eingesetzt wird).

# Background and Related Work

To set the context for this thesis and motivate the research question as well the methodological decisions taken, this chapter covers related work from research contributions across diverse fields of computer and information science, ranging from conversational agents to natural language understanding to artificial neural networks. It also covers fundamental concepts of BERT The chapter is arranged as follows:

* Section 2.1 enthält den Background zu BERT und seine core concepts, die zum Verständnis diese Arbeit beitragen
* Section

## 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 tackles the problem of training data sparsity in NLP. Derartige Modelle können zunächst auf riesigen Mengen and unlabelled text data (that is ausreichend verfügbar im Netz) pretrained werden and the resulting model can then be finetuned for the specific small-data NLP task. Dieses Konzept ist unter dem Namen Transfer-Learning bekannt und beschreibt das Transferieren von gelerntem Wissen beim Erfüllen einer Aufgabe, auf eine andere Aufgabe.
* Early pretraining language respresentations approaches, wie GLoVE oder Word2Vec waren context-free, bsp. Satz einfügen
* Im gegensatz dazu stehen contextual models, welche wiederum unidirectional or bidirectional sein können
* Beispiele: tie
* “The game ended in a **tie**.”
* “I **tie** my hair back.”
* “I wear a **tie** with my suit.”
* Ende: Durch open sourcing konnten viele Researchers die Ergebnisse reproduzieren und BERT für eigene Aufgaben nutzen
* BERT and conversational agents

### Pretraining Objectives

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

BERT additionally makes use of the NSP tasks,

* Was soll es davon lernen

### General BERT workflow

## BERT and Conversational Agents

* + 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

* 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 …**

# Methodology

* Need for Domain Adaption:
* Vokabular overlap zeigen
* Aufzeigen, welche Wörter nicht im Vokabular sind, und wie diese dann Tokenisiert werden
* Verwendete Daten für DAPT
* Welcher Datensatz
* Wie aufgebaut 🡪 Wie viele Daten 🡪 bereinigt im Vergleich zu Recipe1M
* Vokabular Auswahl und insertion
* DAPT
* Worauf wird BERT trainiert (MLM)
* Tools and environment
* Implementation details
  + Welche learning rate, wie viele Epochen, …
* Finetuning
* NER
  + Wie wirds gemacht? 🡪 Extra Layer on top
  + Welche Daten wurden verwendet
  + Wie wurden Daten vorbereitet?

## Data for DAPT

In order to adapt the BERT base model to the cooking domain, a large amount of natural language plain text data of that domain is required.

* Wie in section … schon erklärt, sind führen generell mehr Daten bei DAPT auch zu besserer Performance (wie eigentlich immer in machine learning). Deshalb wird ein möglichst großer Korpus für DAPT ausgewählt.
* FoodBERT verwendet den Recipe1M Datensatz bzw. Die Instruktionen davon
* In meinem Ansatz wird ein noch größerer Korpus verwendet
* Datenmengen bei DAPT in der Literatur
* FinBERT: TRC2-financial, 46.143 documents, 29 million words, 400k sentences
* HateBERT: RAL-E, 1.478.348 messages, 43.379.350 million tokens,
* BioBERT: PubMed Abstracts = 4.5B words + PMC Full-text articles 13.5B words
* CSBERT: 40.505.050 dialogues and 317.093.459 turns coming from all available customer service intents
* TOD BERT: 100.707 Dialoge, 1.388.152 Utterances von verschiedenen task-oriented dialogue Datensätzen
* MenuNER: YELP Dataset (only reviews from the restaurant category), 15.000.000 sentences
* Pretraining BERT on domain resources for short answer grading: Textbooks und QAs mit 1.1m, 0.6m und 1.3m Wörtern
* Außerdem gibt es Paper, die sagen, dass mehr pretraining Data (When do you need billions of words of pretraining data) bzw. Mehr DAPT data (sinnvoll ist)
* Aus diesem Grund wurde der RecipeNLG Datensatz verwendet

RecipeNLG Datensatz:

* Datensatz mit über 2 Millionen Rezepten (2.231.142 um genau zu sein) und ist damit the largest publicly available dataset in this domain
* Datensatz ist wie folgt strukturiert:
* Title: Rezepttitel
* Ingredients: Zutaten mit mengenangaben
* Directions: Instruktionen 🡪 das habe ich verwendet
* Link: link zum Rezept
* Source: Gathered (74%) oder von Recipes1M (26%)
* NER: named food entities; extracted mit einem NER
* Beispielinstruktion genauer ansehen 🡪 welchen einfluss die Eigenschaften der Instruktionen haben, wird in der Limitierungs-section diskutiert
* Baut auf Recipe1M+ Datensatz auf und erweitert diesen mit über 1 Millionen neuen, korrigierten einträgen + Duplikatscheck/ Deduplication
* Die additional recipes wurden von mehreren cooking web pages via web scraping gesammelt.
* Das führt zu 2.231.142 individuellen Rezepten
* Datenanalyse machen: Durchschnittliche Tokens pro Rezept

## Analyzing Domain Similarity



Before the actual DAPT, the similarity of the target domain (cooking) and BERT’s pretraining domain was analyzed. The approach for the analysis is adopted from Gururangan et al. (2020) and quantifies the domain similarity based on the vocabulary overlap. RecipeNLG, Recipe1M+ and the WikiBook corpus from BERT pretraining were used for the analysis. As BERT’s original pretraining data is not distributed, a Wikipedia dump (Merity et al., 2016)(515MB) and randomly sampled books from the “Homemade BookCorpus” (Kobayashi, 2018)(444MB) were used to reconstruct a similar corpus. From RecipeNLG and Recipe1M+, the recipe instructions were used as corpus data (1GB and 619MB respectively). A vocabulary consisting of the 10.000 most frequent unigrams (after lowercasing and removal of stopwords and punctuation) was then created for each of the three corpora.

The vocabulary overlap between the corpora is illustrated in Fig 4. It shows a strong overlap between Recipe1M+ and RecipeNLG, which is not surprising given the fact that both corpora are from the cooking domain and, moreover, Recipe1M is a subset of RecipeNLG. However, the overlap between the WikiBooks corpus and the two cooking corpora is quite small, emphasizing the data shift between the cooking domain and the general text domain. Furthermore, this simple analysis indicates the degree of benefit to be expected by adapting BERT for the cooking domain, as the potential for DAPT is higher, the more dissimilar the domains are (Gururangan et al., 2020, p. 3).

## Domain Vocabulary Insertion

* BERTs handelt OOV Vokabeln, indem es unbekannte Wörter/ Wörter, die nicht im Vokabular (ca. 30000 tokens) in Subtokens teilt.
* Die Repräsentation eines Wortes, welches in mehrere Subtokens aufgeteilt wird, setzt sich also aus mehreren einzelnen Repräsentationen der subtokens zusammen
* Das ist zwar meistens nicht weiter problematisch, da durch einbeziehen des Kontextes durch den Self-attention mechanismus von BERT auch diese Subtokens zu guten Repräsentationen geformt werden
* Allerdings gibt es einige kochspezifische Wörter, welche nicht im BERT Vokabular sind (z.B. Zucchini). BERT handelt diese zwar ganz gut, indem derartige OOV Wörter in Subtokens zerlegt wird, wobei jedes Subtoken eine einzelne Repräsentation erhält. Durch BERTs self-attention mechanismus wird zwar der Kontext mit einbezogen, was generell zu guten Ergebnisse führen zu schein, allerdings ist es reasonable, dass die Zerlegung von Kochspezifischem Vokabular (Zucchini in zu cc ini) nicht zu so einer guten Repräsentation führt, wie wenn es eine gesamte Repräsentation für das Wort zucchini gibt.
* Manche Arbeiten zeigen, dass Vokabular-Adding positiven Effekt hat (exBERT, SciBERT verwendet auch eigenes SciVocab, was zu 0.60% increase on F1 score führt)
* Aus diesem Grund wird das Vokabular von BERT mit den wichtigsten Vokabeln der Kochdomäne erweitert (die es noch nicht enthält)
* Wie hab ich das gemacht? Für das erstellte Vokabular in der vorherigen Section wurde zusätzlich die Frequency des jeweiligen Wortes gespeichert.
* Alle Wörter mit einer Häufigkeit von mindestens 1000, und die noch nicht im BERT vocab enthalten waren 🡪 Das hat den grund, dass die Wörter häufig genug vorkommen, um gute Repräsentationen zu erlernen, nachdem die Gewichtungen für diese quasi von 0 initialisiert werden müssen. Kommen sie zu selten vor, dann kann dies evtl. nicht richtig erlernt werden
* Wie in der vorherigen Section schon erwähnt wurden jedes Rezept erst tokenisiert, lowercased, und stop-words und punctuation wurde entfernt.
* Insgesamt wurden 1229 Wörter zum Vokabular hinzugefügt
* Bsp. Wörter, die nicht enthalten waren: baguette, zucchini, preheat, tortilla, oregano, grilled, smoothie, appetizer, oiled, skillet, eggplant, breadcrumb, sauteed, teaspoonful

|  |  |  |
| --- | --- | --- |
| Word | After tokenization |  |
| baguette | bag ##uet ##te, |  |
| cranberry | cr ##an ##berry |  |
| thicken | thick ##en |  |
| caramelized | cara ##mel ##ized |  |
| zucchini | zu ##cchi ##ni |  |
| preheat | pre ##hea ##t |  |
| tortilla | tor ##till ##a |  |
| eggplant | egg ##pl ##ant |  |

* Irgendwo noch mit reinbringen, wie FoodBERT das gemacht hat: die hatten eien Liste mit Zutaten und haben dann alle, die öfter als 10 mal vorkamen, zum Vokabular hinzugefügt. 🡪 Allerdings: es gibt auch viele Wörter, die nicht zutaten sind, die im Kochjargon vorkommen (skillet, …), diese werden aber vernachlässigt
* Fügen alle Zutaten hinzu, wenn diese mind. 10 mal vorkommen. Allerdings nehme ich an, dass dies nicht ausreichend ist, da die repräsentationen komplett neu initialisiert wurden und erst trainiert werden müssen, wobei 10 beipsiele nicht ausreichen
* With the insertion of our custom legal vocabulary, the tokenization provides different segmentation of words affecting the input sequence for BERT and we expect to slightly improve the performance in the same way SciBERT did by creating their own SciVocab (subsection 3.2.2).
* Effects of inserting domain vocabulary (S. 52)

## DAPT

* BERT kommt in mehreren Ausführungen (BERT large, BERT base, BERT base cased, BERT base uncased, …)
* Als ausgangspunkt wurde der BERT base uncased checkpoint verwendet

DAPT: **BERTBASE**

* If your task has a large domain-specific corpus available (e.g., "movie reviews" or "scientific papers"), it will likely be beneficial to run additional steps of pre-training on your corpus, starting from the BERT checkpoint. (https://github.com/google-research/bert#pre-training-tips-and-caveats)
* The learning rate we used in the paper was 1e-4. However, if you are doing additional steps of pre-training starting from an existing BERT checkpoint, you should use a smaller learning rate (e.g., 2e-5).

## Implementation Details

* Learning rate, epochen, Model startpunkt, …
* Dauer des Learning vorgangs
* Verwendete Library: Huggingface
* Verwendete Umgebung: Google Colab
* GPU: P100 GPU von Google Colab+

## Finetuning

### Intent Classification

* Multi-class classification problem
* Siehe paper von Frummet für Vorgehen (an dem orientiere ich mich eben)
* Datensatz von Frummet
* Alles so wie Frummet gemacht
* 85% train, 15% test
* No resampling
* No stopword removal
* Stratified sampling for 10 fold cross validation
* To avoid catastrophic forgetting: lower learning rate of 2e-5
* Training for 4 epochs, dropout probability of 10%, batch\_size 32
* Early stopping was included
* Wegen computing limitations wurde eine maximale Sequ. Length von 256 verwendet. D.h. wenn mehrere Turns mit angehänt wurden, wurden nur die letzten 256 tokens verwendet.
* Auswertung mit drei contexten:
  + 1. No context
    2. 1 prev turn
    3. All prev turns
* Anders als Frummet gemacht:
* Frummet hat 11 binary classifiers mit jeweils einem classificationHead der Dimension 768,2. Ich habe nur einen classifier mit classification Head mit dimension 768,11.
* Class weights were adjusted by FARMs datasilo 🡪 evlt auch machen, siehe <https://discuss.huggingface.co/t/class-weights-for-bertforsequenceclassification/1674/7>

### Named-entity recognition

### Question Answering

# Evaluation

## Multi-class classification

## Named-entity recognition

## Question Answering

# Discussion

# Limitations

* Aus ressourcengründen wurde nur BERT model als Basis hergenommen 🡪 inzwischen existieren mehrere Verbesserte/ optimierte Modelle, die BERT outperformen
* Es wurde nur der Aspekt der Domänenadaption angeguckt. Andere Aspekte, die für Conversational Agents und die natürliche Sprache, die diese begegnen (Dialoge) werden nicht berücksichtigt.
* Vielle ist BERT mit history embeddings als ausgangsmodell besser
* Viell. Ist ConvBERT, der speziell schon auf Konversationsdaten vortrainiert wurde besser als standard BERT
* Es ist unklar, wie sich das Einfügen von zusätzlichen Vokabeln auf die Perf ausgewirkt hat (Vorherige Arbeiten haben zwar schon positive Effekte festgesetellt)
* Cooking recipes eigenschaften
* <https://jurnal.polban.ac.id/inggris/article/view/3467>
* Meistens im Imperative (selten im declarative) formuliert
* Sind oft keine Grammatikalischen Sätze, Wörter, v.a. Artikel fehlen (“Add egg and beat well”)
* Sind nur Rezepte, und damit keine wirkliche natürliche Sprache. Besser wäre vielleicht natürliche Konversationen übers Kochen zu verwenden (z.B. via Podcasts oder Untertitel)
* Kochdomäne umfasst vielleicht mehr, als nur die Rezeptdaten
* Enthalten Abkürzungen (tbsp, oz., hrs)

# Conclusion

* Vorschlag: andere Datenquelle zum Pretrainnen hernehmen, welche näher an der natülichen Sprache ist 🡪 Kommentare von Rezepten, Koch FAQs, Untertitel von Kochshows, …

# Bibliography

Angara, P., Jimenez, M., Agarwal, K., Jain, H., Jain, R., Stege, U., Ganti, S., Müller, H. A., & Ng, J. W. (2017). Foodie fooderson a conversational agent for the smart kitchen. *CASCON*, 247–253.

Bickmore, T. W., Caruso, L., & Clough-Gorr, K. (2005). Acceptance and usability of a relational agent interface by urban older adults. *CHI'05 Extended Abstracts on Human Factors in Computing Systems*, 1212–1215. https://doi.org/10.1145/1056808.1056879

Brandtzaeg, P. B., & Følstad, A. (2017). Why People Use Chatbots. In I. Kompatsiaris, J. Cave, A. Satsiou, G. Carle, A. Passani, E. Kontopoulos, S. Diplaris, & D. McMillan (Eds.), *Lecture Notes in Computer Science. Internet Science* (Vol. 10673, pp. 377–392). Springer International Publishing. https://doi.org/10.1007/978-3-319-70284-1\_30

Chen, Q., Zhuo, Z., & Wang, W. (2019). *BERT for Joint Intent Classification and Slot Filling*. http://arxiv.org/pdf/1902.10909v1

Chu, J. (2021, September 24–26). Recipe Bot: The Application of Conversational AI in Home Cooking Assistant. In *2021 2nd International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE)* (pp. 696–700). IEEE. https://doi.org/10.1109/ICBASE53849.2021.00136

Cui, L., Huang, S., Wei, F., Tan, C., Duan, C., & Zhou, M. (2017). Superagent: A customer service chatbot for e-commerce websites. *Proceedings of ACL 2017, System Demonstrations*, 97–102.

Dai, A. M., & Le, Q. V. (2015). Semi-supervised Sequence Learning. *Advances in Neural Information Processing Systems*, *28*, 3079–3087. http://arxiv.org/pdf/1511.01432v1

Devlin, J., & Chang, M.-W. (2018). *Open Sourcing BERT: State-of-the-Art Pre-training for Natural Language Processing.* Google AI. https://ai.googleblog.com/2018/11/open-sourcing-bert-state-of-art-pre.html

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018, October 11). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. http://arxiv.org/pdf/1810.04805v2

Elsweiler, D., Harvey, M., Ludwig, B., & Said, A. (2015). Bringing the "healthy" into Food Recommenders. *DMRS*, 33–36.

Elsweiler, D., Trattner, C., & Harvey, M. (2017). Exploiting food choice biases for healthier recipe recommendation. *Proceedings of the 40th International Acm Sigir Conference on Research and Development in Information Retrieval*, 575–584. https://doi.org/10.1145/3077136.3080826

Freyne, J., & Berkovsky, S. (2010). Intelligent food planning: personalized recipe recommendation. *Proceedings of the 15th International Conference on Intelligent User Interfaces*, 321–324. https://doi.org/10.1145/1719970.1720021

Graesser, A. C., Wiemer-Hastings, K., Wiemer-Hastings, P., & Kreuz, R. (1999). AutoTutor: A simulation of a human tutor. *Cognitive Systems Research*, *1*(1), 35–51. https://doi.org/10.1016/S1389-0417(99)00005-4

Gururangan, S., Marasović, A., Swayamdipta, S., Lo Kyle, Beltagy, I., Downey, D., & Smith, N. A. (2020, April 23). *Don't Stop Pretraining: Adapt Language Models to Domains and Tasks*. http://arxiv.org/pdf/2004.10964v3

Howard, J., & Ruder, S. (2018, January 18). *Universal Language Model Fine-tuning for Text Classification*. http://arxiv.org/pdf/1801.06146v5

Huffman, S. (2019). *Here’s how the Google Assistant became more helpful in 2018.* Google. https://www.blog.google/products/assistant/heres-how-google-assistant-became-more-helpful-2018/

Kobayashi, S. (2018). *Homemade BookCorpus*. https://github.com/BIGBALLON/cifar-10-cnn

Liu, Y., Ott, M., Goyal, N., Du Jingfei, Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019, July 26). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. http://arxiv.org/pdf/1907.11692v1

McTear, M. (2020). Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots. *Synthesis Lectures on Human Language Technologies*, *13*(3), 1–251. https://doi.org/10.2200/S01060ED1V01Y202010HLT048

Merity, S., Xiong, C., Bradbury, J., & Socher, R. (2016, September 26). *Pointer Sentinel Mixture Models*. http://arxiv.org/pdf/1609.07843v1

Ni, L., Lu, C., Liu, N., & Liu, J. (2017). MANDY: Towards a Smart Primary Care Chatbot Application. In J. Chen, T. Theeramunkong, T. Supnithi, & X. Tang (Eds.), *Communications in Computer and Information Science. Knowledge and Systems Sciences* (Vol. 780, pp. 38–52). Springer Singapore. https://doi.org/10.1007/978-981-10-6989-5\_4

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018, February 15). *Deep contextualized word representations*. http://arxiv.org/pdf/1802.05365v2

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving language understanding by generative pre-training.*

Vakulenko, S., Longpre, S., Tu, Z., & Anantha, R. (2021). Question Rewriting for Conversational Question Answering. *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, 355–363. https://doi.org/10.1145/3437963.3441748

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017, June 12). *Attention Is All You Need*. http://arxiv.org/pdf/1706.03762v5

Voskarides, N., Li, D., Ren, P., Kanoulas, E., & Rijke, M. de (2020). Query Resolution for Conversational Search with Limited Supervision. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 921–930. https://doi.org/10.1145/3397271.3401130

Weizenbaum, J. (1966). ELIZA-a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, *9*(1), 36–45.

Winkler, R., Hobert, S., Salovaara, A., Söllner, M., & Leimeister, J. M. (2020). Sara, the Lecturer: Improving Learning in Online Education with a Scaffolding-Based Conversational Agent. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–14. https://doi.org/10.1145/3313831.3376781

Xu, L., Zhou, Q., Gong, K., Liang, X., Tang, J., & Lin, L. (2019). End-to-End Knowledge-Routed Relational Dialogue System for Automatic Diagnosis. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*, 7346–7353. https://doi.org/10.1609/aaai.v33i01.33017346

Yang, W., Xie, Y., Lin, A., Li, X., Tan, L., Xiong, K., Li, M., & Lin, J. (2019). End-to-End Open-Domain Question Answering with BERTserini, 72–77. https://doi.org/10.18653/v1/N19-4013

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le V, Q. (2019, June 19). *XLNet: Generalized Autoregressive Pretraining for Language Understanding.*

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *Proceedings of the IEEE International Conference on Computer Vision*, 19–27.

Anhang A: Bausteine wissenschaftlicher Arbeiten

## A1 Theoretische Arbeit

1. Fragestellung (Ziele, Motivation)
2. Überblick über Stand der Forschung und Technik (dabei Bewertung der Ansätze, Beispiele, Identifikation von Defiziten)
3. Synthese: Erstellung einer Gesamtschau (allgemeine Prinzipien, Beschreibung einer eigenen Sicht auf das Problem, Formulierung von Empfehlungen )
4. Zusammenfassung (Was wurde in der Arbeit erreicht, Erklärung des Nutzens für andere)
5. Ausblick (optional)

## A2 Konstruktive Arbeit

1. Problemstellung (Ziele, Ausgangspunkt, Vorgesehener Benutzerkreis, Bedürfnisse der Benutzer)
2. Stand der Forschung und Technik (Bisherige Lösungen, Defizite)
3. Eigenes Konzept (Lösungsansatz, allgemeines Prinzip, Werkzeuge z.B. Programmiersprachen )
4. Vorgehensweise (Beschreibung der durchgeführten Arbeitsschritte)
5. Ergebnis (Vorstellung des System z.B. Screenshots mit Erläuterungen)
6. Evaluation des System (optional, was soll evaluiert werden, welche Methode, Ablauf, Ergebnisse)
7. Zusammenfassung (Was wurde in der Arbeit erreicht; Erklärung des Nutzens für andere)
8. Ausblick (optional)

## A3 Empirische Arbeit

1. Fragestellung der Arbeit (Was soll untersucht werden, warum)
2. Stand der Forschung und Technik (Bewertung der Untersuchungs-Ansätze und Ergebnisse, Identifikation von Defiziten)
3. Präzisierung der Fragestellung (Hypothesen)
4. Untersuchungsmethodik
5. Untersuchungsablauf (Untersuchungsmaterial, Raum, Probandenrekrutierung etc.)
6. Ergebnisse (Darstellung der Ergebnisse in sinnvoller Reihenfolge, Gesamtüberblick, Einzelergebnisse z. B. geordnet nach Testcases)
7. Zusammenfassung (Was wurde erreicht, Rückbezug zu Zielen, Hypothesen, Nutzen, Erkenntnisse für weitere Untersuchungen)
8. Ausblick (optional)

Erklärung zur Urheberschaft

Ich habe die Arbeit selbständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt, sowie alle Zitate und Übernahmen von fremden Aussagen kenntlich gemacht.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt.

Die vorgelegten Druckexemplare und die vorgelegte digitale Version sind identisch.

|  |  |  |
| --- | --- | --- |
| Ort, Datum |  | Unterschrift |

Erklärung zur Lizenzierung und Publikation dieser Arbeit

**Name:** Pascal Strobel

**Titel der Arbeit:** CookBERT

In der Regel räumen Sie mit Abgabe der Arbeit dem Lehrstuhl für Medieninformatik nur zwingend das Recht ein, dass die Arbeit zur Bewertung gelesen, gespeichert und vervielfältigt werden darf. Idealerweise liefern Seminararbeiten, Projektdokumentationen und Abschlussarbeiten aber einen Erkenntnisgewinn, von dem auch andere profitieren können. Wir möchten Sie deshalb bitten, uns weitere Rechte einzuräumen, bzw. idealerweise Ihre Arbeit unter eine freie Lizenz zu stellen.

Die in unseren Augen praktikabelsten Lösungen sind vorselektiert.

Hiermit gestatte ich die Verwendung der **schriftlichen Ausarbeitung** zeitlich unbegrenzt und nicht-exklusiv unter folgenden Bedingungen:

Nur zur Bewertung dieser Arbeit

Nur innerhalb des Lehrstuhls im Rahmen von Forschung und Lehre

Unter einer Creative-Commons-Lizenz mit den folgenden Einschränkungen:

BY – Namensnennung des Autors

NC – Nichtkommerziell

SA – Share-Alike, d.h. alle Änderungen müssen unter die gleiche Lizenz gestellt werden.

(An Zitaten und Abbildungen aus fremden Quellen werden keine weiteren Rechte eingeräumt.)

Außerdem gestatte ich die Verwendung des im Rahmen dieser Arbeit erstellen **Quellcodes** unter folgender Lizenz:

Nur zur Bewertung dieser Arbeit

Nur innerhalb des Lehrstuhls im Rahmen von Forschung und Lehre

Unter der CC-0-Lizenz (= beliebige Nutzung)

Unter der MIT-Lizenz (= Namensnennung)

Unter der GPLv3-Lizenz (oder neuere Versionen)

(An explizit mit einer anderen Lizenz gekennzeichneten Bibliotheken und Daten werden keine weiteren Rechte eingeräumt.)

Ich willige ein, dass der Lehrstuhl für Medieninformatik diese Arbeit – falls sie besonders gut ausfällt - auf dem Publikationsserver der Universität Regensburg veröffentlichen lässt.

Ich übertrage deshalb der Universität Regensburg das Recht, die Arbeit elektronisch zu speichern und in Datennetzen öffentlich zugänglich zu machen. Ich übertrage der Universität Regensburg ferner das Recht zur Konvertierung zum Zwecke der Langzeitarchivierung unter Beachtung der Bewahrung des Inhalts (die Originalarchivierung bleibt erhalten).

Ich erkläre außerdem, dass von mir die urheber- und lizenzrechtliche Seite (Copyright) geklärt wurde und Rechte Dritter der Publikation nicht entgegenstehen.

Ja, für die komplette Arbeit inklusive Anhang

Ja, für eine um vertrauliche Informationen gekürzte Variante (auf dem Datenträger beigefügt)

Nein

Sperrvermerk bis (Datum):

Regensburg, 10.03.2022

|  |  |  |
| --- | --- | --- |
| Ort, Datum |  | Unterschrift |

Inhalt des beigefügten Datenträgers

Beispiel (Ordner + Beschreibung):

|  |  |  |
| --- | --- | --- |
| /1\_Ausarbeitung | Die schriftliche Ausarbeitung als PDF und DOC | |
| /2\_Code | Quellcode und kompilierte Anwendung des Prototypen | |
| /3\_Studie/Design | Fragebogen und Script für die Benutzerstudie | |
| /3\_Studie/Rohdaten | Rohdaten der Studie im CSV-Format, inkl. Beschreibung der Felder | |
| /4\_Quellen | Alle in der Arbeit zitierten Quellen im PDF-Format | |
| /5\_Bilder | Alle selbst erstellten und aus anderen Quellen übernommenen Bilder | |
| /6\_Vorträge | Folien von Antritts- und Abschlussvortrag im PDF-Format | |
| /7\_Sonstiges | Notizen aus Besprechungen, Gedanken, … | |
|  | |

[Datenträger (CD, SD-Karte, o.ä.) hier oder auf Umschlaginnenseite einkleben]

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)