MSE | MASTER OF SCIENCE IN ENGINEERING

Hes·so
Haute Ecole Spécialisée de Suisse occidentale
Fachhochschule Westschweiz
University of Applied Sciences and Arts Western Switzerland

Master of Science HES-SO in Engineering
Av. de Provence 6
CH-1007 Lausanne

Master of Science HES-SO in Engineering

Orientation: Information and Communication Technologies (ICT)

GraphQA, a Deep Retrieval Chatbot

A Multi-hop Conversational Question-Answering Chatbot using Sub-Knowledge Graphs

Author:

Romain Claret

Under the direction of:
Prof. Dr. Jean Hennebert
HES-SO//Fribourg
Institute of Complex Systems (iCoSys)

External expert:
Prof. Dr. Michael Ignaz Schumacher
HES-SO//Valais
Institute of Business Information Systems

Fribourg, HES-SO//Master, February 6, 2020

Project Specification

This project specification for the Master's Thesis has been accepted by Romain Claret (the student) and Jean Hennebert (the supervisor) on the 4th of October 2019 at HES-SO//Fribourg.

Introduction

New technologies are revolutionizing the way humans access and consume information from multiple platforms and providers. Thanks to the emergence of increasingly powerful Artificial Intelligence (AI) algorithms, particularly in the field of Natural Language Processing (NLP), conversational agents, commonly known as chatbots, have come a long way and became popular among information consumers. As it is in late 2019, chatbots are all still Artificial Narrow Intelligence (ANI)[1]. Even if they are improving at providing meaningful sentences, they cannot generalize the tasks toward human-like conversations. Tasks such as understanding and keeping track of context in the long term, or even being intuitive and initiating meaningful conversation, have yet to be accomplished. Nonetheless, as research progress, chatbots are provided new tools which are making them step by step closer to complete human-like discussions, slowly progressing towards Artificial General Intelligence (AGI) chatbots.

Aim of the Study

In harmony with the author's interest, the thesis' orientation goes toward research. Indeed, the study will attempt to explore approaches to get closer to general conversational agents as a premise to AGI. As a fulfillment of the academic requirements, the study will include an experimental part with one or many Proofs of Concept.

Project's Overall Scope

The study is focusing on the English language as an attempt to increase the number of compatible datasets and make community accessible solutions. Complementarily, as the time for the thesis is limited to 19 weeks, the outcomes narrow at providing research conclusions and Proof of Concept (POC) solutions. We will be focusing at exploring two types of systems for Question Answering (QA) chatbots. The first type will produce straight to the point answers, and the second type will generate sentences as answers. Finally, the review of the risks and ethical problems that could be raised by the development of such solutions are not part of this work.

[1] The State of AI Report 2019 report by Nathan Benaich and Ian HogarBenaich et al., 2019

Industrial Interest

iCoSys, the Institut of Complex Systems at the University of Applied Sciences and Arts at Fribourg, Switzerland, is interested in the results of this study for their *AI-News* project[2]. Its goal is to provide a chatbot-based system as a tool to press readers, to help them narrow their interests and deliver the right information. This project is in collaboration with the *Swiss Innovation Agency* from the Swiss Confederation, *La Liberté*, the daily newspaper from Fribourg and *Djebots*, a startup selling narrow chatbots.

Research Questions

We articulate here a set of questions as a driver to our research work. From these questions are declined objectives, and from objectives are declined milestones framing the plan. We also hope to provide meaningful answers to these questions at the end of the thesis.

- What are the components to make QA chatbots?
  - How to tune QA chatbots to make them as human-like as possible?
  - How to tune such systems for the field of journalism?
- What is the state of the art for generative QA chatbots?
  - What are the components to make make generative QA chatbots?
  - Are generative chatbots only as good as the data they consume?
  - Could generative chatbots be a step toward AGI?

Objectives

Intrinsic

This subsection presents the general objectives related to the master's thesis.

Primaries

- Suggest project specification and planning.
- Analyze the state of the art of existing technologies and technics of QA systems and generative AI.
- Overview digital transformation in journalism and review the current status of the AI-News project.
- Document the study and write the thesis.

Fact-based QA Chatbot

The first objective is to make a state of the art software that takes a question as input and outputs a response (See Figure 1).

[2] `AINews.ch`

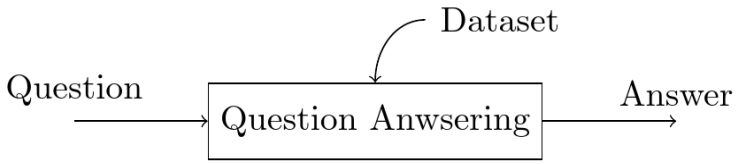


Figure 1: Suggested QA diagram

Primaries

- Select existing papers and projects treating the subject as a starting point.
- Identify relevant datasets.
- Develop one or more POC.
- Test and evaluate solutions.
- Suggest improvements, possible continuation, and future outcomes.

Secondaries

- Extended the QA chatbot using tailored knowledge.

Generative QA Chatbot

The second objective is to improve the output from the prior objective into enhanced answers (See Figure 2).

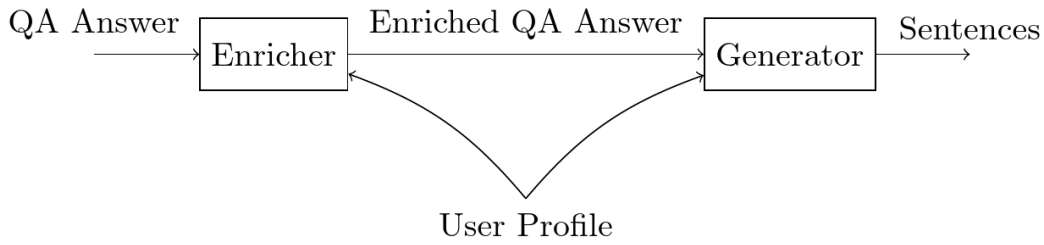


Figure 2: Suggested Generative QA diagram

Primaries

- Investigate a rule-based system for keyword enrichment.
- Generate sentences with keywords.
- Identify relevant datasets.
- Develop one or more POC.
- Test and evaluate solutions.
- Suggest improvements, possible continuation, and future outcomes.

Secondaries

- Use advanced strategies to enrich keywords.
- Use advanced text generation technics such as GTP-2[3].
- Use user profiles to customize the outputs.

[3] OpenAI's GTP-2 AlgorithmRadford et al., 2018

Plan

Contraints

Timeframe: 19 weeks
Starting date: 16.09.2019
Ending date: 07.02.2020

Methodologies

For consistency, the project is split into two methodological parts. The first third, as the project's orientation is going toward information gathering and self-study, uses a standard sequential project management methodology. For the next two-thirds of the project, the author is using an agile methodology intending to reach incremental progress while exploring.

Back to level Milestones

(6 weeks) First third of the study, from **16.09.19 to 25.10.19**.

M1. Initial Master's Thesis (MT) plan and project specification

M2. Review the state of the art of the NLP and Natural Language Understanding (NLU) technologies and refine the plan if needed.

Diving into the subject Milestones

(13 weeks) From 28.10.19 to 07.02.20, the following two-third of the thesis is composed 6 of sprints of two weeks and one week to finalise the thesis.

M3. Basic QA Chatbot

M4. Evaluation of basic QA Chatbot

M5. Basic generative QA Chatbot

M6. Evaluation of basic generative QA Chatbot

Gantt

The Figure 3 represents the chart for the initial plan.

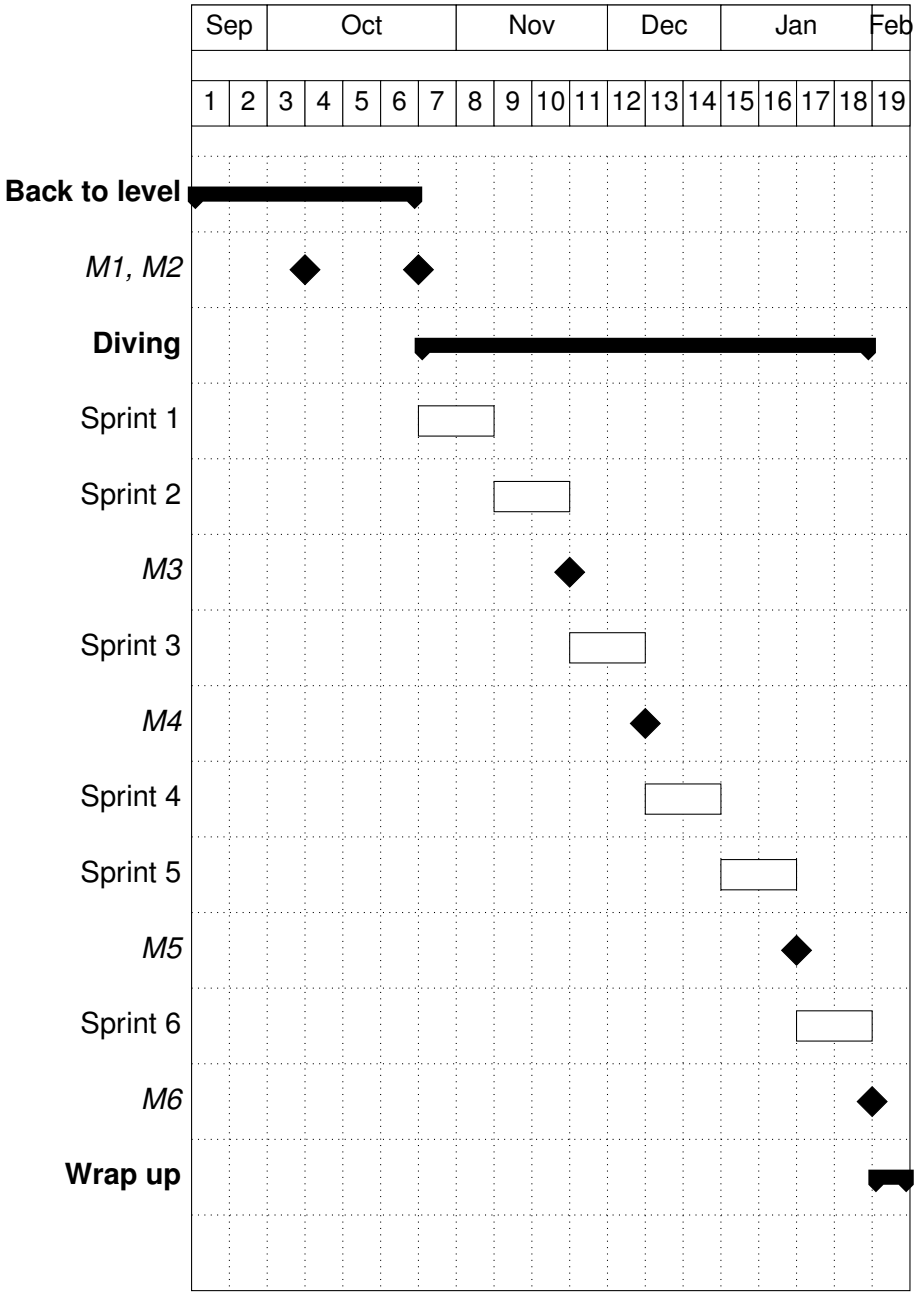


Figure 3: Project Specificiation Gantt Chart

Accepted by the HES-SO//Master (Switzerland, Lausanne) on a proposal from:

Prof. Dr. Jean Hennebert, Master's Thesis Supervisor

Place, date: _____

Prof. Dr. Jean Hennebert	M. Philippe Joye
Supervisor	ICT MRU Leader at HES-SO//Fribourg

Dedicate

To my family that believed in me, and still I don't know why.

Contents

Contents

Acknowledgments

I wish I could thank an AGI for doing my thesis.

Name and thank my proof readers. To my professor Jean for the opportunity to work on a subject that really meaningful to me. To my colleagues that supported me and helped me stay sane during this project.

Glossary

Adversarial Learning

In Machine Learning (ML), the concept of this technique relies on trying to fool models via malicious inputs. Interpretable as a game a model plays with itself, in which the model modifies the input in such a way that it will recognize it as another input, and then learn from its mistake.

Attention Mechanism

In Natural Language Processing (NLP), the Attention Mechanism is an algorithm used to calculate the relational weight between elements in a sequence of elements (most often words).

Bidirectional Encoder Representations from Transformers

In Natural Language Processing (NLP), *Google* BERT is a large Transformer-based model trained at predicting masked tokens within sequences.

Bidirectional Language Model

In Natural Language Processing (NLP), a Bidirectional Language Model represents a Language Model (LM) combining the forward pass and the backward pass of the same corpora.

BLEU

In Natural Language Processing (NLP), The Bilingual Evaluation Understudy (BLEU) is an evaluation metrics particularly popular in machine translation as it is an automatic processing.

Close-ended

A closed-ended question is designed to allow a limited amount of responses.

Encoder-Decoder

In Machine Learning (ML), Encoder-Decoder are two Neural Networks (NNs) that work in pair. The Encoder generates a fixed-size output vector from any sized vector input, and the Decoder generates from the Encoder output a vector that could be any size.

Few-Shot Learning

In Machine Learning (ML), Few-Shot Learning is a technique used to solve tasks with a very small amount of training data.

Glossary

Generative

In the context of the Thesis, we are using the generic word “Generative” as the concept of an algorithm able to generate outputs in a meaningful but unpredictable manner from an input, which includes Language Model (LM)s and Generative Models.

Generative Model

In Machine Learning (ML), Generative Models are generating random outputs from a single input by using the probability of observing the output based on the input. In other words, it models the probability of observation for a given target.

Generative Pre-Training 2

In Natural Language Processing (NLP), *Open-AI* GPT-2 is a large Generative Model using Transformers to generate outputs based on the probability of the token observation.

Ground Learning

In the context of the Artificial Intelligence (AI), Grounded Learning is based on the Grounded theory from the social sciences, which uses inductive reasoning. The mechanism combines structured and unstructured data as small conceptual parts to then apply machine reasoning inductively.

Hop

In Question Answering (QA) Systems, a Hop is a quantitative measure of the number of supporting facts or combinations necessary between indirectly related pieces of information to provide an answer.

Knowledge Base

In Information Systems (IS), a Knowledge Base is a Knowledge Representation using a Linked Data database for storing and interlinking structured and unstructured data using a standard.

Knowledge Graph

In Information Systems (IS), a Knowledge Graph is a Knowledge Base (KB) organized as a graph using semantics.

Language Model

In Natural Language Processing (NLP), a Language Model is a Model trained to provide likelihood probabilities of the following sequence of words in addition at providing the probability for each sequences of words.

Linked Data

In Information Systems (IS), Linked Data is a structured interlinked database mainly used for semantic queries.

Machine Reasoning

In Machine Learning (ML), Machine Reasoning represent the ability to apply reasoning for a given input by using knowledge representations and logic patterns such as inductions, analogies, or abductions.

Machine Understanding

In Machine Learning (ML), Machine Understanding stays ambiguous in its definition. However, we use the term as the ability to represent knowledge as atomic building blocks and fundamental relations.

Markov Decision Process

In the context of the Reinforcement Learning (RL), this process models the ability to predict the next state of a finite-state machine-like process, such as a game, with only the information contained in the present state.

Mean Reciprocal Rank

In Information Retrieval (IR), the Mean Reciprocal Rank provides a statistic measure of the quality of a returned ranked list of items for a query. MRR takes only in account the highest-ranked relevant item to the query.

Model

In Machine Learning (ML), a model is the representation of the assumptions made by the algorithm during the training phase. Models are used to output a result based on a provided input and the learned patterns.

Model Fine-Tuning

In Machine Learning (ML), Fine-Tuning a Model is the technique of using a trained Neural Network (NN) model as a base and tune it for a specific task.

Multi-Hop

In Question Answering (QA) Systems, a Multi-Hop implies that the answer is within multiple Hop of the question. In other words, the answer requires a combination of different information to be answerable. Generally, extra qualifying Subject-Predicate-Object Tuples (SPOs) are separating the question Subject and the answer Object.

Named-Entity Linking

In Natural Language Processing (NLP), Name-Entity Linking extends the Named-Entity Recognition by providing an unique identifier to each word allowing a mapping in various databases (useful in translations).

Named-Entity Recognition

In Information Extraction (IE), Named-Entity Recognition is a technique used to extract, from unstructured texts, words from a predefined vocabulary.

Open Domain

In Information Retrieval (IR), the support of Open Domain questions provides no restriction to the questions' subject.

Oracle

In Machine Learning (ML), an Oracle is defined by an entity who knows that knows the ground truth to all questions. It can be a human, or an algorithm querying perfectly a database.

Part of Speech

In Natural Language Processing (NLP), Part of Speech is a technique used to categorize words that behave syntactically similarly.

Part of Speech Tagging

In Natural Language Processing (NLP), The Part of Speech Tagging is extending the Part of Speech by adding a label to the word depending on its context (the neighboring words).

Reinforcement Learning

In Machine Learning (ML), this type of learning combines generally a Markov Decision Process (MDP) environment with an approach similar to Unsupervised Learning (UL) as it does not require labelled data. The particularity of this technique is that it uses a notion of rewards to predict the best next-step by running a large amount of simulations as training.

Sequence-to-Sequence

In Machine Learning (ML), a Sequence-to-Sequence or Seq2Seq is an Encoder-Decoder Neural Network (NN) that, for a given sequence of elements as input, outputs another sequence of elements.

Shallow Neural Network

In Machine Learning (ML), similar to Deep Learning (DL), Shallow Neural Networks have a Encoder-Decoder approach by having a single hidden layer, which often has a high amount of parameters.

Single-Hop

In Question Answering (QA) Systems, a Single-Hop implies that the answer is within a single Hop of the question. Generally, a unique Predicate separates the question Subject and the answer Object.

Supervised Learning

In Machine Learning (ML), this type of learning implies the uses of labelled datasets to perform the training.

Transformer

In Natural Language Processing (NLP), Transformers are similar to Sequence-to-Sequence (Seq2Seq) architectures but are using a parallelized Attention Mechanism.

Unsupervised Learning

In Machine Learning (ML), this type of learning implies the use of unlabelled datasets to perform the training.

Wikidata

Wikidata is a community-based Knowledge Base (KB), based on Freebase originally. It stores its data into a linked-data format with Subject-Predicate-Object Tuples (SPOs).

Word Embedding

In Natural Language Processing (NLP), the Word Embedding is a technique for word representation as vectors in an embedding matrix. Additionally, it has often the particularity of preserving the semantical analogies of word-vectors.

Zero-Shot Learning

In Machine Learning (ML), Zero-Shot Learning is a technique used to solve tasks without training on examples.

Acronyms

AGI

Artificial General Intelligence.

AI

Artificial Intelligence.

AIML

Artificial Intelligence Markup Language.

AL

Adversarial Learning.

ANI

Artificial Narrow Intelligence.

ANN

Artificial Neural Networks.

AT

Adversarial Training.

BERT

Bidirectional Encoder Representations from Transformers.

biLM

Bidirectional Language Model.

CE

Character Embedding.

CNN

Convolutional Neural Network.

CoQa

Conversational Question Answering.

CWE

Context-based Word Embedding.

Acronyms

DL

Deep Learning.

DNN

Deep Neural Networks.

FAQ

Frequently Asked Questions.

GAN

Generative Adversarial Networks.

GLUE

General Language Understanding Evaluation.

GPT-2

Generative Pre-Training 2.

GS

Generative System.

ICT

Information and Communications Technologies.

IE

Information Extraction.

IR

Information Retrieval.

IS

Information Systems.

KB

Knowledge Base.

KD

Knowledge Distillation.

KG

Knowledge Graph.

LM

Language Model.

LSTM

Long Short-Term Memory.

MDP
Markov Decision Process.

ML
Machine Learning.

MN
Memory Network.

MR
Machine Reasonsing.

MRR
Mean Reciprocal Rank.

MRU
Master Research Units.

MT
Master's Thesis.

MU
Machine Understanding.

NL
Natural Language.

NLG
Natural Language Generation.

NLP
Natural Language Processing.

NLU
Natural Language Understanding.

NN
Neural Network.

OOV
Out-of-Vocabulary.

POC
Proof of Concept.

QA
Question Answering.

Acronyms

QuAC
Question Answering in Context.

RL
Reinforcement Learning.

RNN
Recurrent Neural Network.

Seq2Seq
Sequence-to-Sequence.

SL
Supervised Learning.

SNN
Shallow Neural Network.

SOTA
State of the Art.

SPO
Subject-Predicate-Object Tuple.

SQuAD
Stanford Question Answering Dataset.

UL
Unsupervised Learning.

Weak AI
Weak Artificial Intelligence.

Abstract

We propose an innovative approach for question-answering chatbots to handle conversational contexts and generate natural language sentences as answers. In addition to the ability to answer open-domain questions, our zero-shot learning approach, which uses a pure algorithmic orchestration in a grounded learning manner, provides a modular architecture to swap statically or dynamically task-oriented models while preserving its independence to training.

In the scope of this research, we realize the proof-of-concept of an Open-domain and Closed-ended Question-Answering chatbot able to output comprehensive Natural Language generated sentences using the Wikidata Knowledge Base.

To achieve the concept, we explore the extraction, and the use of sub-knowledge graphs from the Wikidata knowledge base to answer questions conversationally and to use the sub-graphs as context holder. Additionally, we are extracting Subject-Predicate-Object tuples from the graph and using Language Models to join the SPOs and extend the answers as natural language sentences.

The proof-of-concept architecture uses a combination of state-of-the-art and industry-used models with a fine-tuning strategy. As a motivational target, we use a Zero-Shot Learning approach, by combining various models with an algorithmic orchestrator and using pure algorithmic for the graph manipulation and answer extraction.

Finally, we evaluate the answers and compare the results with state-of-the-art Single-Hop and Multi-Hop question-answering systems on question-answering datasets. We find out that, aside from the computation time and the computational resources needed, our proof-of-concept performs similarly at question-answering compared to its competitors.

Keywords: Machine Learning (ML), Natural Language Processing (NLP), Single-Hop, Multi-Hop, Question Answering (QA), Wikidata, Wikipedia, Knowledge Graph (KG), Knowledge Base (KB), Word Embedding, Part of Speech Tagging, Named-Entity Recognition, Named-Entity Linking, Language Model (LM), Model Fine-Tuning, Graphs, Sub-Knowledge Graphs, Transformer, Bidirectional Encoder Representations from Transformers (BERT), Generative Pre-Training 2 (GPT-2), Information Extraction (IE), Spacy, GloVe, DeepCorrect, Chatbot, Conversational, Information Retrieval (IR), Queries, Python

How to read this document

To be completed at the end of the work

Describing the structure of the document with a redline and its reasoning.

Project preface

Introducing the project

State-of-the-art

In this part, we will be exploring the state of the art of NLP technologies as it is at the beginning of 2020.

Design and realization

Explaining how we got to build a proof of concept, what happened during the process of the initial plan, and how when came up with an innovative solution while solving and starting the design project from scratch.

Retrospective

The results are here; it's awesome what we accomplished!

Part I

Project preface

Chapter 1

Introduction

New technologies are revolutionizing the way humans access knowledge as a service from multiple platforms and providers. Thanks to the emergence of increasingly powerful AI algorithms, particularly in the field of NLP, conversational agents, commonly named chatbots, have come a long way and have become popular among information consumers. As it is in early 2020, chatbots are all still ANIs[1]. Even if the chatbots are continually improving at providing the best outputs for specific tasks as well as providing meaningful human-like sentences, they still cannot generalize the tasks toward human-like conversations. The task of conversation, as humans are applying it, a complex integration of tasks including understanding, reasoning, context linking, context tracking, curiosity, initiatives, Few-Shot Learning or Zero-Shot Learning and learning on the fly, have yet to be accomplished. Nonetheless, as research progresses, chatbots are improving with new technics and tools that are making them step by step closer to complete human-like discussions, slowly progressing towards AGI chatbots. As for the scope of the thesis, we are humbly focusing on the combination of few NLP tasks with a Zero-Shot Learning approach to help Machine Learning (ML) and NLP research getting closer to General QA Conversational Chatbots.

1.1 Aim of the Research

The initial goal of the thesis was to explore and combine State of the Art (SOTA) QA Systems and Language Models (LMs) to into an experimental POC of a Conversational QA Chatbots.

During our research journey, we discovered a new purpose to the project, and took a step into the unknown with a Zero-Shot Learning approach with sub-knowledge graphs.

1.1.1 Project's Overall Scope

We are focusing on the English language as an attempt to increase the number of compatible datasets and make community accessible solutions. We are exploring and combining two types of systems as an attempt to build QA chatbots. The first system will produce factual answers, and the second system will generate human-like sentences from the answers found by the primary system. For the factual answers, we will be evaluating the results of our combined system against SOTA QA

[1] The State of AI Report 2019 (Benaich et al., 2019)

systems on QA testing datasets. Humans will manually evaluate the answered sentences from our combined system. Finally, as the time allocated for the thesis is 19 weeks, the outcomes are narrowed at providing non-exhaustive research and a POC solution. On a side note, the review of the risks and ethical problems that could be raised by the development of such solutions are not part of this work.

1.1.2 Industrial Interest

iCoSys, the Institut of Complex Systems at the University of Applied Sciences and Arts at Fribourg, Switzerland, is interested in the results of this study for their *AI-News* project[2]. Its goal is to provide a chatbot-based system as a tool for press readers, to help them narrow their interests and deliver the right information. This project is in collaboration with the *Swiss Innovation Agency* from the Swiss Confederation, *La Liberté*, the daily newspaper from Fribourg and *Djebots*, a startup selling scenario-based narrow chatbots.

1.1.3 Personal Interest

In harmony with the thesis subject, as the author is particularly interested in exploring the premises to AGI related technologies such as Zero-Shot Learning, Ground Learning, Machine Understanding, and Machine Reasoning for a Multi-Domain Task Generalization. The human-like QA frame of this project is particularly motivational.

1.2 Research Questions

We articulate here the initial set of questions as a driver to our research work. From these questions are declined objectives, and from objectives are declined milestones framing the plan.

- What are the components to make QA chatbots?
  - What is the SOTA of chatbots and QA systems?
  - How to tune QA chatbots to make them as human-like as possible?
  - How to tune such systems for the field of journalism?
- What is the state of the art for Generative QA chatbots?
  - What are the components to make Generative QA chatbots?
  - Are Generative chatbots only as good as the data they consume?
  - Could Generative chatbots be a step toward AGI?

[2] `AINews.ch`

Part II

State-of-the-art

Chapter 2

Chatbots

Based on latest MMC's state of AI report[1], it appears that 26% of the AI-Startups studied by Gartner[2] are using or making chatbots (see Figure 2.1). The same study, made a year earlier, in 2018, shows that chatbots are not present as an application, which implies that either chatbots were not referenced as AI or that their popularity exploded within a year.

As it is at the beginning of 2020, based on The State of AI Report 2019 (Benaich et al., 2019) and the two previously mentioned studies, chatbots are commonly present but limited to narrow tasks. In most cases, they are scenario-based with sequences of if-else conditions that we classify as non-learning AI. Moreover, hard-coded scenarios are requiring an infinite amount of human power to create generic Chatbots able to maintain a conversation at a human level. However, progress in the field of ML and NLP is demonstrating that providing large corpora to an unsupervised algorithm is enough to maintain a passive conversation with users, which results into a shifting of the human power into data engineering. Increasingly complex algorithms and techniques are emerging at a monthly in the field, demonstrating a trend towards conversational performance improvements. Note that even if they are getting better at providing meaningful sentences, current Chatbots are still not able to orchestrate the generalization of all the tasks required to a human-like conversation. E.g., such as understanding and reasoning based on the context, initiatives to search and learn for missing information, initiate dialogue in a meaningful manner, intuition, and much more. As a side note, the generalization of those tasks would reduce the steps significantly towards general Chatbots.

From a user-centric point of view, chatbots are currently trending and rising global interest for various reasons. Big companies such as *Google* or *Apple* are believing in the technology and are making a lot of effort at pushing the chatbots into the mainstream. Even if the word "chatbot" is commonly used as a buzzword without a proper definition, people have at least a mental representation of its concept. Indeed, whether they call it "Digital Assistant", "Siri", "ok Google" or "Alexa", they all expect to a more or less human-like conversations after using those triggering keywords.

[1] The State of AI 2019: Divergence (Kelnar, 2019)

[2] 2'791 European AI Startups from the 2019 CIO Survey: CIOs Have Awoken to the Importance of AI (Rowsell-Jones et al., 2019)

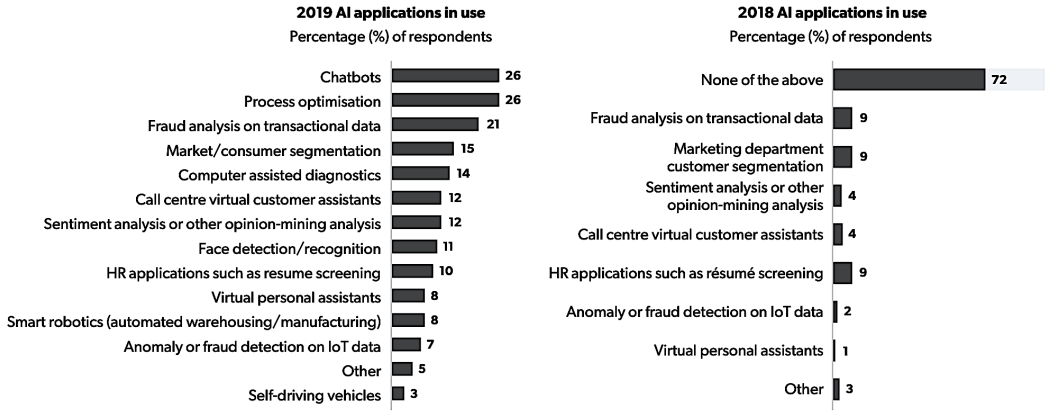


Figure 2.1: Figure 31 from *The State of AI 2019: Divergence (Kelnar, 2019)*. The top AI applications used in European AI Startup in 2019 are Chatbots and Process optimization.

It is interesting to note that the majority of the following sections could be included in the field of AI in general. The extrapolation of the chatbot subject to AI as a whole is worth further studying, but it not part of this work. Instead, the focus of this chapter is Chatbots; we provide a synthesis and classification of the different methods used to build chatbots. We will define the main categories identified and continue on the main sub-categories and conclude with a cartographical chart of our chatbot vision.

2.1 Chatbot History

Not mentioning *Alan Turing* or *Joseph Weizenbaum*, both considered as the fathers of AI and chatbots, would not be fair to this research. Indeed, in 1950 they forecasted human-like communication with computers and proposed a test to differentiate humans from machines, the Turing Test (Turing, 1950). The test performs as follows: a supervisor asks a human to talk to a masked entity and determine rather he is talking to a human or a computer. If the human cannot recognize speaking to a computer, then the machine passes the Turing test.

In 1966,*Joseph Weizenbaum* wrote Eliza(Dunlop, 1999), a computer program simulating a psychotherapist, it is seen today as one of the first well-documented attempts to make a Chatbot designed at passing the Turing test. However, due to techniqueal restrictions, Eliza was not performing particularly well in all contexts. As for today, it is still possible to play with the chatbot on a dedicated website.

Since Eliza, a lot of progress has been made until 2020, From conditional IF-ELSE, Artificial Intelligence Markup Language (AIML), up to ML with Artificial Neural Networks (ANN) and Deep Neural Networks (DNN), the improvements in the field of chatbots increased drastically over the years. Each iterations delivering algorithms being continuously more sophisticated and better at using the Natural

Language (NL), resulting in a new field of ML called NLP. As a reminder of the chatbots history and progress from 1966 to 2016, the infographic(Futurism, 2016) from Futurism is particularly speaking.

2.2 Main Categories in the Chatbot Realm

While performing the state-of-the-art, we identified three main chatbots categories.

2.2.1 Conversational

We like to call them the Chatty bots, and they are great for interaction and structured replies, well designed for their ability to talk. E.g., *User*: “Hello, how are you?”, *Bot*: “Good, what about you?”.

2.2.2 Task-Oriented

The Task-Oriented bots are performing particularly well at specific tasks as smart-assistants. As their design is not toward generalization, their abilities are limited and will fail at off-tasks. A common workflow used by those bots is to detect the Intent and the Entities of the user request, often in NL, then apply a rule-based matching to perform the command intended by the user. E.g., *User*: “Book the next flight to Geneva from Zürich.”, *Bot*: “Alright! Your ticket number is 00XXYYZZ. Have a great flight!”

2.2.3 Dispatcher

The dispatcher acts as a middleware, who’s unique job is to categories the user input and forward the input to the task executor from any of the previous two categories that the user requested. E.g., If the user request the following "What is the weather in Geneva?", the dispatcher will categories the question as the task of providing the weather and sent it to the weather module. As a second example, if the user provides the following input "Hey! Let’s talk about random stuff!", the dispatcher will forward the request to the chatty module.

2.3 Retrieval Chatbots

As it is today, Retrieval-based Chatbots are popular in the industry. Indeed, a lot of tools are available, and they perform well for specific tasks. However, the response capabilities are limited to their databases and the retrieval algorithm used. Indeed, for a given input, the system is using heuristics to find the best output from the pre-defined responses. The choice of the algorithms is wide and depends on the task the chatbot is required to perform. Regardless of the heuristic used, from keywords matching up to Deep Learning (DL), the output will always be retrieved from the database. Concerning the database itself, the data needs a pre-processing step to generate indexes linking the questions, answers, and apply pre-calculated scores. Pre-processing also implies that if the database is updated, a new pre-processing batch is required, which implies that the scalability or fine-tuning if compromised in the long run. We like to call this type of chatbots “Keywords-based”. See Figure 2.2.

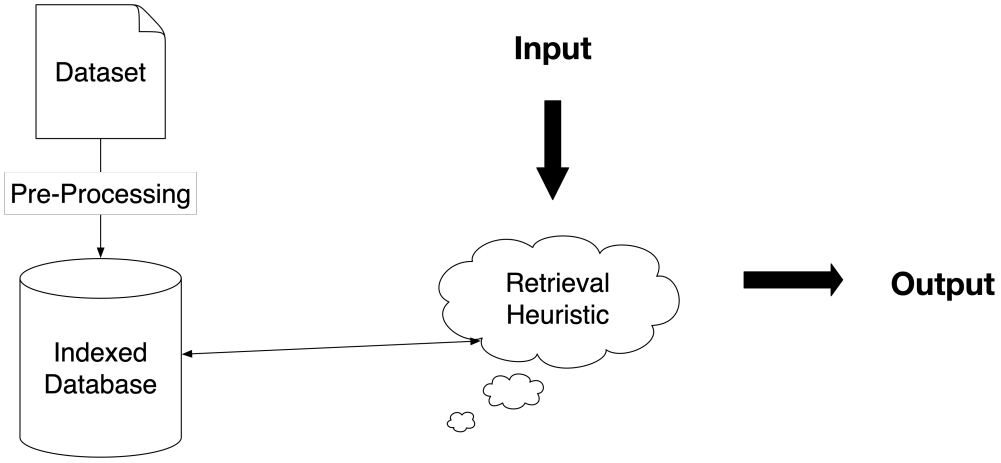


Figure 2.2: Illustrative representation of frequent retrieval chatbots architecture.

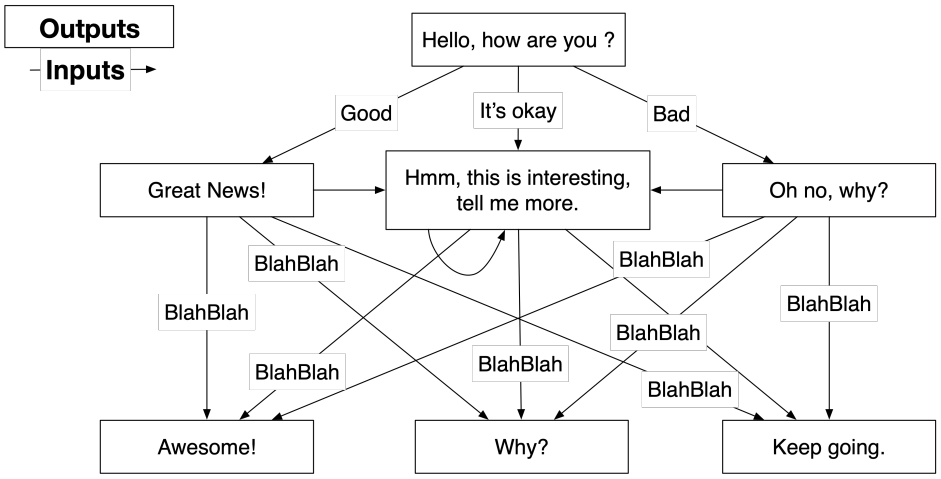


Figure 2.3: Illustrative representation of frequent rule-based chatbots process.

2.4 Rule-Based Chatbots

"Scenario-based", as we name it, is the oldest and relatively straightforward system for chatbots. The ElizaDunlop, 1999 Chatbot, as mentioned in the Chatbot History 2.1, is scanning the input text for keywords, calculates a ranking for each keyword, and finally goes through a series of conditions called rules, and some randomness to reach the best ending leaf. Usually, the bot also includes a default output if the matching process fails, which we can still nowadays see in chatbots: "Hmm, this is interesting, tell me more.". Such bots are often used for interactive chatbots, as it can, in a controlled environment, give a sense of deep meaning in the context of the conversation. Note that such systems require a lot of human power to build a frame for the bot to play in, and by this mean makes rule-based chatbots great for the specific scenario but is hard particularly hard to generalize. See Figure 2.3.

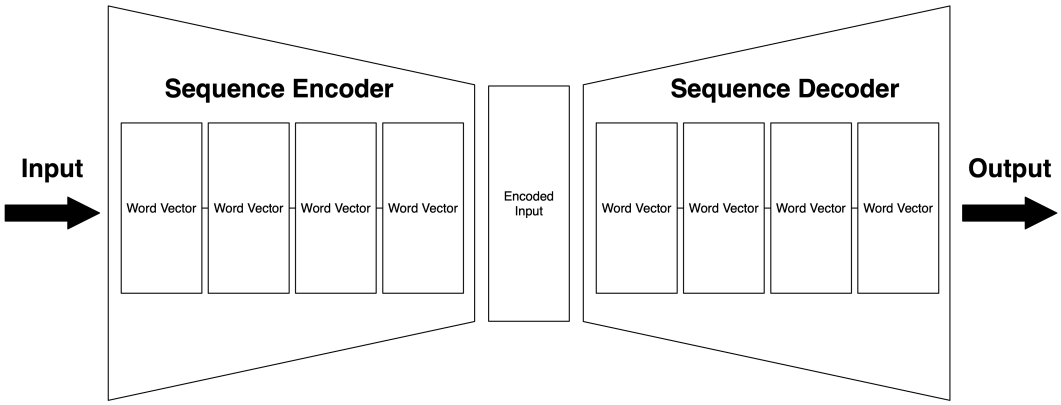


Figure 2.4: Illustrative representation of a Sequence to Sequence architecture.

2.5 Generative Chatbots

As the current result of all the incredible innovations made in the past years in NLP, and is a premise to true conversational chatbots, generative methods are overcoming the limitations of the Retrieval 2.3 and Rule-Based 2.4 Chatbots, by its ability to generate new content. Either Supervised 2.5.1, Unsupervised Unsupervised Learning (UL) or Adversarial 2.5.2, no pre-defined outputs are used, the models are trained on large corpora to learn the language patterns and outputs relatively meaningful responses to give inputs. Another particularity of generative chatbots, is that building a domain-oriented chatbot does not require the engineers to have the domain expertise, as the expertise is embedded into the data, which allows a relative scalability to new domains. However, even if the trained models can output responses at nearly no timespan, the data-engineering of the datasets and the training phase is most often long and complicated. As a final note, the responses generated by such chatbots are only as good as the data it was fed during the training.

2.5.1 Supervised Learning

Supervised Learning (SL) is probably the most common method used by Generative Chatbots, as it provides relative control over training. Sequence-to-Sequence (Seq2Seq) is commonly used as architecture for those chatbots, a NLP version of the Encoder-Decoder, which encodes the input words sequence and decode it into a words sequence as an answer into a framed conversation fashion. The training only requires a dataset containing a sentence and its desired response, the model will then map similar inputs with similar outputs. However, a clear limitation for this learning is that the model will for any input always have an answer, regardless of the overall meaning. Additionally, Seq2Seq will prioritize the highest word apparition probabilities, meaning that data duplicates and requiring sentences will create a trend during decoding. E.g., “I don’t know the answer.”. See Figure 2.4

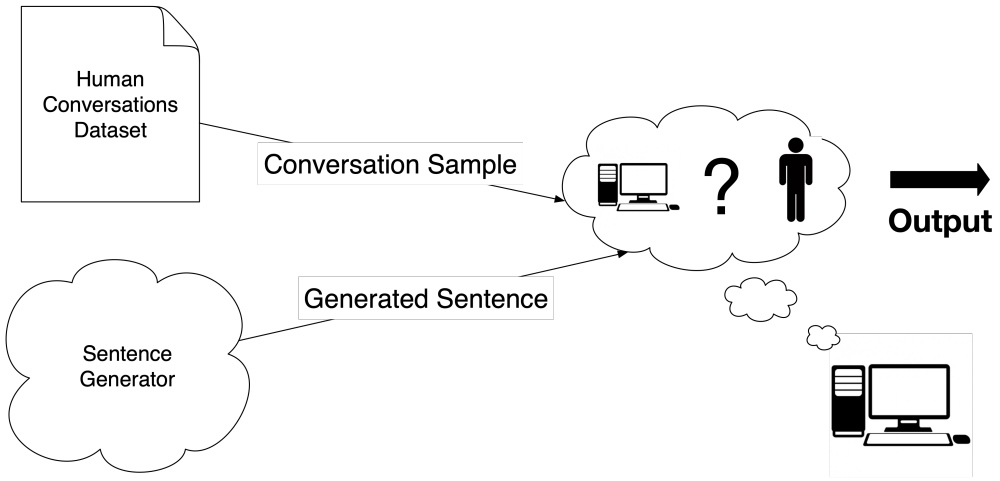


Figure 2.5: Illustrative representation of an adversarial architecture in a chatbot context.

2.5.2 Adversarial Learning

Adversarial Learning (AL) has driven attention thanks to Computer Vision Generative Adversarial Networks (GAN) (Karras et al., 2019) by proving that it is possible to generate realistic human faces (Wang, 2019). In the chatbots context, it can be extrapolated into a futuristic version of the Turing Test 2.1, in which machines are confronting themselves instead of humans. The concept implies the use of a training dataset containing human conversations, and compare them against the generated answer; the discriminator will then judge which is from a human and which is from an algorithm. Note that adversarial methods such as GAN are working well because of the nature of the data it plays with; indeed, pixels can be deeply noised, but words cannot be due to their discrete nature. See Figure 2.5

2.5.3 Pre-trained Language Models

Language Models are currently the most recent and the most promising models due to their ability to model language itself instead of conversations and then tune the outputs as a chatbot would. It can been seen as semi-supervised learning, as it uses UL for training and supervised learning 2.5.1 for fine-tuning 2.5.4. We will dive into LM in the NLP chapter 3.3.

2.5.4 Model Fine-Tuning

With Model Fine-Tuning (see Figure 2.6), LM have, by design, the ability to be enhanced to perform particularly a various NLP task such as chatbots. Because pre-trained Language Models (LMs) are based on the grounded blocks of language itself, implying model post-training customization as a light learning task. Indeed, it is relatively easy to fine-tune a QA dataset to a LM, making the model able to answer questions instead of descriptively filling sentences. The main downside to those models is the large memory size required to run them. However, due to their nature, they are trained once and then fine-tuned. Note that training requires an enormous amount of computational power. E.g, The largest form of BERT (Devlin

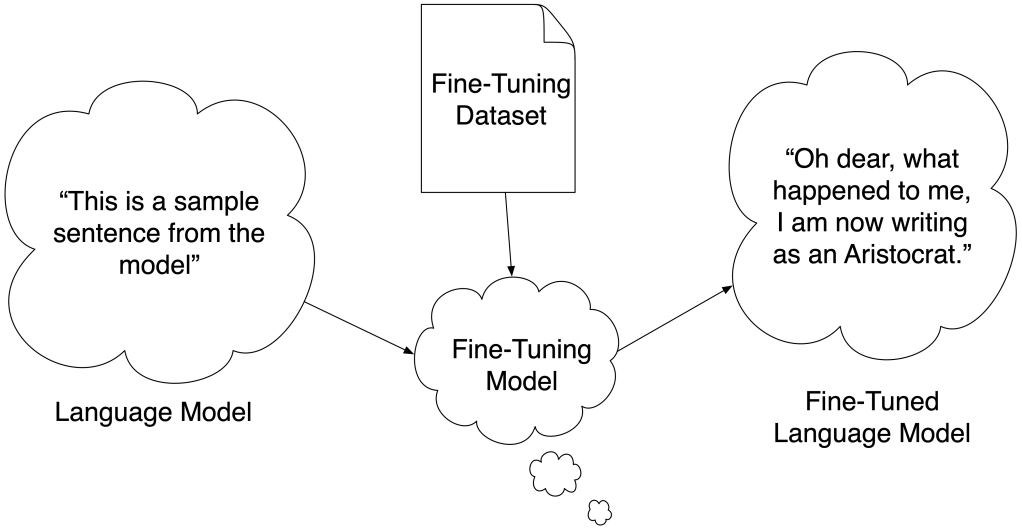


Figure 2.6: Illustrative representation of fine-tuning in a chatbot context.

et al., 2019) was trained on 16 TPUs for 4 days. Fine-tuning, on the other hand, scales down to few hours on a single TPU, which makes it relatively scalable to new domains.

2.5.5 Reinforcement Learning

Reinforcement Learning (RL) is proven to be very powerful by the latest research made by *Open-AI* with its DOTA2 bot or *Google's Deepmind* with AlphaZero, so we believe that it is worth mentioning it. However, this type of learning requires a finite state similar to a Markov Decision Process (MDP), which matches game cases but not conversations, and impacting by this means the motivation to export the technique to NLP. Indeed, this methodology requires that all information required for the next step are wrapped into a single state to predict it, which makes it hard to use the dialogue case. For now, NLP research does not provide a conclusion as, even with billions of simulations, RL Chatbots could reach comparative results to Generative Chatbots 2.5.

2.6 Grounded Chatbots

Falling in a particularly rare research field of ML and NLP, Ground Learning can be seen as the future of Machine Understanding (MU) and Machine Reasoning (MR). In a chatbot context, the goal is to simulate, based on the Grounded Theory from the social sciences, how humans are using inductive reasoning to create conversations with unstructured knowledge. The idea is to give the ability to the bot, for any given input, to gather information from any data sources and provide an inductive output. E.g., Combining Knowledge Bases with weather forecaster. As a second example, for the given input: "What is the color in autumn of a leaf in Switzerland?", 1) the bot would have first to identify the context keywords (color, leaf, autumn, switzerland), 2) the bot would select where to gather the information, 3) the would investigate the Wikidata Knowledge Base, Wikipedia, and The Weather Channel API, 4) the bot would formulate an answer based on the information it gathered. 2.7

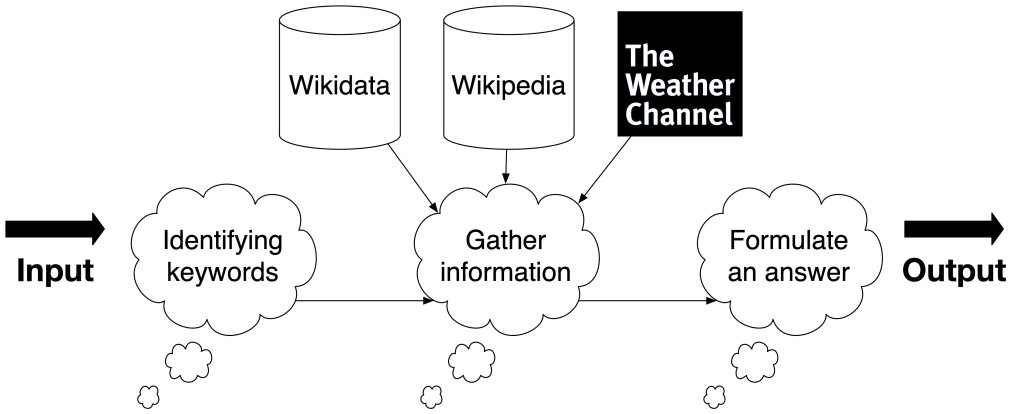


Figure 2.7: Illustrative representation of a grounded chatbot.

2.7 Question-Answering Chatbots

QA is a prevalent task for chatbots; indeed, they are widely used for questioning tasks in either Single or Open Domain, Open or Close-ended, Single or Multi-Hop with applications such as FAQs, Supports, help to find the meaning of life, and so on. Due to the broadness of the field, no defined methodology has been generalized; instead, it uses either one or multiple techniques described in the previous sections. It is interesting to note that the field of QA is raising a lot of interest in NLP research lately, and the benchmarking game of creating the new baselines, with increasingly complex datasets, is still in progress. In this section, we will overview some recent baselines.

Fine-Tunning Language Models Large LM such as BERT (Devlin et al., 2019) or GPT-2 (Radford et al., 2018) are often fine-tuned on QA datasets similar to SQUAD 2.0 (Rajpurkar et al., 2018a) which are particularly tricky, even for humans.

Querying Models Based on QA datasets, a model is trained to fill structured templates. The generated output is a structured query for a particular querying language such as SPARQL for Wikidata.

Retrieval A popular approach in the industry is to use tools such as Elasticsearch for indexing and additional tools using ML heuristics to perform the queries.

2.8 Common Chatbot Features Overview

In this section, we are non-exhaustively naming a few recurring features appearing during our targeted research.

2.8.1 Context

Humans are intuitively and extensively relying on the context for conversational purposes, which implies similar capacities from dialogue-based chatbots. In the scope of the Thesis, we are also using the term Multi-Turns Conversations to mention context holding implicitly. On a side note, one-way style dialogues such as commands or none-nested questions do not need to keep context to perform well.

Short term context Implying the ability for the bot to hold context for at least the current conversation, e.g., few keywords or on-the-fly Model Fine-Tuning.

Long term context Often, chatbots would use user-profiles as part of their architecture to remember information such as the favorite pizza flavor of a client.

2.8.2 Proactivity

Simulating personalized interest as a human would do is not new to chatbots, as it has been proven by becoming a standard in marketing and customer support chatbots. Messages such as “Hey, you have been on our web store for a while, can I help you?”, are carrying a sense of proactivity; however, beyond asking general pre-made questions, limitations are clear, and not much progress has been made yet in the field. Indeed, human-like proactive chatbots imply algorithms capable of initiating conversations by initiating a dialogue or asking information in a meaningful manner based on the long and short term context.

2.8.3 Narrow vs General Chatbots Scope

Beyond the three main categories 2.2 identified during the study, in general, chatbots can additionally be classified within a scope starting at Narrow Chatbots up to General Chatbots. To position them, we defined a two axes classification using Tasks and Knowledge as represented on Table 2.1.

Tasks Axis To name a few examples of task-oriented Chatbots: Talk, Frequently Asked Questions (FAQ), Customer Support, or Ordering.

Knowledge Axis Non-exhaustively, as follows, a few knowledge-centric examples for chatbots: Health, Weather, or Customer Service.

Narrow Chatbots Narrow chatbots are limited by the range of tasks they can accomplish and the knowledge they can use. By design, they are very good at a particular task for a particular knowledge requirement.

General Chatbots They are neither limited by the range of tasks they can accomplish nor by the knowledge they can use. However, they often have an average performance for any task or knowledge. We go in more details at section 2.8.4.

Tasks	Knowledge	
	Expert in a specific Field Expert at all Tasks	**General Chatbots** Expert in all Fields Expert at all Tasks
	Narrow Chatbots Expert in a specific Field Expert at specific Task	Expert in all Fields Expert at specific Task

Table 2.1: This table represents categories in Narrow and General Chatbots in a Tasks versus Knowledge format.

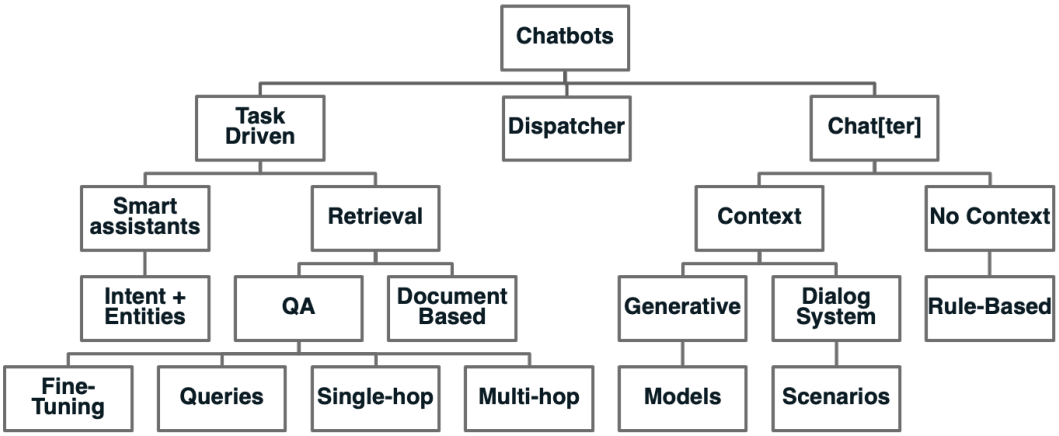


Figure 2.8: Represents the chatbots cartography as conclusion to the chatbot state-of-the-art chapter.

2.8.4 General Chatbots

As research progresses in the NLP field, chatbots are improving as an effort to perform simultaneously well in various tasks and multi-knowledge bases. As a contemporary goal, in addition to any chatbot related tasks and broad knowledge expertise, General Chatbots must not be limited to their current capabilities, but on the contrary, be able to learn new tasks and subjects continuously. As far as we as this study went, we could not find SOTA general chatbots as defined. However, companies like *Amazon* are selling to a large public a feel to general chatbots with Alexa. Indeed, apart from ordering goodies from *Amazon* and roughly conversing with Alexa, users can command their smart homes, use it as a personal assistant, or even program *skills* to perform custom actions.

2.9 Chatbots Cartography

As a result of this chapter, we created a chart on Figure 2.8 representing the current state of chatbots from our point of view. Note that a particular use-case could be in multiple leafs.

Chapter 3

Natural Language Processing

It is often challenging to realize the complexity behind Natural Language (NL), even to experts. First of all, Language is an academic field of study, implying multi-disciplinary skills. And secondly, staying up to date with evergrowing tools and new SOTA algorithms proves to be challenging. NL is the fundamental communication element for humans, NLP is the field of ML studying NL with the goal of providing the ability to machines to handle and mimic NL to create human-like verbal interactions. Beyond words and grammar rules, NL is a complex orchestration of subtleties, intuitively handled by humans, but not easily handled by machines. Nonetheless, NLPs technologies are massively used in our daily lives, including information extraction, summarization, and conversation simulation. However, even if machines are given the same language rules as humans, they do not yet understand the manipulation they are processing, as humans would do. Indeed, NLP algorithms are applying pre-defined or multiple examples-based learned rules, which may result in ambiguities while applying NL. Using a rule-based approach (as seen in Chapter 2.4) to build a NL model would result into near to infinite amount of conditions, this is the main reason for NLP to be particularly present ML, particularly in DL.

3.1 Word Embeddings

The technique is commonly used as the first data pre-processing for DL in NLP tasks. Those Unsupervised Learning (UL) algorithms capture syntactical and semantical words representation from large unlabelled corpora datasets as vectors by building a multi-dimensional matrix. On average, dimensions are held in a scope of 100 to 400, and thanks to the vectorized nature of captured words, geometrical operations can be applied, such as the cosine functions to calculate word similarities. Another feature related to word embeddings is the ability to apply analogical operations such as *'king' - 'man' + 'woman' = 'queen'*, which popularize Word2Vec 3.1.1 and gave credits to the method, even if the justification to this effect has been theorietized 4 years later [1] by stating that the compositionality is only seen when assumptions are held, in particular when words are uniformly distributed in the embedding space.

[1] Skip-Gram - Zipf + Uniform = Vector Additivity (Gittens et al., 2017)

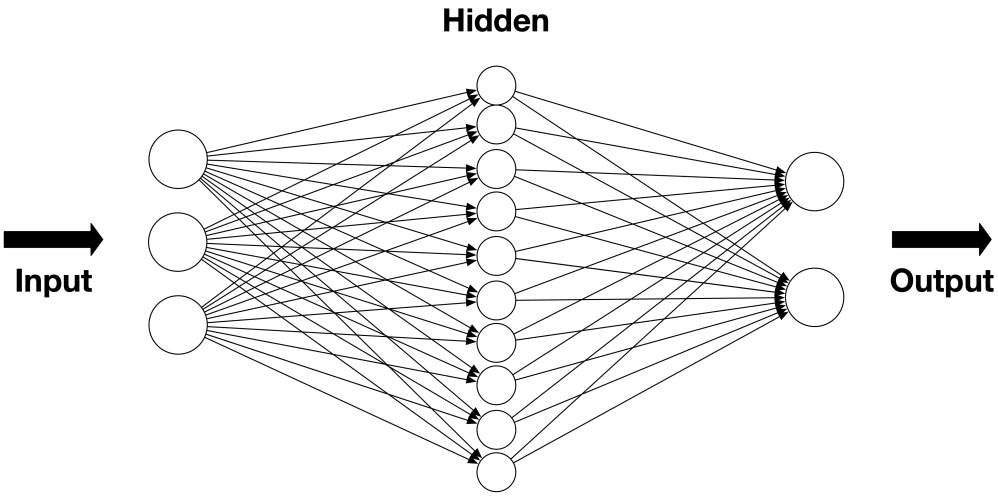


Figure 3.1: Illustrative representation of a Shallow Neural Network

3.1.1 Word2Vec and GloVe

Published by *Google* in 2013, Word2Vec (Mikolov et al., 2013), and its competitor GloVe (Pennington et al., 2014) published by the *University of Standford* in 2014, both use a Shallow Neural Network (SNN), as illustrated on Figure 3.1, similarly to SL by feeding as input a text corpora, and outputting word vectors with a given vocabulary. Training and testing is straightforward but painful tweaking makes it hard to build good generalized word embedding representations. Even if the SNN could remind a DL approach, it is only has one hidden layer; however, the output word vectors are particularly useful for DNN as input.

3.1.2 Out of Vocabulary Problem

A common issue in Word Embedding is related to the vocabulary itself when words are unknown, called the Out-of-Vocabulary (OOV) issue. The issue occurs when post-training the model is requested to provide a vector representation that it has never seen before. A solution could be to handle the exception by forwarding it to a default or pre-defined error vectors such as a series of zeros. We could approach the problem sophisticatedly, by defining on-the-fly OOV words with at a high learning rate as the sum of word-vectors contextualizing the OOV (Herbelot et al., 2017). Another solution would be to fallback to 3.2 by either training a model to compositional map characters to words (Pinter et al., 2017), or using Character Embedding (CE) as a whole instead of Word Embedding 3.2.

3.2 Character Embeddings

Additionally to Word Embedding similar abilities to capture semantics and syntactic relations, CE handles by design OOV issues 3.1.2, which is common for rich vocabularies languages. Instead of using words as vocabulary, CE uses individual characters and semantics embeds words using the characters compositionally, which avoids word segmentation and makes it useful for language such as Chinese (Chen et al., 2015). Moreover, CE can also perform complementary NLP tasks

such as Part of Speech Tagging (Santos et al., 2014), Named-Entity Recognition (Ma et al., 2016), Sentiment Analysis (Hao et al., 2017) and LM (Kim et al., 2015). As it is at the time of writing, *FastText* based on the a morphologically-rich skip-gram approach (Bojanowski et al., 2016) has been popularilized due to its ability to be scalabely trained on large corpora fast, and effectively.

3.3 Language Models

Beyond complex semantics and syntaxes provided by Word Embedding 3.1 and CE 3.2, Language Models (LMs) handles Context-based Word Embedding (CWE) by additionally capturing the polysemy across multiple contexts. Indeed, it was discovered that a distributed semantic, such as Word Embedding and CE are not sufficient to infer context within the embeddings (Lucy et al., 2017). A solution is to combine overall word representations from Word Embedding with *ELMo* (Peters et al., 2018), as its authors suggest, a Bidirectional Language Model (biLM) able to build deep contextual word embeddings by handling multiple word representations. As mentioned in the study, handling polymesy is just one of the Language Models (LMs) features as they are theoritized to capture meaningful NL traits used in NLU and Natural Language Generation (NLG). To increase the LM quality, defined by language syntactic and semantical complexities captured, UL on large corpora is popularly used, as no labeled data is required.

3.4 Transformers

The year 2017 has set a milestone in NLP, transformers (Vaswani et al., 2017) are since then defining the SOTA for multiple NLP tasks mainly due to its parallelized attention 3.4.1 architecture. Large multi-directional pre-trained LM such as Generative Pre-Training 2 (GPT-2) or the Bidirectional Encoder Representations from Transformers (BERT) family are, additionally to their ability to capture features at sentence level, out-performing by a large margin previously mentioned NLP technics at tasks such as QA by performing Model Fine-Tuning, an adaptation of the very popular Transfer Learning feature from computer vision. Making those new LM currently trendy among NLP researchers and engineers.

3.4.1 Attention Mechanism

Introduced in 2014, The Attention Mechanism (Bahdanau et al., 2014) solved the problem raised by tasks such as text summarization, machine translation, or sentiment analysis, where the input is often too rich to perform a selective encoding. Originally, the last hidden state of the decoder is used by a multi-layer perceptron to define the attention from an input hidden state. The mechanism even got adapted from NLP to Computer Vision and shown its ability to replace Convolutional Neural Network (CNN) with SOTA results (Ramachandran et al., 2019).

3.4.2 The architecture

Even if Transformers, Figure 3.2, are using a Seq2Seq approach similar to Encoder-Decoder, which reminds of Recurrent Neural Network (RNN) and CNN, the overall architecture focuses on the attention mechanism to capture the relation between

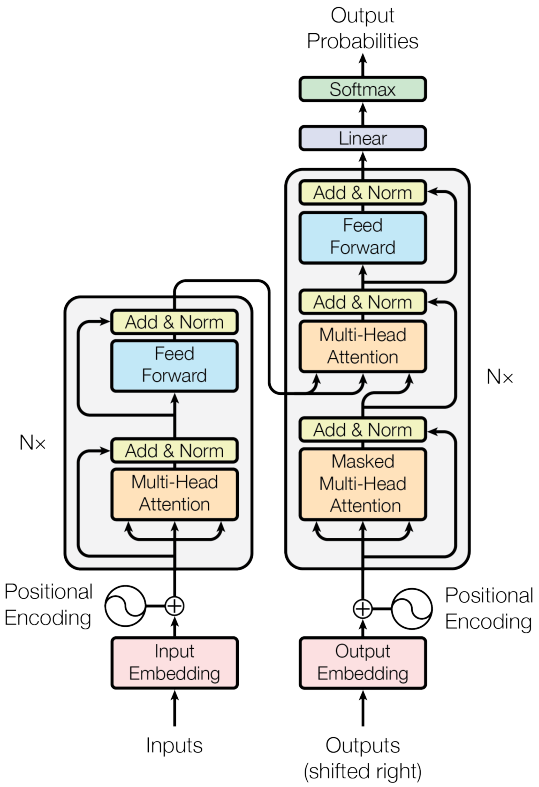


Figure 3.2: Represents the Transformer architecture. Figure 1 from (Vaswani et al., 2017)

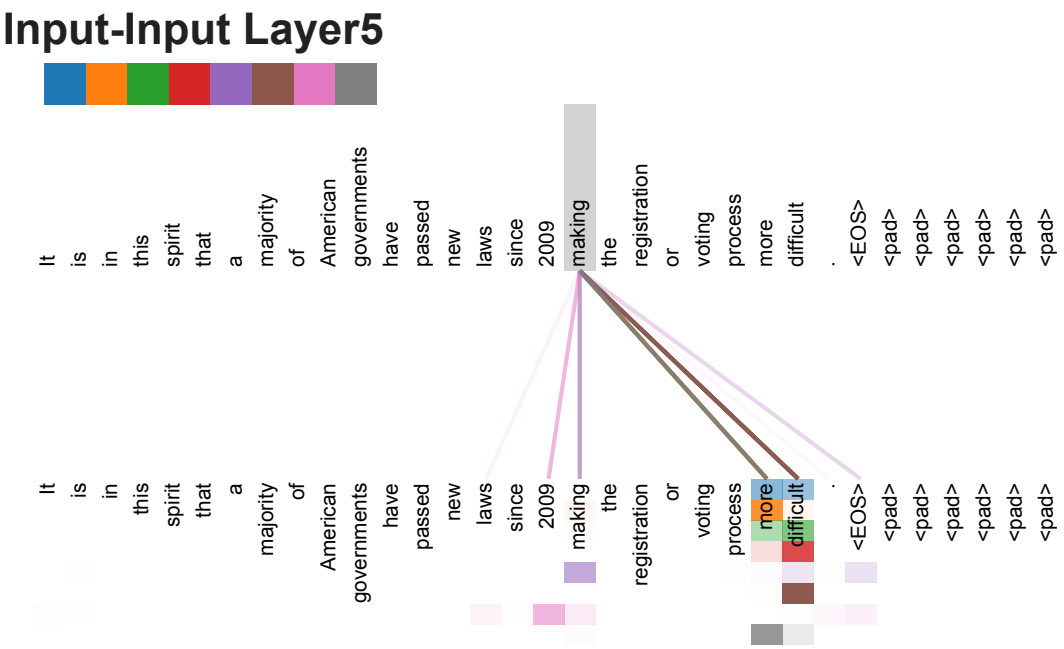
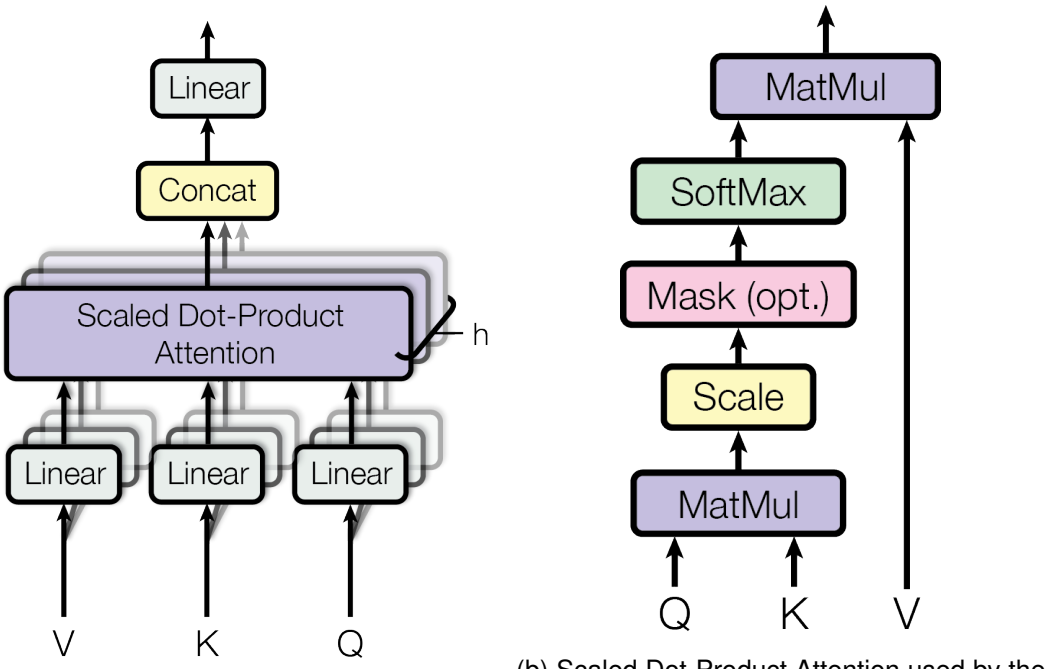


Figure 3.3: Illustrates the attention mechanism for long-distance dependencies handled via multiple attention heads used in transformers. Figure 3 from (Vaswani et al., 2017)



(a) Multi-Head Attention mechanism.

(b) Scaled Dot-Product Attention used by the Multi-Heads Attention 3.4a

Figure 3.4: Multi-head attention anatomy extracted from Figure 2 of *Attention is All you Need* (Vaswani et al., 2017)

the input and the output, making it well parallelizable and less time consuming during training with its multi-attention heads approach. Multi-heads, Figure 3.4a, uses sets of queries Q, keys K and values V to perform attention with dot-products, Figure 3.4b. In other words, the multi-head attention mechanism builds a multi-dimensional matrix representing each word vectors the attention relatives to all word vectors in a predefined window, such as a sentence, then computes the overall attention for each word vectors. In addition to the attention centric mechanism, transformers are also using proven DL techniques such as layer normalization, dropouts, and positional encodings.

3.5 Honorable Mentions

Even if Transformers have deprecated CNN and RNN in NLP by solving their main bottleneck implying the sequential processing during encoding with the Attention Mechanism 3.4.1. We still wanted to mention them as those techniques have defined baselines at multiple NLP tasks for many years.

3.5.1 Convolutional Neural Networks

Commonly used in sentence modeling thanks to their good ability at mining semantics; however, their models are relatively heavy for the task performed. Additionally, they do not perform well on large windows, resulting in bad context handling for long-distance spread information and order tracking. In the field of QA, interesting approach as been researched, such as Multi-Column CNN (Dong et al., 2015) able to treat multiple aspects of questions by building compatible representations

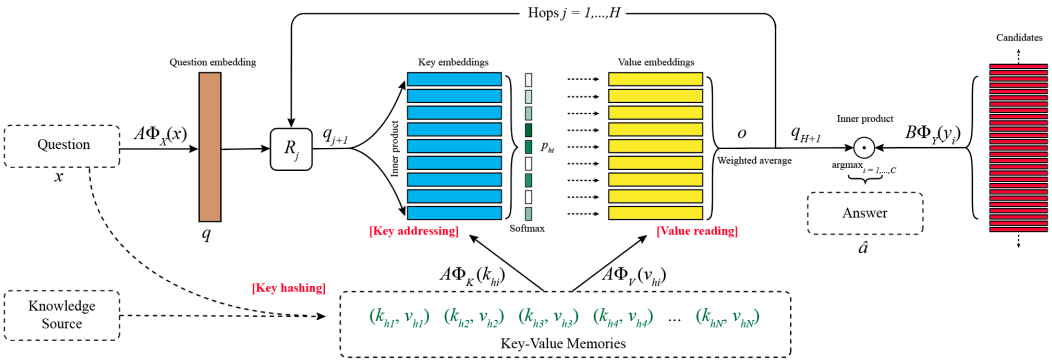


Figure 3.5: Illustrates a Key-Value Memory Network model used in QA. Figure 1 from (Miller et al., 2016)

with Wikidata's ancestor *Freebase* (Bollacker et al., 2008). In 2016, one of the final promising CNN approach was introduced for QA with a model able to handle relational information by word matching question and answer pairs (Severyn et al., 2016).

3.5.2 Recurrent Neural Networks

By design and compared to CNN, RNNs try to take advantage of their ability to remember previous computations. However, it appears that no clear performance winner at NLP tasks demarks RNN from CNN (Yin et al., 2017); indeed, their parallel performances depends on the global semantics and the task itself. Similarly to CNN, RNN is broadly used for NLP tasks such as Language Modeling, Machine Translation, and Word/Sentence Classification.

3.5.3 Memory Networks

Also named MemNet (Weston et al., 2015), the technique is still actively researched in the field of NLP as it provides an intuitive approach to attention by using Multi-Hop (Tang et al., 2016), and sets the technique as an interesting competitor to Transformers 3.4. As the Attention Mechanism 3.4.1 builds sets of hidden vectors with its encoder, Memory Networks (MNs) uses the hidden vectors as internal memory instead of feeding them to a decoder for token generation. Further in the Transformers competition, MNs can be applied to similar NLP tasks such as QA (Kumar et al., 2015) by extending the *(Representation, Attention, Answer)* tuples to *(Memory, Question, Answer)* tuples. The Figure 3.5 presents a QA Memory Networks based architecture using knowledge base as knowledge source as initial Key-Value Memories provider, Subject-Predicate-Object Tuples (SPOs).

3.6 Problems

Maybe move this part to the discussion / conclusion

A few comments about the current generative algorithms state as we observed, but our comments may be extrapolated to ML as a whole. Big data is currently starting to make sense in Computer Science as algorithms get incrementally more

optimized, complex, and powerful. In addition to continuously improving computational power, it appears that the large amount of data produced for many years can finally deliver some premise to its potential. However, it raises new questions, such as the privacy implications.

With its impressive 774 Million parameters, GPT-2 uses a large dataset combination of News articles, Reddit comments, IRC conversations, Books, Wikipedia pages, and much more, to train the model, making it humanly impossible to review. Meaning that the model is holding potentially private information, to prove our words, we performed a potential privacy attack on GPT-2 and succeed. Indeed, by merely using meta-information as input, e.g., "[DD/MM/YYYY, HH:MM:SS AM] <USER1>", the model could retrieve a private conversation it once has seen. The method could also recover potential passwords, and so on.

It is essential to repeat; we believe that we are currently at the beginning of a rich potential to NLP, particularly for text generation, as explored in this chapter and confirmed in the next, which implies additional breakthroughs within the incoming year. However, we believe that large models are meant to do good to the world, and must be under a light control to avoid bad actors damaging the image of the upcoming even more impressive technologies.

As a final note, in the scope of our project, we wish not to start a trend toward Sophist Machines, as our study may provide the tools to build algorithms able to manipulate knowledge in the way the author wants.

Chapter 4

Datasets

As the thesis aims at exploring a knowledge-based QA (see Chapter 2.7) and dialogue (see Chapter 2.2) for chatbots combination, this chapter aims at synthesis and compare the current SOTA datasets. With no surprise, we noticed that both NLP research fields are currently in nested competitions, each field plays at defining the new SOTA algorithm and dataset as baselines. In addition to NLP breakthroughs, the competitive attitude makes the community results particularly active and exciting as new techniques, architectures, and incrementally more complex datasets releases at a monthly rate.

4.1 Scope Criterions

Even if the datasets pool base for QA systems and Dialogues is not as exhaustive as datasets present in other NLP fields such as LM or even in other computer science fields, e.g., Computer Vision. The datasets at our disposal are still large, lucky to the recent interest in the fields; however, they vary significantly from each other, e.g., based on their features such as Data Sources, Quantity, or Quality. To narrow the research, as many traits are subjective and intrinsically dependent on the required tasks themselves, we defined high priority criterias.

Knowledge-based The use of KB, such as Wikidata, been defined in the specifications as an ideal knowledge database.

Open Domain The ability to respond to Open Domain question is meaningful to the project as we use a KB containing by design Open Domain relations.

Multiple Supporting Facts As an elegant KB and Open Domain combination, Multi-Hop allows to profit from the KB linked-data architecture to handle more complex questions.

Converstational Support of contexts to handling nested questions is particularly meaningful in our opinion as it provides to QA additional details layers to an answer.

No Reasoning Even if mentioning supporting criteria is essential, we believe that referring to explicitly not supporting criteria is also important. In our case, we ex-

pressly do not support MR in any manner, as reasoning could be a quantifiable task in some datasets.

4.2 Question-Answering

Based on the criterions defined in (see Chapter 4.1), we made an overview Table 4.1 scoping QA datasets related to our work. The following subsections describes the chosen datasets, ending with the worth mentioning datasets.

4.2.1 ConvQuestions

Late 2019, ConvQuestions, a crowdsourced Multi-Hop datasets of 11'200 augmented questions on 5 domains, released with CONVEX (Christmann et al., 2019). The data augmentation is done by asking the Turker to paraphrase each question once and keeping it semantically equivalent and interchangeable. To the initial 5-turns 350 conversations, a non-reordering permutation is applied to each question and its paraphrasing. Finally, the dataset provides a Wikidata Named-Entity Linking to each answer.

4.2.2 SimpleQuestions casted into Wikidata

Initial built with crowd workers by *Facebook AI Research*, SimpleQuestions(Bordes et al., 2015) was entity-linked to the Freebase KB for its 108'442 (question, answer, language) triples. For their research, the AskPlatypus (Diefenbach et al., 2017) team used automatically generated mappings of 49'202 SimpleQuestions triples to Wikidata. Note that at the time of building their dataset, only 21'399 were answerable over Wikidata.

4.2.3 Worth Mentioning

Even as part of this work., the opportunity to use the following datasets was not appropriate; we believe that they are worth mentioning due to the attention they attracted lately with fine-tuned pre-trained language models (see Chapter 3.4) QA systems.

Stanford Question Answering Initially presented in 2016, Stanford Question Answering Dataset (SQuAD) (Rajpurkar et al., 2016) was propelled by the raised of Transformers as it was the first massively crowdsourced reading comprehension Wikipedia-based dataset with a 86.6% human performance, making it an exciting challenge for future QA systems. In 2018, SQuAD 2.0 (Rajpurkar et al., 2018b) was released with an additional set of 50'000 unanswerable questions, adversarially similar to answerable questions present in the first version. Note that by design, SQuAD focuses on confusing questions, approachable by combining paraphrasing and text summarization NLP tasks.

Conversational Question Answering Challenge Released in 2019, Conversational Question Answering (CoQa) (Reddy et al., 2018) is a Multi-Hop dataset containing 127'000 questions obtained from 8'000 conversation over 7 difference sources. With a human-performance of 88.8% CoQa implies reading comprehension with coreference and pragmatic reasoning.

Question Answering in Context Since 2018, Question Answering in Context (QuAC) (Choi et al., 2018) is challenging QA system with its Multi-Hop dataset containing 100K crowdsourced evidence-based questions aiming at providing QA in a dialog manner with context holding. Questions are designed to be open-ended, unanswerable without context and focusing on missing information.

Compare SQuAD, CoQa and QuAC In 2019, a comparative study (Yatskar, 2018) have been conducted for the previously mention three datasets, comparing them on the basis of unanswerablility, Multi-Hop, and question abstraction.

4.3 Dialogue Datasets

As the project scope requires a chatbot able to answer NL, we explored the available conversational datasets. Compared to the QA datasets (see Chapter 4.2), we discovered an underrepresented field of NLP as the presence is relatively poor. Recapitulated in the Table 4.2, we focused on the datasets featuring on QA setups and multi-domains dialogue openness. We identified a unique dataset matching our requirements and two worth mentioning datasets, reinforcing an overall subjective view implying that dialog-based NLP research currently in standby. Indeed, it would not surprise us, as pre-trained language models have started reaching a popular peak of interest, that the field of Dialogue gains a sudden interest and challenges GPT-2 (Radford et al., 2018).

4.3.1 Natural Questions Corpus

Another jewel of 2019, with over 323'000 dialogues, *Google's* Natural Questions Corpus dataset (Kwiatkowski et al., 2019) is a benchmarking approach for NL generated answers in a QA environment, making it particularly interesting for fine-tuned pre-trained language models like GPT-2 (Radford et al., 2018). Its goal is to provide an appropriate training and testing set for QA systems, by pairing *Google Search Engine*'s real user queries to a large pool of crowdsourced cross-annotations, they call "high quality annotations", to guarantee answer quality over documents. Additionally, their mythology defines new metrics to evaluate answering performances. Interestingly, the dataset provides statistics, a long answer, a short answer, and an answer Named-Entity Linking to a Wikipedia page in most cases.

4.3.2 Worth Mentioning

The following datasets are particularly interesting from a Chatty (see Chapter 2.2) point of view, but no further out-of-the-box features without pre-processing or data augmentation are present. In our case, no Wikidata Named-Entity Linking is available, nor the data is set explicitly in a QA manner, making it particularly random in various contexts. However, the datasets are still impressive by their quantities and their conversational feature.

Twitter Conversation Triple This 2015 dataset uses Context-Message-Response triples as storage architecture, making it particularly interesting for parallelized training. Additionally, with its impressive 129 Million tweets tuples, it makes it the most

substantial research dataset released until the time of writing. The dataset is currently combined with BLEU (see Chapter 5.2) to evaluate generated dialogue, often present in the field of machine translation.

Ubuntu Dialogue Corpus *Ubuntu* released its chat logs (Lowe et al., 2015), in 2016, containing over 1 Million multi-turns dialogues. The interest to this dataset comes from its relatively large size containing long technical contexts, and by design, it does not require exhaustive feature engineering to train over out-of-the-box.

Datasets	Release Date	Nested Questions	Hops	Open Domain	Queries	Docs	Query Source	Answer Type
ConvQuestions	2019	**Yes**	**Multi**	**5**	11K	350	**Wikidata**	Spans
Google Natural Questions Corpus	2019	No	Single	**Yes**	323K	??	Wikipedia	Spans
SQuAD 2.0	2018	No	Single	**Yes**	151K	853	Wikipedia	Spans, Unanswerable
CoQa	2018	**Yes**	Single	**Yes**	127K	8K	Children's Stories, Literature, Mid/High School Exams, News, Wikipedia, Reddit, Science	Spans, Unanswerable
QuAC	2018	**Yes**	Single	**Yes**	100K	14K	Wikipedia	Spans, Unanswerable
HotpotQA	2018	No	**Multi**	**Yes**	113K	591	Wikipedia	Spans
DuReader	2018	No	Single	**Yes**	300K	1.5M	Web Search	Spans
TriviaQA	2017	No	**Multi**	**Yes**	650K	95K	Trivia	Spans
RACE	2017	No	Single	No	97K	28K	Mid/High School Exams	Multiple choice
Narrative QA	2017	No	**Multi**	**Yes**	47K	1.6K	Movie Scripts, Literature	Spans
SearchQA	2017	No	**Multi**	**Yes**	140K	6.9M	Jeopardy	Spans

Datasets	Release Date	Nested Questions	Hops	Open Domain	Queries	Docs	Query Source	Answer Type
NewsQA	2017	No	Single	**Yes**	100K	10K	News	Spans
QAngaroo WikiHop	2017	No	**Multi**	**Yes**	51K	528K	**Wikidata**	Spans
QAngaroo MedHop	2017	No	**Multi**	No	2.5K	528K	Medline, Drugbank	Spans
CNN / Daily Mail	2016	No	Single	**Yes**	1.4M	93K / 220K	News	Spans
Children's Book	2016	No	Single	No	688K	108	Children's stories	Multiple Choice
SQuAD	2016	No	Single	**Yes**	108K	536	Wikipedia	Spans
MS MARCO	2016	No	Single	**Yes**	100K	200K	Web Search	Spans, Unanswerable
SelQA	2016	No	Single	**Yes**	8K	486	Wikipedia	Spans
INFOBOXQA	2016	No	Single	**Yes**	15K	150	Wikipedia	Spans
WikiQA	2015	No	Single	**Yes**	3K	29K	Wikipedia, Web Search	Sentence Selection
SimpleQuestions	2015	No	Single	**Yes**	109K	6K	**Freebase**	Spans
bAbI tasks 1 to 6	2015	No	**Multi**	**Yes**	6M	??	??	Spans
MCTest	2013	No	Single	**Yes**	2.6K	660	Children's stories	Multiple Choice

Table 4.1: Overview of Question Answering Datasets. In bold the features identified to be meaningful for the Thesis.

Datasets	Release Date	QA	Open Domain	Dialogues	Utterances	Query Source	Dialogue Type
Google Natural Questions Corpus	2019	**Yes**	**Yes**	323K	??	Wikipedia	Human to Human,
Reddit	2017	No	**Yes**	54M	??	Comments	Human to Human
DSTC4	2016	**Yes**	No	35	??	Chat logs	Human to Human
Twitter Triplets Corpus	2015	No	**Yes**	129M	87M	C-M-R Tweats	Human to Human, blogging
Ubuntu Dialogue	2015	No	**Yes**	1M	7M	Chat logs	Human to Human
Sina Weibo	2015	No	**Yes**	4.4M	8.8M	Posts, Comments	Human to Human, blogging
DSTC2	2014	No	No	3K	24K	Chat logs	Human to Computer
DSTC3	2014	**Yes**	No	2.3K	15K	Chat logs	Human to Computer
DSTC1	2013	**Yes**	No	15K	210K	Chat logs	Human to Computer
Twitter Corpus	2010	No	**Yes**	1.3M	3M	Posts, Tweats	Human to Human, blogging
OpenSubtitles	2009	No	**Yes**	70M	??	Movies	Human to Human

Table 4.2: Dialogues Datasets Overview. In bold the features identified to be meaningful for the Thesis.

Chapter 5

Evaluation

In this chapter, we overview current evaluations in NLP for the two tasks our project is combining, QA systems and computer-generated dialogues. Often, determining if a model is working as expected is a hard task, it depends on the tasks itself the datasets used. Naively, one could build a complex supervised protocol to evaluate a model's success. Still, often it is not enough due to the chaotic nature of training or to multiple exceptions to handle; indeed, it would require an unrealistic amount of human-power to build a general evaluation protocol, particularly for NL tasks. Instead, it is common to make and combine grounded tasks evaluation and sub-tasks to get metrics.

5.1 Question Answering Systems

Empirically to the QA Datasets SOTA from chapter 4.2, we targeted our research at providing comparative results to our later described QA system GraphQA (see Chapter 7). We explored additionally results from SQuAD (see Chapter 4.2.3) and (see Chapter CoQa) as grope to Transformer performances at QA tasks. With difficulty, we represented in the Table 5.1 benchmark results; indeed, it appeared that the majority of baseline competitor to the QA task are using different datasets, making particularly difficult to syntethis as no evaluation dataset baseline is defined yet.

5.1.1 CONVEX

CONVEX (Christmann et al., 2019) has been developed to handle the ConvQuestions dataset (see Chapter 4.2.1), and at time of writing, it was not evaluated on additional datasets. CONVEX uses subgraphs to manipulate Wikidata entities extracted via TAGME (Ferragina et al., 2010) from the Wikidata KB to answer turn-based Multi-Hop conversational questions. The authors use the Mean Reciprocal Rank (MRR) metric in addition to the Top-1 and Top-5.

5.1.2 qAnswer

Initially build for the DBpedia KB, qAnswer converts NL questions into SPARQL query with a template-based model (see Chapter 2.7), and later got extended to the Wikidata KB. It uses Wikipedia pages to extract lexicalizations and match NL, entities, and relations. The authors originally used F1 as metric on the QALD-5 test

corpus (Lopez et al., 2013), and later got evaluated on the ConvQuestions dataset (see Chapter 4.2.1) by the CONVEX authors (see Chapter 5.1.1) with MRR, Top-1 and Top-5 metrics.

5.1.3 Platypus

Similar to qAnswer, Platypus (Pellissier Tanon et al., 2018) is a template-based model (see Chapter 2.7) trained to build SPARQL request for Wikidata as well, but has been released in open-source. It was initially trained, tested, and evaluated with F1 on the Wikidata mapped SimpleQuestions dataset (see Chapter 4.2.2). Later Playtypus got additionally evaluated on the ConvQuestions dataset (see Chapter 4.2.1) by the CONVEX authors (see Chapter 5.1.1) with MRR, Top-1 and Top-5 metrics.

5.1.4 Honorable Mention

It is no spoiler to mention that Fine-Tuned Pre-Trained Language Models 3.4 are currently under the spotlights. By curiosity, we wanted to get hold of the phenomenon by investigating and reporting a few comparative results of Transformer, Long Short-Term Memory (LSTM), and RNN Models. We noticed based on the Table 5.1 that BERT-based (Devlin et al., 2019) models in their base form with pre-training are largely outperforming non-Transformer-based models. As a final note, we didn't judge it necessary to include extended BERT-based models to the table as we believe that it is a field of study by itself and out of our scope. However, we just wanted to mention that, at time of writing, the best baseline in CoQa (see Chapter 4.2.3) leaderboard with an overall 90.7%, is a compositional model combining RoBERTa (Liu et al., 2019) (BERT-based), Adversarial Training (AT), and Knowledge Distillation (KD) (Ju et al., 2019), and as a friendly reminder, that Human Performance is set at 88.8% on the same dataset.

5.2 Generative Systems

As mentioned in chapter 4.3, it appeared that so far, progress is in standby for the computer-generated text NLP task, taking machine translation and the Oracle approach apart. As far as our research went, we found two papers supporting interesting facts about computer-generated texts. According to the first study (Gehrmann et al., 2019), the authors describe a technique to detect computed generated text by focusing on the induced artifact from generating text. The second paper (Graefe et al., 2018) performed a study on the Readers' perception of computer-generated news. They concluded that people enjoy reading computer-generated texts because it is, in fact, computer-generated, which is currently fascinating to humans; however, a long term study still has to be conducted.

BLEU Originally the Bilingual Evaluation Understudy (BLEU) was created as a completely automated metric for machine translation, but is in theory, adaptable to other NLP tasks such as NL generation. By simply comparing the machine output and the ground truth, BLEU is relatively faster than human translators, computationally friendly, and benchmarkable. It implies that by design, BLEU does not take into account the meaning of the sentence, nor it is taking into account the sentence structure, nor it evaluates how a human would interpret the sentences.

Meaning that rich languages do not evaluate well and imply that BLEU perform well in machine translation to measure entire corpora for a good reason, but it is not acceptable in our study as we expect a meaningful human-like evaluation.

ROUGE ROUGE is a BLEU (see chapter 5.2) adaptation focusing on Recall instead of Precision, by evaluating the ground truth to the output.

GLUE Since 2019, the General Language Understanding Evaluation (GLUE) benchmark is a complete framework used to train, evaluate on a collection of datasets and then compare models relative to one another on various NLP tasks. The dataset collection is composed of nine (four are kept private) relatively difficult datasets designed to test MU, which is particularly interesting for fine-tuned models.

Natural Questions Corpus Metric As mentioned in chapter 4.3.1, *Google's* Natural Questions Corpus dataset (Kwiatkowski et al., 2019) is a benchmarking approach that defined a new metric to answer evaluation combining F1 from Long Answers and Short Answers with multiple annotators, making it 25-way Annotated. The technique consists of asking annotators for given questions to say if the question is fact-seeking or not, return a long and short answer pair. As the next step, the previously annotated questions are sent to 4 annotators; whose goal is to evaluate the annotations. Based on majority expert judgements, the annotated questions are categories as "Correct", "Debatable" or "Wrong". Finally, the annotated questions evaluation is measured by calculating the Precision and Recall of the long and short answers.

Humans Although it is evident that automatic evaluation metrics are not entirely reliable for text related NLP tasks. The only available solution so far is the use of Humans to perform manual validation, either by Crowdsourcing via Mechanical Trucks, or by asking colleagues. However, often the results even obtained from humans are not qualitatively optimal. Indeed, distraction is human nature, particularly in Mechanical Trucks setups, resulting in protocols such as *Google's* Natural Questions Corpus dataset (see chapter 4.3.1) are established, implying multiple verifications, with the goal to produced the optimal labelings and evaluations.

Models	Release Date	Handles Nested Questions	ML	**SimpleQuestions F1**	**ConvQuestions MRR**	SQuAD2.0 F1	SQuAD1.1 F1	CoQa F1
ALBERT	2019	**Yes**	Transformer	-	-	92.215%	-	-
BERT-base finetuned	2019	**Yes**	Transformer	-	-	83.061%	93.16%	81.1%
CONVEX	2019	**Yes**	Information Retrieval	-	0.2012	-	-	-
qAnswer	2019	No	Template	-	0.0294	-	-	-
BiDAF++	2018	**Yes**	CNN	-	-	-	-	67.8%
Platypus	2018	No	Template	79.96%	0.0022	-	-	-
QANet	2018	No	Transformer	-	-	-	82.7%	-
DrQA	2017	**Yes**	RNN	-	-	-	79.353%	-
BiDaF	2016	No	LSTM	-	-	-	81.525%	-
MemNet	2015	No	Memory Network	77.97%	-	-	-	-

Table 5.1: Question Answering Benchmarking Overview

Part III

Design and realization

Chapter 6

Analysis

In this chapter, we analyze the SOTA (see Part II) to define the scope of the POC and add additional knowledgable details to project specifications as our initial understanding of the thesis subject increased. As a kickoff to research the SOTA and for the analysis, in addition to peer consulting with our lab colleagues in the field of NLP, we used curated lists e.g., awesome-nlp on Github (Keon, 2017). The knowledge accumulation started naturally to chain as we began to read papers mentioned in other papers.

6.1 Rescoping and Motivations

Extrapolated from the SOTA part II, we present in this section the process held to get to the final project scope based on the accumulated knowledge thru NLP SOTA exploration.

6.1.1 Initial Project

Our research initially started as a satellite to the *AI-News* project (see Chapter 1.1.2), which is currently using a Retrieval approach (see Chapter 2.3) combined with an intents and entities extraction, to return highly pondered recent articles from an elastic search in a Ruled-Based chatbot (see Chapter 2.4) format. The *AI-News* project scoped our research indirectly toward finding solutions to bring QA systems to the field of journalism; however, it early rescoped to open domain knowledge as the Wikidata KB provides crowdsourced general knowledge. Additionally to the QA scope, the specification implies NL answers generation. We intended to build a QA system to generate BLEU approved corpora based on the Wikidata KB, then compare them to SQuAD benchmarking (see Chapter 4.2.3), with the ultimate purpose to compare Pre-Trained Language Models such as BERT on your new dataset. As the starting point, we decided to bootstrap our project with SOTA related papers filtered by code availability and reproducibility.

6.1.2 Initial Ideas

To achieve our initial project, we brainstormed the following ideas. Indeed, we expected to explore in detail the field of QA evaluation in particular Oracle-based solutions, oracles are particularly meaningful in our context as we would use Wikidata as an oracle. Additionally, we planned to explore in more detail the effects

of fine-tuning pre-trained language models such as BERT, with the intent to create a challenging dataset. Based on a finite state grammar, our evaluation would initially generate a set of 100'000 Wikidata SPO-based questions and using word-embedding similarity feature to paraphrase those questions to increase complexity. We even discuss an additional feature to generate QA multi-turns conversations (see Chapter 2.8.1) by self-training a model with an AL approach.

6.1.3 Second Brainstorming Iteration

As research progressed, we took the party that we did not want to be just another benchmarking system for QA systems, similarly to our initial decision not to become an nth fine-tuned pre-trained language model. As a result, our second brainstorming iteration brought quite a new scope to the project. Indeed, we focused mainly on a Multi-Hop multi-turns conversations approach, by aiming at building an interactive and proactive reversed Akinator-like (Elokence, 2007) QA system. The system would use a "child" learning approach as the model itself starts with no knowledge, and incrementally learns new knowledge by interacting with users. The goal is to teach the model to retrieve information from a KB by itself. The game consists of a randomly selected Wikidata entity as the answer, and the user is requested to help, via a gamified conversational interaction, the bot to build a path in Wikidata, as it asks questions to the user. The proactive ability would a generative approach to scenario-based chatbot like HelloJam (SAS, 2014), which uses intermediary steps as a proactive approach. As a training bootstrap, we initially planned to build a conversational simulator using multi-hop datasets such as ConvQuestions, SQuAD, or QuAC.

6.2 Third Brainstorming Iteration

Quickly, the previous brainstorming iteration raised problems such as the truth user interest for such games and the bias induced by genuine or intentional human errors, without mentioning the pre-processing needed to build a meaningful training dataset, but we kept some ideas. We believed that the Multi-Hop multi-turns conversations are an essential feature to our work, combined with the interactive and proactive approach between the bot and the user, which is a particularly exciting application in the NLP field. We then suggested a new shift in the project scope to build NL QA chatbot allowing the user to interact with the conversation and the knowledge in a meaningful manner. The goal is to provide the user the ability to check returned answers with facts and correct them on the fly if needed. Additionally, we wanted to add a multi-model approach to handle virtual personalities as a flavor of personalization to the user.

6.2.1 Final Brainstorming Iteration

The project scope shifted one last time. We noticed that the applications in the field of QA, and Generative chatbot are extensive, and it appeared that it is possible to find a paper already mentioning, even lightly, what we believed original ideas. As a contradiction, we decided at aiming directly at the roots of QA and Generative systems as a whole. To do so, we kept the multi-hops reasoning, multi-turn conversations, and the Wikidata KB as constraints to the project. Indeed, we did not want to be just another Transformer related project trying to define the new baseline.

Our resources at disposal were limited, and we wanted to take a new approach by exploring an original technique for QA, in particular, Sub-Knowledge Graphs. To achieve our latest project, we examined existing SOTA systems providing, in addition to the paper, a runnable code to get started. We initially aimed at incrementally improve the original work to impact the NLP field. Our first step was to reproduce the results; then, the second step was to retrain and provide additional value to the original work. And finally, adapt the initial project to the News field with few tweaks.

6.3 Question-Answering Systems Choices

Based on the SOTA (see Part II) and the previous rescoping section 6.1, we aim at building a QA Multi-Hop and multi-turn converstational chatbot using sub-graphs from Wikidata. We found a unique direct competitor to our work, and we are using its baselines and nested baselines to compare our work.

6.3.1 Competitors

We initially explored QA system using the Wikidata KB by default, and providing the must-have features defined in the project scope, our research revealed a unique candidate and two related candidates as they define the baseline of our main candidate.

CONVEX As our direct competitor, CONVEX (Christmann et al., 2019) (see Chapter 5.1.1) extracts sub-graph from Wikidata KB. It employs the sub-graphs as context holders and uses them to answer context-related questions via a frontier algorithm. It finally extends the context-graph with the new answers, making the graph more precise at answering detailed questions.

qAnswer and Platypus Both QA systems are using a template-based model (see Chapters 5.1.2 and 5.1.3), and are defined as baseline in our main candidate and they are designed to for Wikidata queries. Note that they are Single-Hop and trained on the SimpleQuestions dataset, which we also plan to use as a fair evaluation.

6.3.2 Datasets

Our initial predetermined dataset was SQuAD due to its popularity and its public leaderboard, as it would have been a pleasant additional comparison to our work. However, SQuAD is designed to answer fact-based questions with a span extracted from the given context paragraph, often implying a mismatch with fact entities such as in the Wikidata KB. It is also a reason for CONVEX to not evaluate on this dataset, similarly to qAnswer and Playtypus. Finally, in the scope of the project, we did not plan to adapt the SQuAD dataset for Wikidata-based QA systems.

ConvQuestions As descried in Evaluation Chapter (see 5.1.1), CONVEX (Christmann et al., 2019) is design to use the ConvQuestions dataset (see Chapter 4.2.1), which is a multi-turn conversations and Multi-Hop QA crowdsourced dataset. CONVEX uses also this dataset with its baselines providing an initial evaluation for all three competitors.

SimpleQuestions As presented in the Dataset Chapter (see 4.2.2), SimpleQuestions has a Wikidata KB adaptation provided by Playtpus authors (Pellissier Tanon et al., 2018). This dataset will be used to benchmark our system in addition to all three competitors, as a fair approach to evaluate Single-Hop systems on Single-Hop design datasets.

6.3.3 Benchmarking

To measure and compare our work, we will set up two benchmarks. The first benchmark will evaluate the Single-Hop capabilities of each competitor on the Wikidata adapted SimpleQuestions dataset. The second benchmark is focusing on the Multi-Hop and multi-turn conversations capabilities from each competitor. As qAnswer and Platypus are not designed for multi-turn conversations, we will extend them with CONVEX and our work during the multi-turn task, as both projects can work on top of other algorithms (see the next Chapter 7).

6.4 Texts Generation Choices

We plan to used two SOTA pre-trained LM in there vanillia large format, BERT (Devlin et al., 2019) and GPT-2 (Radford et al., 2018), for our text generation NLP task. BERT will be used as a text filler for the SPOs paths extracted complementary to the answer from the Wikidata Sub-Knowledge Graphs. GPT-2 will be used as a complementary facts generator to the answer.

Evaluation We consider *Google's* dataset (see Chapter 4.3.1) as promising; however, we did not plan to evaluate our generated NL answers on this dataset as the project planning is too tight to include this protocol. We instead, we will assess the dialogues generated manually. An evaluation idea would be to run a campaign with 10 to 20 users evaluating 10 to 20 questions, using the evaluation tags: "Good", "Neutral", or "Bad". However, this dataset is worth considering for future extended work.

6.5 Final Project Scope

GraphQA, as we name it, is a QA Sub-Knowledge Graphs chatbot using the Wikidata KB database. It extracts answers to questions from a complete Linked Data database by extracting small portions of the main database and manipulates it as a graph. The graph manipulation implies context holding, extension, and refining. Initially, based on CONVEX, we plan to improve their Information Retrieval (IR)-based answering to the first answer, as it is their bottleneck, which is particularly sensitive as it defines a successful anchor to a conversation-driven QA. We wish to also explore sub-graph capabilities with temporally depend contexts, and their overall performances. Additionally, NL capabilities must be present by combining two complementary pre-trained language models.

Expected Features

We define below the features we expect to be present in GraphQA as a Multi-Hop Conversational QA Chatbot.

- Extract question keywords.
- Extract a question-related Sub-Knowledge Graph.
- Compute the answer from the Sub-Knowledge Graph.
- Extract an SPO Tuple meaningful to the question.
- Prune the Sub-Knowledge Graph of context meaningless elements
- Generate a NL answer from the SPO Tuple.
- Extend the NL generate an answer with additional facts.
- Extend the Sub-Knowledge Graph with new questions.

Nice to have

As we do not expect to have the time to add additional features to the primary features described in the above subsection. We are mentioning some features that we would like to see in the project one day. Note that KB databases are by design only as good as the data the relations they hold, which is limited also by the employed schemas, implying if the reasoning information such as quantities were not manually referenced into the database, the answer is impossible to retrieve.

- Give the ability to merge pre-generated sub-graphs.
- Use of multiple modules to manage different processes.
- Include reasoning modules such as inductive logic or quantification.
- Provide an auto-correct tool to users.
- Provide an auto-complete tool to users.
- Provide a paraphrasing tool to generated answers.
- Use the multilingual Wikidata's propriety paraphrase into a translation.
- Use Wikidata's multi-properties to paraphrase questions and answers.
- Handle pre-built sub-graphs for particular subjects such as Articles, Countries, Movies, or Famous People
- Track users custom sub-graphs to track the overall context.
- Use consensus-based modules to complementarily provide the best answers.
- Show visually to users their sub-graph generation
- Provide on-the-fly tools to modify sub-graphs.

6.6 CONVEX Q0 Solutions

As mentioned in previous sections and represented on the Figure 6.1, CONVEX (Christmann et al., 2019) has an issue with the first answer. The issue has been briefly explored and confirmed in the CONVEX paper. Indeed, for a0 (the initial answer), CONVEX uses a proprietary Named-Entity Recognition system, TAGME (Ferragina et al., 2010), as its Wikidata entities identify system. Then places into an empty a subgraph returned entities with their relation extracted from the Wikidata KB. This process is solely relying on a TAGME to answer the initial question, which are often interpreted as lucky guesses. As one of our main contributions, we want

to fix this issue, and to do so, we propose five different solutions, including a naive one.

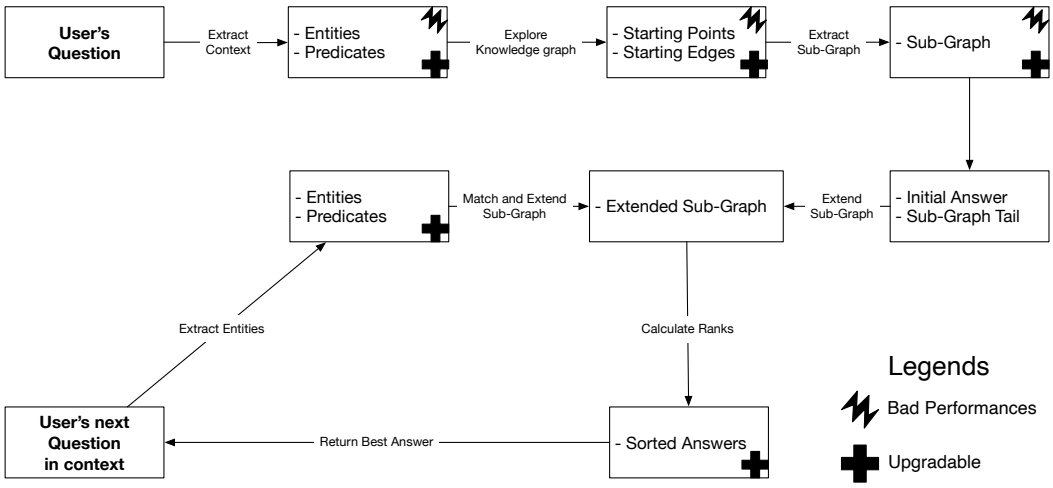


Figure 6.1: Illustrative representation of the high level CONVEX architecture. the diagram includes the identified part having bad performances, and shows the upgradable components.

6.6.1 0th Solution: Naive Approach

Our naive approach is to use text summarization to extract relevant information and build the initial sub-knowledge graph by matching the entities present in the Wikidata KB. It is indeed not an impressive approach by at least we have control over the extracted data, and we can study the related induced behaviors, then tune the system with additional models.

6.6.2 1st Solution: BiDAF++

The BiDAF++ model, presented with QuAC (Choi et al., 2018), is based on BiDAF (Seo et al., 2016) augmented with self-attention (Clark et al., 2017) and ELMo (Peters et al., 2018), and particularly used as non-Transformer baseline on multiple QA datasets. We could explore training on multiple dataset such as Google's Natural Questions Corpus (Kwiatkowski et al., 2019), ConvQuestions (Christmann et al., 2019), CoQA (Reddy et al., 2018) and NewsQA (Trischler et al., 2016). The model would provide us the answers to the questions, and we would build the initial subgraph based on the matched entities id found in the question and the entity found as the answer.

6.6.3 2nd Solution: Multi-task learning

Multi-task learning for large scale KB (Shen et al., 2019) is by design handling the conversational QA format, making it particularly appealing to our project. The techniques model trained to parse questions and point them into the KB via pointers, which avoids the propagation of errors and simultaneously exploits the pointer property to share the linked information.

6.6.4 3rd Solution: Knowledge Graph Embedding

Knowledge Graph Embedding (Wang et al., 2019) is model trained to locate entities (subject) and predicates individually in a Knowledge Graph (KG) and then predict the tail (object). However, this technique, as described by the authors, works only for simple questions, which conflicts with our Multi-Hop constraint, implying to extend the model capabilities to Multi-Hop handling.

6.6.5 4th Solution: Fine-tuned Pre-trained Language Model

Even if we do not expect to use this solution, it is still worth mentioning. Indeed, the final solution would be to fine-tune a transformer-based model on multiple QA datasets similarity mentioned in the 1st solution above (see Subsection, 6.6.2).

6.6.6 Our representation in the Chatbot Cartography

To conclude this chapter, we updated the chatbots cartography as defined in the chatbot state-of-the-art chapter (see 2.9) to illustrate our position. (See Figure 6.2)

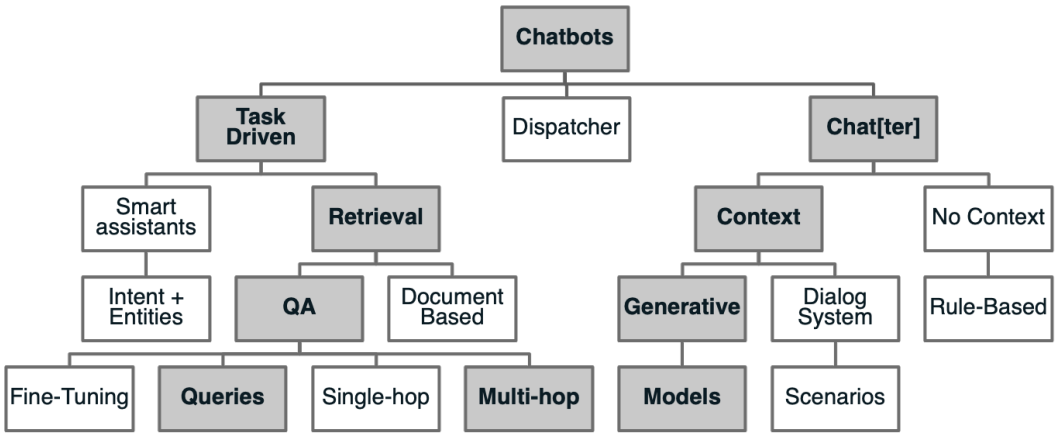


Figure 6.2: Represents GraphQA's positions in the chatbots cartography as defined in the chatbot state-of-the-art chapter.

Chapter 7

GraphQA

7.1 Initial Architecture

retrospectives to the analysis

7.2 Going Further

Compare wordnet and word embedding, For unknown words or predicates try to transition the vector based on a synonyme from the wordnet, and compare with the word embeddings similarity word.

7.3 GraphQA Architecture

Multi models from presentation Grounded approach with zero learning similar results

3 Versions, detail all feature from version 3

Compatible with telegram until version 2.0 due to the tensorflow session not being correctly managed by the python telegram bot package, so autocorrect is not working.

7.4 Version 0

initial features Pre print problems, made up values, confusing writing so we don't know what is motivational made ups and what are actual results.

7.5 Version 1

7.6 Version 2

Filling paths holes with BERT

7.7 Version 3

Extending sentences with GPT-2

7.8 The technologies we used

HDT to index and compress and charge in ram the whole wikidata linked data database. Query compression format for linked data Spacy, industry famous for POS, POS-tagging, NER, NEL, etc, wikidata and wikipedia entity linker Spacy is an impressive industry used multi tool framework that we used as it allows combine Named-Entity Recognition Named-Entity Linking Part of Speech Part of Speech Tagging .. We used the new 2019 released version 2 Deepcorrect NetworkX

7.9 Premise

We tried to extend wikidata KB as the version we found is 2 years old. What would it mean to build it from scractch? Try to optimise change the word embedding. Optimise Entity Linking. The solution to use the online database in addition to the local version.

7.10 Problems

Slow download of the database 50 GB Uncompressed data is 500 GB No enough ram to compress the latest version of the database Ram Crashes Use of a second machine Initial problems at downloading the dataset as the main source was down, so we had to contact the convex author to get a hold on the dataset Tried to extrapolate to other KB such as ?DEBD?, the algorithm focused on the Wikidata schematics, each KB as their own. Problems with cache

Part IV

Retrospective

Chapter 8

Results

Talk about how the methodology involved /resulted during the project

8.1 Problems

Memory Leaks in the latest version

8.2 Hardware

iCoSys provided two Dedicated Servers, a lambda lab for the whole project, and a cpu based machine
CPUs Specification • CPU: 8x 1.2Ghz AMD Opteron 6176 • RAM: 192GB DIMM

8.3 Benchmarks

On ConvQuestions, we did a complete reevaluation of all alorithms for consistency. and the missing precision and recall metric. GraphQA is very long to run, which resulted in this amount of results. But it's already a good pool of questions to evaluate from. It's clear that it would be better to evaluate on more data. Concerning the generated sentences evaluation we asked humans to do it, as we didn't have enough time to implement the google protocol. And the main focus of GraphQA is the ability of exploiting sub-graphs, and how transformers are performing at what they are designed to do.

Chapter 9

Project Management

The milestones were meant to be adjustable based on how the project go

To avoid getting overwhelmed with the latest NLP papers in the field of QA systems, and Generative Systems (GSs), the author defined workflow components to gather valuable information:

- Get up to date with the NLP technologies used at *iCoSys*.
- Explore community-made curated lists[1].
- Stay informed of the breakthroughs via social medias[2].
- Find reviews and articles vulgarizating recent papers[3].
- Read papers[4].

9.1 Objectives

9.1.1 Intrinsic

This subsection presents the general objectives related to the master's thesis.

Primaries

- Propose a project specification and planning.
- Analyze the state of the art of existing technologies and technics of QA systems and Generative AI.
- Overview digital transformation in journalism and review the current status of the AI-News project.
- Document the study and write the thesis.

9.1.2 Fact-based Question Answering Chatbot

The first objective is to make, based on the State of the Art (SOTA), an algorithm that takes a question as input and outputs a response, as illustrated on Figure 9.1

[1] Using *Awesome* lists from `github.com` as starting point
[2] Examples from `reddit.com` /r/MachineLearning, /r/LanguageTechnology, /r/deeplearning
[3] Particularly from community based `medium.com` articles
[4] Most of the articles are coming from `arxiv.com` and `aclweb.org`

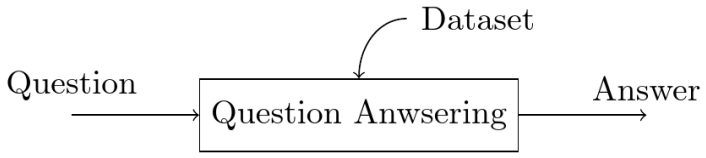


Figure 9.1: Suggested QA diagram

Primaries

- Select existing papers and projects treating the subject as a starting point.
- Identify relevant datasets.
- Develop one or more POC.
- Test and evaluate solutions.
- Suggest improvements, possible continuation, and future outcomes.

Secondaries

- Extend the QA chatbot using "tailored" knowledge, e.g., Model Fine-Tuning with press content.

9.1.3 Generative QA Chatbot

The second objective is to extend the output from the QA system, from the first objective, by enhancing the answers and generate human-like sentences from the enhanced answers. The initial vision for this objective is as illustrated in Figure 9.2, a two parts system. The *Enricher* enriches the answer from the QA system, e.g. using a knowledge base[5]. The *Generator* aims at creating readable text from the enriched answer. Besides, we could also use user profiles[6] as input to those two parts.

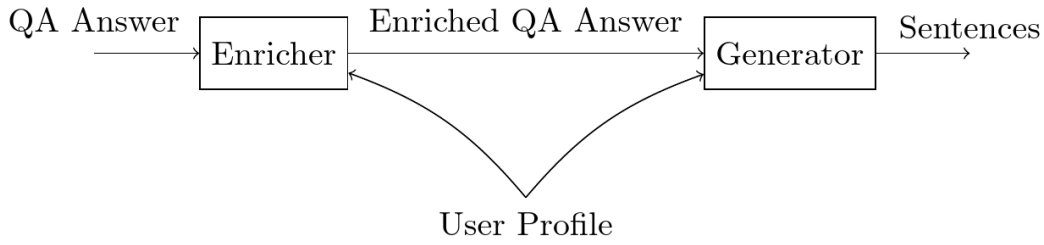


Figure 9.2: Suggested Generative QA diagram

Primaries

- Investigate a rule-based system for keyword enrichment.
- Generate sentences with keywords.
- Identify relevant datasets.
- Develop one or more POC.
- Test and evaluate solutions.
- Suggest improvements, possible continuation, and future outcomes.

[5]Wikidata.org, a Freebase-based (Bollacker et al., 2008) knowledge base or Google's Knowledge Graphs (Singhal, 2012)

[6]Fictive profiles in the context of the thesis

Secondaries

- Use advanced strategies to enrich keywords.
- Use advanced text generation technics such as GTP-2[7].
- Use user profiles to customize the outputs.

9.2 Initial Plan

9.2.1 Contraints

Timeframe: 19 weeks
Starting date: 16.09.2019
Ending date: 07.02.2020

9.2.2 Methodologies

For consistency, the project is separated into two methodological parts. The first third, as the project targets information gathering and self-study, we use a standard sequential project management methodology. For the next two-thirds of the project, we will be using an agile methodology to perform incremental progress while exploring.

Back to level Milestones

First third of the study, from **16.09.19 to 25.10.19 (6 weeks)**.

M1. Initial MT plan and project specification

M2. Review the state of the art of the NLP and NLU technologies and refine the plan if needed.

Diving into the subject Milestones

From 28.10.19 to 07.02.20 (13 weeks), the following two-third of the work is composed of 6 sprints of two weeks each and one week to finalize the thesis.

M3. Basic QA Chatbot

M4. Evaluation of basic QA Chatbot

M5. Basic generative QA Chatbot

M6. Evaluation of basic generative QA Chatbot

9.2.3 Gantt

The Figure 9.3 represents the chart for the initial plan.

[7] OpenAI's GTP-2 Algorithm (Radford et al., 2018)

9.3 Tasks

9.3.1 Initial Tasks

Primaries

1. AI in journalism state of the art
2. NLP and NLU state of the art
3. Find relevant datasets
4. Find existing projects and papers responding to the questions
5. Explore documents' topics extraction
6. Explore the Wikidat and knowledge graphs
7. Explore question-answering technologies and technics
8. Evaluate by comparing to similar systems

Milestones

1. Initial MT plan and project specification
2. Overview topics extraction technics
3. Overview Wikidata and knowledge graphs technics
4. Overview text transformative and generative technics
5. Mindmap of the current NLP and NLU technologies
6. Pytorch hands-on

Secondaries

- Explore AI implications in journalism
- Explore AI personalization implications
- Explore text generative technologies
- Explore profile-based customization
- Explore text transformative technologies
- Explore the attention mechanism
- Explore text summarization
- Explore text flavoring to write as a specific author
- Explore news extraction from social media
- Explore news baseline extraction
- Explore news drafts and briefs generation
- Explore text adapted suggestions for journalists
- Explore knowledge graphs as content enrichment
- Explore multiple sources cross-checking to reduce fake news
- Explore tracker for the original source
- Explore autonomous knowledge gathering

- Explore machine-generated factual discussions
- Explore machine self-training
- Explore chain reasoning
- Explore artificial common sense
- Explore artificial intuition
- Explore on the fly translations
- Make overall improvements

Milestones

- Basic topic extraction from documents
- Basic conversational agent
- Basic journalistic agent

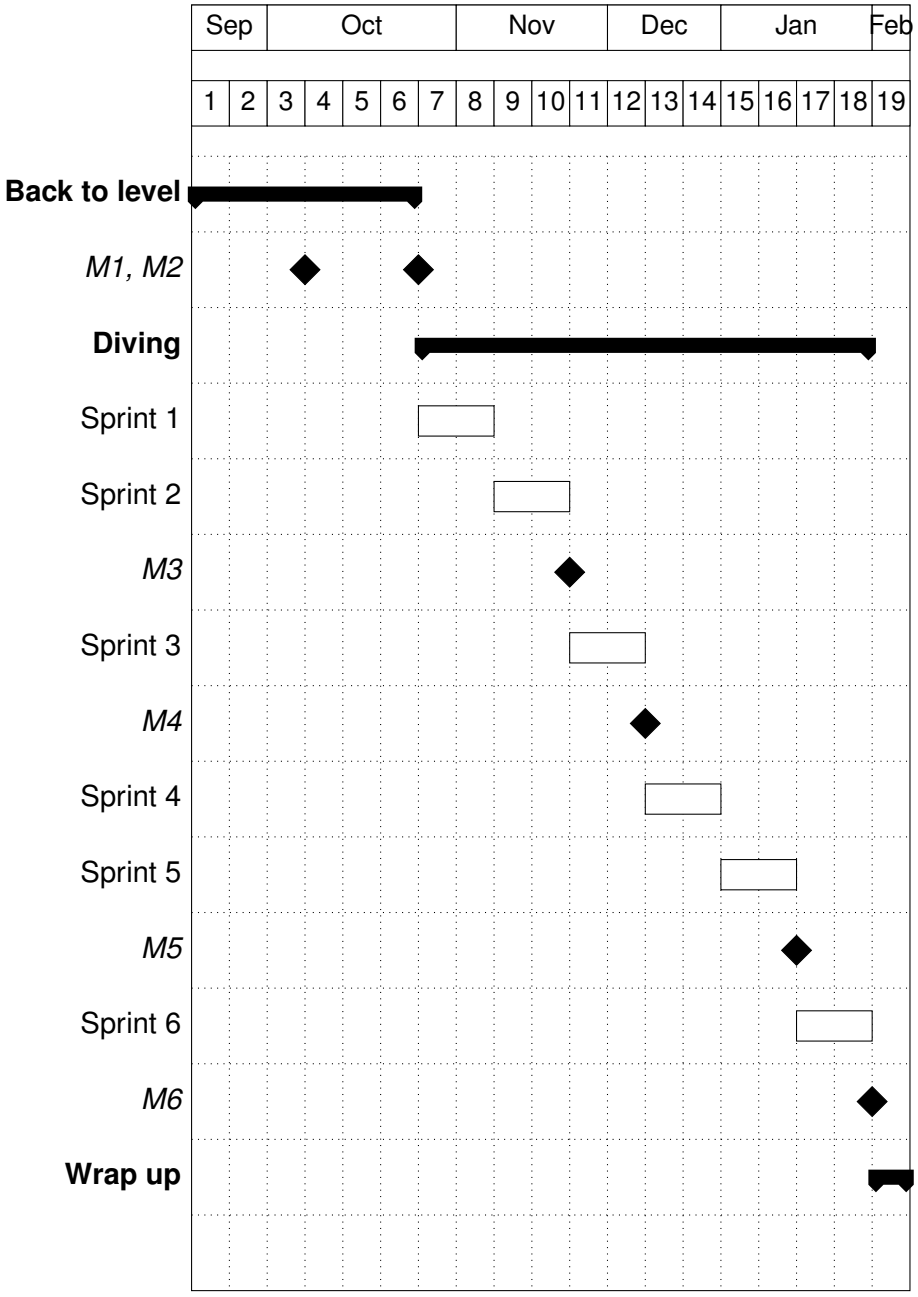


Figure 9.3: Initial Gantt Chart

Chapter 10

Discussion

10.0.1 Constatations

At the edge of technologies and sot we are only using techs release in late 2019

10.1 Convex Dataset

10.1.1 Data augmentation

Example:

Who is the author of the Harry Potter series? OR Who wrote Harry Potter? What was the year of publication for the first book? OR When was it first published? Title of the first book? OR The first book was called what? What country was the book set in? OR It was set in what country? Which book has the highest page count? OR What's the longest book?

10.1.2 Human errors

Mechanical Truck gathering mistakes during the dataset building due to humans not respecting the standard format. Implying 32 wrong question-answers for a single mistake. When did the first The Fast and the Furious film come out? TO "answer": `"https://www.wikidata.org/wiki/Q155476"` instead of 22 June 2001 however: "answer_text": "The first film came out 22 June 2001."

10.1.3 Data inconstancy

"question": "When was he born?", "answer_text": "1 August 1819" same as answer

Sometimes it is binary, and some times it is in NL

"question": "When was it published?", "answer_text": "The book came out 30 June 1997 in the UK."

This is an important inconstancy biaises for training.

10.1.4 Wrong answers

When did the first The Fast and the Furious film come out?

GraphQA answers 1955, which is the date of publication of the original The Fast and the Furious movie. And non of the competing qa systems answer correctly to the question, neither to the provided false answer, neither the correct one.

We didn't take time to go thru the whole dataset because time was missing, but funnily, GraphQA most often triggered warnings when the dataset had such errors, that's why we saw them.

This all implies that GraphQA, could even perform better than the concurrents, but more exhaustive evaluations on additional datasets are required. But in the current version GraphQA is very time and computing resources consuming, which made it hard to evaluated it on multiple datasets in parallel to development

10.1.5 Don't trust Mechanical Trucks

response format not respected `"https://www.wikidata.org/wiki/Q5951550?wprov=srpw1_0"` instead of `"https://www.wikidata.org/wiki/Q5951550"`

How tall is Avril Lavigne? -> (157 centimetre) additionally to the spelling mistake and a none standard format for the answer has the centimeter information is considered has a unit qualifier to the value 157, which must be the answer in this case, by checking the latest version of wikidata we couldn't find any spo containing the information.

10.2 What we learned from the project

10.2.1 Only trust yourself

Preprints: some good and mostly bad Published articles: some good, but mostly interative research with name dropping Published in conferences: some good, but be careful at where it is published, china is worrying Sadly everything looks alike with time Never trust what's written, always cross check the results and the given datasets or code if any Be critique with state-of-the-art and baseline clams as long as it was not reproduced.

10.3 What happens

Conversational, It tests each conversation with convex and graphqa as extension. Note that if no initial answer if found, no graph is built for platypus or qanswer, which skips the graph extension, as it's part of the nature of GraphQA and Convex

10.4 CONVEX

Terrible at many levels.

We knew we started that the intial answer part was bad, but the results shown were still proimising. At the end we noticed that it was bullshit, as we couldn't reproduce the results in the paper, and even after contacting the author we got the confirmation that the results where not reproducable has expected as the best part of the paper was written for motivatitional purposed, but we didn't understand it that way at first.

However, as a result we had no choice to go with it, and take the best of what we got so far, as it was too late to step back and take another approach by redesigning

the whole process. So we built from scratch a system fully based on sub-graphs, see chapter GRAPHQA.

Contacted the author and debugged his code together at first, then we decided to fork the project and continue on our own by fixing and adapting the code to our needs as GraphQA progressed.

10.5 Questions left

It would be interesting to see if the machine becomes sophist with our algorithm.

In another work, we would like to explore the possibilities of merging these two modules, and evaluate if it's pertinent.

Another question would be toward generalization, does generalisation needs to be only one and unique module able to do everything or can it be a clever composition of modules. Or does it need a specific module which is lightweight and able to generalise the tasks to send to the right module, as a coordinator.

Explore the possibilities to combine sub-graphs to keep context and history of the conversation and been able to connect facts from various subjects together and summarise them cleverly.

Say that sub-graphs could be pre generated to specific contextes, like news articles, and a module linking the initial question to a predefined sub-graph is possible.

Sub-graphs can also be generate manually by news authors by using keywords and sub-graphs linking, or automatically.

10.5.1 Subgraphs

Try to find a correlation between human thinking / reasoning the numerical knowledge graph representation, compared to IR, access structure, or even algorithms.

10.6 Convex

Results returned are often pure luck, as if multiple objects are present in for the same subject and predicate, convex will return the first one. Additionally, as the NER depends on TAGME, it is up to them to return the right entities for entities with the same name in the wikipedia. Same when CONVEX do a web search.
PROBABLY NOT Results are choatic, as randomness is possible during the IR operation done with TAGME, and particularly when retrive

Chapter 11

Conclusions

11.0.1 Project Management

11.1 Final words

Say that it looks like the more the concept or the model is simple, the best it is. It is a nice comparison at how nature works. The simplest survives the best.
Say that transformer are currently leading, their architecture is relatively simple, that's maybe why it's working so well. Say that with addition work, the multiple brains strategy / grounded tasks used by GraphQA can be also seen as the logical simplicity by breaking down difficult tasks into smaller tasks.
And the field of study should be explored further in NLP and combined with other fields like Machine Vision, or Sensory Robotics, to build a Multi-Domain Grounded Task Generation model. Setting new standard in machine reasoning and understanding with grounded symbolics.
Additionally, subgraphs are very close to how humans think, we believe that the field of knowledge graph must be explored even further.
Note that our work can later one be adapted to ML by training models to perform at similar tasks as ours. Additionally, we wanted to prove in a first step that the concept is working.
The approach used in this paper is to have analogically to human, a reasoning and the ability to talk. We are using multiple modules to accomplish this is a composite architecture.

Bibliography

BAHDANAU, Dzmitry; CHO, Kyunghyun; BENGIO, Yoshua, 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

BENAICH, Nathan; HOGARTH, Ian, 2019. State of AI 2019, pp. 126. Available also from: `\url{https://www.stateof.ai/}`.

BOJANOWSKI, Piotr; GRAVE, Edouard; JOULIN, Armand; MIKOLOV, Tomas, 2016. Enriching Word Vectors with Subword Information. *CoRR*. Vol. abs/1607.04606. Available from arXiv: `1607.04606`.

BOLLACKER, Kurt; EVANS, Colin; PARITOSH, Praveen; STURGE, Tim; TAYLOR, Jamie, 2008. Freebase: A collaboratively created graph database for structuring human knowledge. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 1247–1249. ISBN 9781605581026. ISSN 07308078. Available from DOI: `10.1145/1376616.1376746`.

BORDES, Antoine; USUNIER, Nicolas; CHOPRA, Sumit; WESTON, Jason, 2015. Large-scale Simple Question Answering with Memory Networks. *CoRR*. Vol. abs/1506.02075. Available from arXiv: `1506.02075`.

CHEN, Xinxiong; XU, Lei; LIU, Zhiyuan; SUN, Maosong; LUAN, Huan-Bo, 2015. Joint Learning of Character and Word Embeddings. In: *Joint Learning of Character and Word Embeddings*. *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pp. 1236–1242. Available also from: `http://ijcai.org/Abstract/15/178`.

CHOI, Eunsol; HE, He; IYYER, Mohit; YATSKAR, Mark; YIH, Wen-tau; CHOI, Yejin; LIANG, Percy; ZETTLEMOYER, Luke, 2018. QuAC : Question Answering in Context. *CoRR*. Vol. abs/1808.07036. Available from arXiv: `1808.07036`.

CHRISTMANN, Philipp; SAHA ROY, Rishiraj; ABUJABAL, Abdalghani; SINGH, Jyotsna; WEIKUM, Gerhard, 2019. Look before you Hop: Conversational Question Answering over Knowledge Graphs Using Judicious Context Expansion. *arXiv e-prints*, pp. arXiv:1910.03262. Available from arXiv: `1910.03262 [cs.IR]`.

CLARK, Christopher; GARDNER, Matt, 2017. Simple and Effective Multi-Paragraph Reading Comprehension. *CoRR*. Vol. abs/1710.10723. Available from arXiv: `1710.10723`.

DEVLIN, Jacob; CHANG, Ming-Wei; LEE, Kenton; TOUTANOVA, Kristina, 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. *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). Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. Available from DOI: 10.18653/v1/N19-1423.

DIEFENBACH, Dennis; TANON, Thomas Pellissier; SINGH, Kamal Deep; MARET, Pierre, 2017. Question Answering Benchmarks for Wikidata. In: *Question Answering Benchmarks for Wikidata*. *Proceedings of the ISWC 2017 Posters & Demonstrations and Industry Tracks co-located with 16th International Semantic Web Conference (ISWC 2017), Vienna, Austria, October 23rd - to - 25th, 2017.* Available also from: http://ceur-ws.org/Vol-1963/paper555.pdf.

DONG, Li; WEI, Furu; ZHOU, Ming; XU, Ke, 2015. Question answering over freebase with multi-column convolutional neural networks. *ACL-IJCNLP 2015 - 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference*. Vol. 1, pp. 260–269. ISBN 9781941643723.

DUNLOP, Michal Wallace & George, 1999. *Eliza, the Rogerian Therapist* [http://psych.fullerton.edu/mbirnbaum/psych101/Eliza.htm]. (Accessed on 10/09/2019).

ELOKENCE, Scimob, 2007. *Akinator* [https://en.wikipedia.org/wiki/Akinator]. (Accessed on 10/09/2019).

FERRAGINA, Paolo; SCAIELLA, Ugo, 2010. TAGME: on-the-fly annotation of short text fragments (by wikipedia entities). In: *TAGME: on-the-fly annotation of short text fragments (by wikipedia entities)*. *Proceedings of the 19th ACM International Conference on Conference on Information and Knowledge Management*. ACM, pp. 1625–1628. Available from DOI: 10.1145/1871437.1871689.

FUTURISM, LLC, 2016. *The History of Chatbots Infographic* [https://futurism.com/images/the-history-of-chatbots-infographic]. (Accessed on 10/09/2019).

GEHRMANN, Sebastian; STROBELT, Hendrik; RUSH, Alexander M., 2019. GLTR: Statistical Detection and Visualization of Generated Text. *CoRR*. Vol. abs/1906.04043. Available from arXiv: 1906.04043.

GITTENS, Alex; ACHLIOPTAS, Dimitris; MAHONEY, Michael W., 2017. Skip-Gram â^' Zipf + Uniform = Vector Additivity. In: *Skip-Gram â^' Zipf + Uniform = Vector Additivity*. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, pp. 69–76. Available from DOI: 10.18653/v1/P17-1007.

GRAEFE, Andreas; HAIM, Mario; HAARMANN, Bastian; BROSIUS, Hans-Bernd, 2018. Readers' perception of computer-generated news: Credibility, expertise, and readability. *Journalism*. Vol. 19, no. 5, pp. 595–610. Available from DOI: 10.1177/1464884916641269.

HAO, Yazhou; ZHENG, Qinghua; LAN, Yangyang; LI, Yufei; WANG, Meng; WANG, Sen; LI, Chen, 2017. Improving Chinese Sentiment Analysis via Segmentation-Based Representation Using Parallel CNN. In: CONG, Gao; PENG, Wen-Chih; ZHANG, Wei Emma; LI, Chengliang; SUN, Aixin (eds.). *Advanced Data Mining and Applications*. Cham: Springer International Publishing, pp. 668–680. ISBN 978-3-319-69179-4.

HERBELOT, Aurélie; BARONI, Marco, 2017. High-risk learning: acquiring new word vectors from tiny data. *CoRR*. Vol. abs/1707.06556. Available from arXiv: 1707.06556.

JU, Ying; ZHAO, Fubang; CHEN, Shijie; ZHENG, Bowen; YANG, Xuefeng; LIU, Yunfeng, 2019. Technical report on Conversational Question Answering. *arXiv e-prints*, pp. arXiv:1909.10772. Available from arXiv: 1909.10772 [cs.CL].

KARRAS, Tero; LAINE, Samuli; AITTALA, Miika; HELLSTEN, Janne; LEHTINEN, Jaakko; AILA, Timo, 2019. Analyzing and Improving the Image Quality of StyleGAN. *CoRR*. Vol. abs/1912.04958.

KELNAR, David, 2019. The State of AI, pp. 151. Available also from: https://www.stateofai2019.com/summary/.

KEON, 2017. *keon / awesome-nlp* [https://github.com/keon/awesome-nlp]. GitHub.

KIM, Yoon; JERNITE, Yacine; SONTAG, David A.; RUSH, Alexander M., 2015. Character-Aware Neural Language Models. *CoRR*. Vol. abs/1508.06615. Available from arXiv: 1508.06615.

KUMAR, Ankit; IRSOY, Ozan; SU, Jonathan; BRADBURY, James; ENGLISH, Robert; PIERCE, Brian; ONDRUSKA, Peter; GULRAJANI, Ishaan; SOCHER, Richard, 2015. Ask Me Anything: Dynamic Memory Networks for Natural Language Processing. *CoRR*. Vol. abs/1506.07285. Available from arXiv: 1506.07285.

KWIATKOWSKI, Tom et al., 2019. Natural Questions: a Benchmark for Question Answering Research. *Transactions of the Association of Computational Linguistics*.

LIU, Yinhan et al., 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*. Vol. abs/1907.11692. Available from arXiv: 1907.11692.

LOPEZ, Vanessa; UNGER, Christina; CIMIANO, Philipp; MOTTA, Enrico, 2013. Evaluating question answering over linked data. *Web Semantics Science Services And Agents On The World Wide Web*. Vol. 21, pp. 3–13. ISSN 1570-8268. Available from DOI: 10.1016/j.websem.2013.05.006.

LOWE, Ryan; POW, Nissan; SERBAN, Iulian; PINEAU, Joelle, 2015. The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems. *CoRR*. Vol. abs/1506.08909. Available from arXiv: 1506.08909.

LUCY, Li; GAUTHIER, Jon, 2017. Are distributional representations ready for the real world? Evaluating word vectors for grounded perceptual meaning. *CoRR*. Vol. abs/1705.11168. Available from arXiv: 1705.11168.

MA, Yukun; CAMBRIA, Erik; GAO, Sa, 2016. Label Embedding for Zero-shot Fine-grained Named Entity Typing. In: *Label Embedding for Zero-shot Fine-grained Named Entity Typing. Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan: The COLING 2016 Organizing Committee, pp. 171–180. Available also from: https://www.aclweb.org/anthology/C16-1017.

MIKOLOV, Tomas; CHEN, Kai; CORRADO, Greg; DEAN, Jeffrey, 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv e-prints*, pp. arXiv:1301.3781. Available from arXiv: 1301.3781 [cs.CL].

MILLER, Alexander H.; FISCH, Adam; DODGE, Jesse; KARIMI, Amir-Hossein; BORDES, Antoine; WESTON, Jason, 2016. Key-Value Memory Networks for Directly Reading Documents. *CoRR*. Vol. abs/1606.03126. Available from arXiv: 1606.03126.

PELLISSIER TANON, Thomas; ASSUNÇÃO, Marcos Dias de; CARON, Eddy; M. SUCHANEK, Fabian, 2018. Demoing Platypus – was A Multilingual Question Answering Platform for Wikidata. In: *Demoing Platypus – was A Multilingual Question Answering Platform for Wikidata*. *ESWC 2018 - Extended Semantic Web Conference*. Heraklion, Greece. hal-01824972.

PENNINGTON, Jeffrey; SOCHER, Richard; MANNING, Christopher D., 2014. GloVe: Global Vectors for Word Representation, pp. 1532–1543. Available also from: http://www.aclweb.org/anthology/D14-1162.

PETERS, Matthew E.; NEUMANN, Mark; IYYER, Mohit; GARDNER, Matt; CLARK, Christopher; LEE, Kenton; ZETTLEMOYER, Luke, 2018. Deep contextualized word representations. *CoRR*. Vol. abs/1802.05365. Available from arXiv: 1802.05365.

PINTER, Yuval; GUTHRIE, Robert; EISENSTEIN, Jacob, 2017. Mimicking Word Embeddings using Subword RNNs. *CoRR*. Vol. abs/1707.06961. Available from arXiv: 1707.06961.

RADFORD, Alec; WU, Jeffrey; CHILD, Rewon; LUAN, David; AMODEI, Dario; SUTSKEVER, Ilya, 2018. Language Models are Unsupervised Multitask Learners.

RAJPURKAR, Pranav; ZHANG, Jian; LOPYREV, Konstantin; LIANG, Percy, 2016. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. *CoRR*. Vol. abs/1606.05250. Available from arXiv: 1606.05250.

RAJPURKAR, Pranav; JIA, Robin; LIANG, Percy, 2018a. Know What You Don't Know: Unanswerable Questions for SQuAD. In: *Know What You Don't Know: Unanswerable Questions for SQuAD*. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Melbourne, Australia: Association for Computational Linguistics, pp. 784–789. Available from DOI: 10.18653/v1/P18-2124.

RAJPURKAR, Pranav; JIA, Robin; LIANG, Percy, 2018b. Know What You Don't Know: Unanswerable Questions for SQuAD. *CoRR*. Vol. abs/1806.03822. Available from arXiv: 1806.03822.

RAMACHANDRAN, Prajit; PARMAR, Niki; VASWANI, Ashish; BELLO, Irwan; LEVSKAYA, Anselm; SHLENS, Jonathon, 2019. Stand-Alone Self-Attention in Vision Models. *CoRR*. Vol. abs/1906.05909. Available from arXiv: 1906.05909.

REDDY, Siva; CHEN, Danqi; MANNING, Christopher D., 2018. CoQA: A Conversational Question Answering Challenge. *CoRR*. Vol. abs/1808.07042. Available from arXiv: 1808.07042.

ROWSELL-JONES, Andy; HOWARD, Chris, 2019. *2019 CIO Survey: CIOs Have Awoken to the Importance of AI* [https://www.gartner.com/en/documents/3897266/2019-cio-survey-cios-have-awoken-to-the-importance-of-ai]. (Accessed on 10/09/2019).

SANTOS, Cícero Nogueira dos; ZADROZNY, Bianca, 2014. Learning Character-level Representations for Part-of-Speech Tagging. In: *Learning Character-level Representations for Part-of-Speech Tagging. ICML*. JMLR.org, vol. 32, pp. 1818–1826. JMLR Workshop and Conference Proceedings. Available also from: http://dblp.uni-trier.de/db/conf/icml/icml2014.html#SantosZ14.

SAS, Blackbird, 2014. *HelloJam.fr* [https://www.hellojam.fr]. (Accessed on 10/09/2019).

SEO, Min Joon; KEMBHAVI, Aniruddha; FARHADI, Ali; HAJISHIRZI, Hannaneh, 2016. Bidirectional Attention Flow for Machine Comprehension. *CoRR*. Vol. abs/1611.01603. Available from arXiv: 1611.01603.

SEVERYN, Aliaksei; MOSCHITTI, Alessandro, 2016. Modeling Relational Information in Question-Answer Pairs with Convolutional Neural Networks. Available from arXiv: 1604.01178.

SHEN, Tao; GENG, Xiubo; QIN, Tao; GUO, Daya; TANG, Duyu; DUAN, Nan; LONG, Guodong; JIANG, Daxin, 2019. Multi-Task Learning for Conversational Question Answering over a Large-Scale Knowledge Base. *arXiv e-prints*, pp. arXiv:1910.05069. Available from arXiv: 1910.05069 [cs.CL].

SINGHAL, Amit, 2012. *Official Google Blog: Introducing the Knowledge Graph: things, not strings* [https://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html]. (Accessed on 10/09/2019).

TANG, Duyu; QIN, Bing; LIU, Ting, 2016. Aspect Level Sentiment Classification with Deep Memory Network. *CoRR*. Vol. abs/1605.08900. Available from arXiv: 1605.08900.

TRISCHLER, Adam; WANG, Tong; YUAN, Xingdi; HARRIS, Justin; SORDONI, Alessandro; BACHMAN, Philip; SULEMAN, Kaheer, 2016. NewsQA: A Machine Comprehension Dataset. *CoRR*. Vol. abs/1611.09830. Available from arXiv: 1611.09830.

TURING, A. M., 1950. Computing Machinery and Intelligence. *Mind*. Vol. 59, no. 236, pp. 433–460. ISSN 00264423. Available also from: http://www.jstor.org/stable/2251299.

VASWANI, Ashish; SHAZEER, Noam; PARMAR, Niki; USZKOREIT, Jakob; JONES, Llion; GOMEZ, Aidan N.; KAISER, Lukasz; POLOSUKHIN, Illia, 2017. Attention Is All You Need. *CoRR*. Vol. abs/1706.03762. Available from arXiv: 1706.03762.

WANG, Phil, 2019. *This Person Does Not Exist* [https://www.thispersondoesnotexist.com]. (Accessed on 10/09/2019).

WANG, Quan; HUANG, Pingping; WANG, Haifeng; DAI, Songtai; JIANG, Wenbin; LIU, Jing; LYU, Yajuan; ZHU, Yong; WU, Hua, 2019. CoKE: Contextualized Knowledge Graph Embedding. *arXiv e-prints*, pp. arXiv:1911.02168. Available from arXiv: 1911.02168 [cs.AI].

WESTON, Jason; CHOPRA, Sumit; BORDES, Antoine, 2015. Memory Networks. *CoRR*. Vol. abs/1410.3916.

YATSKAR, Mark, 2018. A Qualitative Comparison of CoQA, SQuAD 2.0 and QuAC. *CoRR*. Vol. abs/1809.10735. Available from arXiv: 1809.10735.

YIN, Wenpeng; KANN, Katharina; YU, Mo; SCHÜTZE, Hinrich, 2017. Comparative Study of CNN and RNN for Natural Language Processing. Available from arXiv: 1702.01923.

List of Figures

List of Tables

Appendix

.1 Worklog

.2 Jupyter Notebooks

.3 Spreadsheet

.4 Meeting Notes