Journal of the Brazilian Computer Society, 2024, 30:1, doi: 10.5753/jbcs.2024.2882
This work is licensed under a Creative Commons Attribution 4.0 International License.

Advancing Chatbot Conversations: A Review of Knowledge Update Approaches

Luis Antonio L. F. da Costa [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***lalfcosta@unisinos.br***]
Mateus Begnini Melchiades [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***mateusbme@unisinos.br***]
Valéria Soldera Girelli [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***vsgirelli@unisinos.br***]
Felipe Colombelli [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***colombelli@unisinos.br***]
Denis Andrei de Araújo [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***denandrei@unisinos.br***]
Sandro José Rigo [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***rigo@unisinos.br***]
Gabriel de Oliveira Ramos [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***gdoramos@unisinos.br***]
Cristiano André da Costa [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***cac@unisinos.br***]
Rodrigo da Rosa Righi [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***rrrighi@unisinos.br***]
Jorge Luis Victória Barbosa [**Universidade do Vale do Rio dos Sinos (UNISINOS)** | ***jbarbosa@unisinos.br***]

Applied Computing Department, Universidade do Vale do Rio dos Sinos (UNISINOS) Av. Unisinos, 950 – 93.022-750 – São Leopoldo – RS – Brazil

Received: 09 September 2022 • **Accepted:** 08 September 2023 • **Published:** 25 April 2024

Abstract Conversational systems like chatbots have emerged as powerful tools for automating interactive tasks traditionally confined to human involvement. Fundamental to chatbot functionality is their knowledge base, the foundation of their reasoning processes. A pivotal challenge resides in chatbots' innate incapacity to seamlessly integrate changes within their knowledge base, thereby hindering their ability to provide real-time responses. The increasing literature attention dedicated to effective knowledge base updates, which we term content update, underscores the significance of this topic. This work provides an overview of content update methodologies in the context of conversational agents. We delve into the state-of-the-art approaches for natural language understanding, such as language models and alike, which are essential for turning data into knowledge. Additionally, we discuss turning point strategies and primary resources, such as deep learning, which are crucial for supporting language models. As our principal contribution, we review and discuss the core techniques underpinning information extraction as well as knowledge base representation and update in the context of conversational agents.

Keywords: Chatbots, Natural Language Processing, Artificial Intelligence, Data Extraction.

1 Introduction

From well-known assistants like Amazon Alexa, Apple Siri, and Google Assistant to context-specific chatbot systems, more and more activities are being carried out with the help of intelligent applications. Such systems must be able to build a dialogue in human language to interact with people and answer their questions. In order to deal with the subtle details of the different languages, to capture the context of the interactions, and to correctly encode the knowledge, one needs to perform several complex steps through techniques assimilated from many Artificial Intelligence (AI) areas, like Machine Learning (ML), Deep Learning (DL), and Natural Language Processing (NLP) Reis *et al.* [2021]; Kacupaj *et al.* [2021].

The development of any chatbot application begins with data acquisition. Any intelligent agent needs data to learn from, and the raw information assembled often needs to be cleaned and pre-processed. Some techniques applied are document parsing, spell checking, Unicode normalization, special characters and digits removal, split sentences and words, and processing language-related structures. Depending on the context, the data might already be available (e.g., the users' purchase history) or might be acquired by identifying patterns on the requests arriving at the system Russel [2021]. However, data in its original format is not very useful for a chatbot system. Some NLP methods to improve the available data include using a public dataset similar to the context and task to be performed, collecting data from external sources, data augmentation (techniques that leverage language properties to create new text similar to the source data), and even a combination of datasets Vajjala *et al.* [2020].

The human-readable data needs to be described in terms of structures understandable by the system. The meaningful information can appear in the form of entities, relations, sentiments, intents, coreferences (multiple ways of mentioning the same entity), or relevance. Damerau [2010]. Considering for instance a situation in which we have data from the users' purchase history, we can identify the products bought as entities and store information like brand and price. However, some information like how two products relate to each other (e.g., toothbrush and toothpaste) might not be easily detected without deeper processing over the available data. If they are related, an intelligent chatbot designed for recommendation actions should suggest the toothpaste to clients who bought the toothbrush. Hence, we need a procedure that extracts the entities and the patterns that express the relation between the entities in a sentence, that is, a procedure that can extract knowledge into the intelligent assistant. This can be achieved through many different approaches, from heuristic-based models to powerful ML and DL methods Wang and Hao [2020].

The output of information and content extraction (entities and the relations between them, intents, sentiments, and so on) can be used to build Knowledge Bases (KB) Vajjala *et al*. [2020]. These resources represent acquired knowledge in a way that the conversational agent can make new inferences about the known data Levesque [1986]. The format in which the data will be stored depends on the goal. We can use simple resources like dictionaries and thesauruses or more sophisticated KBs that are able to store the relations between entities or even semantic relationships like synonyms, hyponyms, and meronyms. Some chatbot platforms can use huge KBs, such the ones employed in prominent platforms nowadays like Google and Bing Search API Microsoft [2024a]; Vajjala *et al*. [2020] to support queries at web-scale.

Despite the meaningful progress made by both industry and academia in the last decades, there are still gaps to be filled regarding the efficiency and capacity of chatbot systems. The inherent complexity of human language makes its understanding quite challenging. Ambiguity is a good example of this. The Winograd Schema Challenge Levesque *et al*. [2012] presents pairs of sentences that differ by a few words but have their meaning changed because of this difference. Consider the sentence: *"The man couldn't lift his son because he was so weak."*. *Who* was so weak? While many ambiguous sentences can be easily understood by a person, most NLP techniques applied in the development of chatbots are unable to solve them Zhao *et al*. [2021].

Another crucial detail in a conversation is the existence of common knowledge, the information assumed as known by most humans. The challenge lies in encoding all the human knowledge in a computational model in order to fully recognize any context involved. Humans are also capable of creating new knowledge and new ways of communicating, making it necessary for the intelligent system to understand such creativity. However, there is a lot of diversity among the languages, and an approach that addresses one of these aspects in a given language may not always work for other languages Pandya and Kalani [2021].

Although the available chatbot platforms can provide consistent features for their use in most different contexts, there is still a need to foster the actual technology by bringing support for some of the critical tasks. One of these tasks is the content update. Nowadays, available AI, KB, and NLP resources can provide chatbots with the possibility of automatically or semi-automatically include new content in their interactions and responses. However, most of the works focus only on integrity related updates, while only recent works have adopted approaches taking leverage of Linked Data characteristics, Knowledge Bases and Machine Learning resources Bagwan *et al*. [2021]; Ngai *et al*. [2021]; Heinzerling and Inui [2021].

Throughout this article, we explore the entire chatbot development pipeline, discussing the techniques and approaches that shape each of the topics mentioned so far. We highlight the gaps that remain to be filled and how promising approaches and current research trends aim to increase the chatbot's autonomy, from information extraction to knowledge base update.

The remainder of the document presents general aspects of chatbots, Information extraction, Knowledge representation, and known platforms and services to implement chatbots applications. Then are discussed current research trends regarding language models and knowledge-based resources applied to chatbots. Finally, we present the conclusion of the work.

2 Chatbots

Chatbots can be characterized into distinct models according to their capabilities and scope. For instance, chatbot systems aimed at answering Frequently Asked Questions (FAQ) usually work with a narrowed set of questions and responses, providing fixed responses to similar questions Vajjala *et al*. [2020]. Chatbots that follow a flow-based model, on the other side, are able to track the information on the conversation and build simple dialogues with the user, with the ability to generate more complex and variable answers. This model of chatbot asks specific and pre-defined questions to the user in order to complete the task (e.g., booking a flight). A more advanced level of chatbots, directed mostly for entertainment, are the open-ended agents. They can carry a conversation with the user about different topics without the necessity of keeping track of flows. Open-ended chatbots do not follow a specific template and are not restricted to a narrowed set of question-answer pairs.

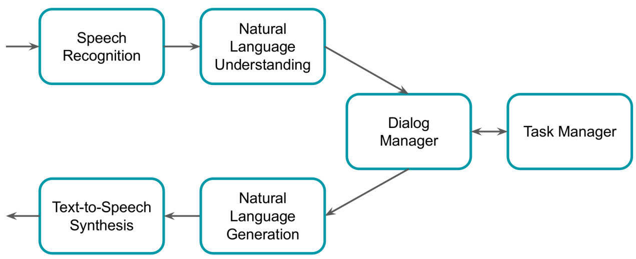

Figure 1. Pipeline of NLP tasks for building a dialogue system.

The aforementioned models can be classified into goal-oriented or conversational chatbots. The FAQ-directed and the flow-based chatbots fall into the goal-oriented category, a domain-specific system that requires domain-specific knowledge and focuses on accomplishing a goal. The open-ended systems are classified as conversational chatbots, where open-domain conversations without a specific goal play a part.

Despite the distinct purposes, the development of any dialogue system shares the employment of a set of NLP tasks, as depicted in Figure 1. *Speech recognition* is the task of transcribing human speech into text, working as an interface between the human and the intelligent application. However, many chatbots do not need to interact with humans through speech, only through text, which leads us to the next pipeline stage, the *natural language understanding* (NLU). The task of understanding the natural language is responsible for identifying and analyzing the objects present in the sentences, such as named entities, sentiments, and co-references. Some objects might be implicit in the text, such as sentiments, while others might appear explicitly, such as named entities.

Following the pipeline, next task is to build the modules in charge of controlling the dialogue and deciding the conversation flow. The *dialog and task managers* modules establish which segments of information are relevant or not based on the information extracted by the NLU module. This managing process can often be based on rules or on complex mechanisms such as reinforcement learning Vajjala *et al*. [2020].

Dialogue managers are common in goal-oriented conversational models since there is a well-defined objective in the conversation. Finally, the *natural language generation* step consists in building response in a human-readable fashion according to the decisions made by the dialog manager. The answer is then provided to the user through the *speech synthesis* module, which transforms the text-based information into speech, if required.

Behind every step above, several other minor tasks are happening. In contrast to simple question-answering systems, chatbots or dialogue systems must be able to understand the context and the nuances of the user's input, as the ambiguity and the language diversity mentioned previously. To do so, the NLU module needs to extract the user's intent and the entities related to it.

The intent is associated with the chatbot's domain of operation (e.g., an assistant that books a flight, finds a restaurant, and so on), and it helps the agent to decide the actions to be taken. The entity is the ontological construct that holds information regarding the objects related to the intent. Considering the example: *"Book me a flight to Berlin"*. The intent is *"book a flight"*, and the entity is *"Berlin"*, which is the information related to the intent.

In order to better understand the complexity involved in the development of a chatbot system, the next sections will delve into the details of some stages of the NLP pipeline. More specifically, we will discuss the state-of-the-art on knowledge extraction, knowledge representation, and knowledge base update.

3 Knowledge Extraction

Knowledge extraction involves identifying, organizing, and representing knowledge from various sources, including structured data sources such as databases, as well as unstructured sources such as text documents and multimedia content. The goal is to create a structured knowledge base that can be used for decision-making, problem-solving, and other applications. Knowledge extraction algorithms may use techniques such as data mining, machine learning, and semantic analysis to identify patterns, relationships, and concepts in the data Unbehauen *et al*. [2012].

One prominent challenge when working with NLP and Chatbots is the fact that information is often not readily available or properly structured for utilization. For instance, a bot hosted in an e-commerce website to help users navigate around can easily map a graph of where each page leads to. But what happens if the bot also had to retrieve a specific information about the product which happens to be contained inside an image advert from the product's description?

Challenges such as the one mentioned above inspired work in dynamically extracting content from websites. Such a task dispenses the need for manual extraction which, depending on the source and size, can be impractical at best and impossible in the worst scenarios. A very flexible implementation of website extraction is to parse the HTML page into a Document Object Model (DOM) Tree and apply certain heuristics to filter out parts unrelated to content like headers and sidebars Lou *et al*. [2013].

The literature provides a plethora of heuristics to the DOM Tree extraction approach. One such implementation uses the words-leaves ratio (WLR) to determine whether a subtree is relevant or not Insa *et al*. [2013]. The WLR can be summarized as the number of nodes that contain text divided by the number of leaves in a determined subtree. The aforementioned approach proved to be efficient at the time the paper was written; however, its performance is likely to have diminished nowadays due to the popularization of modern web frameworks that can lead to websites with much more complex structures. The authors' method assumes that the content is aggregated in a specific section of the page, but websites like social media often have dozens of nested HTML elements such as `<div>`s before reaching the content, as shown in Figure 2.

A more recent approach to DOM Tree content extraction proposed by Liu *et al.* Liu *et al*. [2017] analyses each node alongside with its neighboring nodes to acquire the main content. The authors also apply a Node Fusion mechanism that, based on the difference between two neighboring nodes, can merge similar blocks from the website, simplifying the analysis process and partially circumventing the complexity issue exemplified in Figure 2.

3.1 Information Extraction

Information extraction is the process of automatically extracting useful information from unstructured or semi-structured data sources, such as text documents, emails, or social media feeds. The goal is to identify relevant data and convert it into a structured format that can be used for further analysis. Information extraction algorithms typically use techniques such as natural language processing, pattern recognition, and machine learning to identify key entities, relationships, and events in the text.

Although being a concept with various similarities with knowledge extraction, information extraction focuses on extracting specific data from unstructured or semi-structured sources, while knowledge extraction is a broader process that involves extracting and organizing knowledge from various

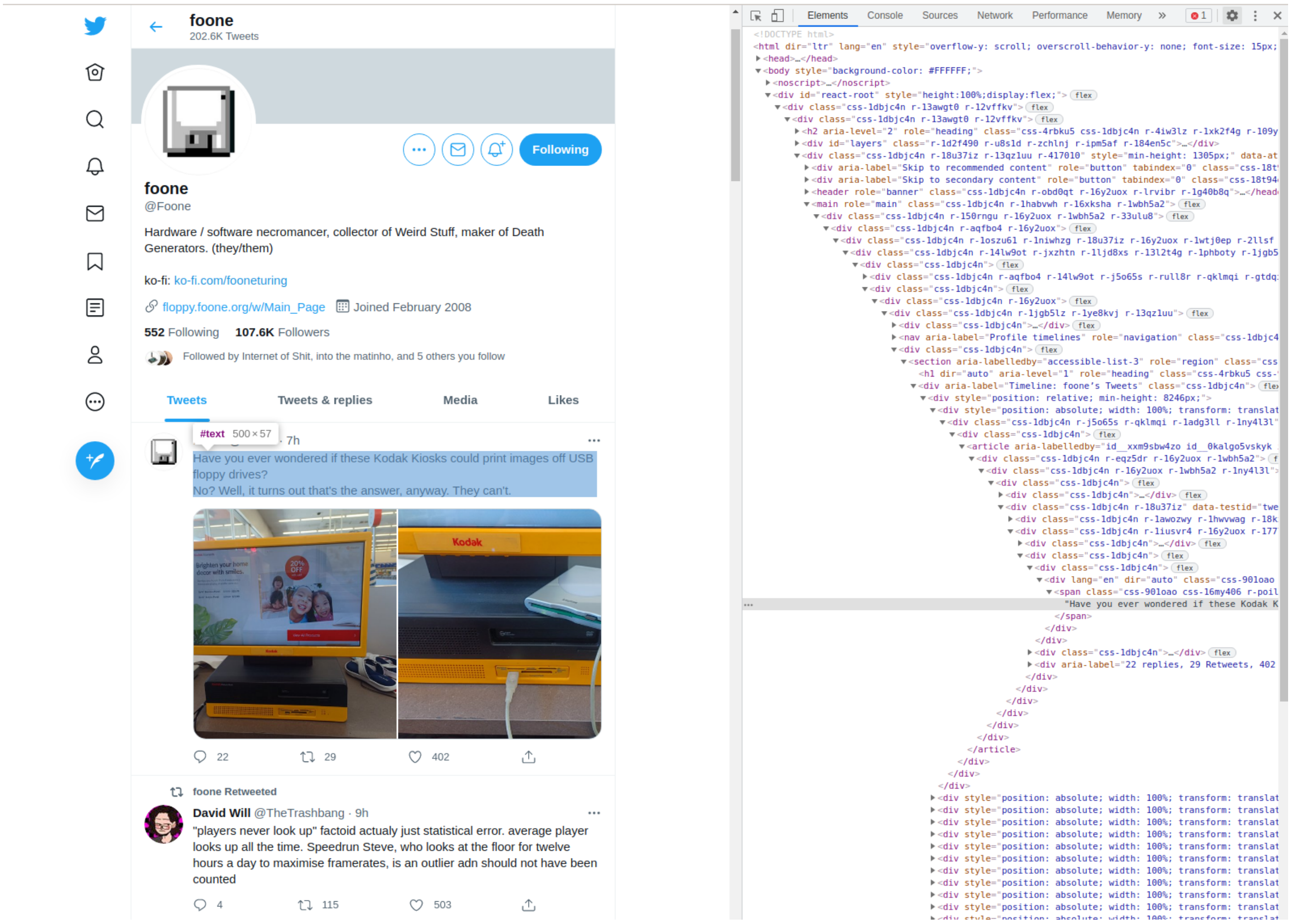

Figure 2. An example of highly segmented web page.

sources, including structured and unstructured data Ahmed and Pathan [2018].

Moreover, information extraction is a well known NLP task, which can be relevant for the context of chatbot content automatic update. The relevance of extracting useful information from text is not limited to NLP applications. UNESCO UNESCO [1975] has published back in 1975 guidelines for indexing documents and manually extracting keywords from them. The document proposed two main characteristics for determining relevant keywords inside a text: exhaustivity and specificity. By exhaustivity, the authors suggest that keywords should be able to identify all concepts inside a document that contains a possible informational value, thus not being restricted to some arbitrary limit regarding the number of terms selected. The latter characteristic determines that keywords should be as subject-specific as possible to represent all areas covered by the document accurately.

UNESCO acknowledges that the growth of information networks allows the collaboration of experts from multiple areas, therefore it is imperative that all fields present in a specific document are properly represented. For example, a technical document in a contentious Computer Science subject may reference social implications of its implementation, covering topics which belong to Social and Political Sciences. These topics could in the future be referenced by researchers in the area, so it is imperative that they are properly represented using the correct terms from the field instead of "Social Sciences".

In summary, keywords should be representative of both the document and the subjects it covers. Following the aforementioned guidelines will enable a reader to determine whether the document is relevant or not for their purposes UNESCO [1975].

3.2 Keyword Extraction in Recent Work

Keyword extraction applications are broadly used nowadays for generating metadata inside documents to provide insight for text analysis algorithms. By using the provided metadata inside a text document, a computer can easily process and extract essential information for content management tasks such as browsing, indexing, topic detection, content classification, and recommendation Firoozeh *et al.* [2020].

Prior to the turn of the millennium, attempts at intelligent keyword extraction centered attention towards supervised approaches Frank *et al.* [1999]; Tumey [1999]. However limited results led to more recent work using unsupervised strategies. One such work is TextRank Mihalcea and Tarau [2004], which uses an unsupervised graph-based ranking model for determining keywords inside a text. In summary, the word of a text is represented by a vertex on a graph. The connection to another vertex carries significance among these two words and casts a vote for it. After running the algorithm, each vertex is attributed a score representing its importance within the graph. Furthermore, the importance of the vertex casting the vote determines the weight of the vote itself. This approach shows superior performance to previously compared models. Moreover, TextRank does not require deep linguistic knowledge prior to utilization, which increases its portability to different applications.

Both supervised and unsupervised works mentioned above serve as basis for recent research in the topic Litvak and Last [2008]; Rossi *et al.* [2014]; Campos *et al.* [2020]; Vega-Oliveros *et al.* [2019]. One notable fact is that the most recent works found in the literature tend to use unsupervised graph-based approaches, highlighting TextRank's contribution to flexible keyword extraction applications.

4 Knowledge Representation

The advances observed in the Knowledge Representation field can support improvements in the way chatbots access and update the information they use to provide answers. When developing an intelligent mechanism to solve a given problem, we need to represent the extracted information in a way that Artificial Intelligence software modules can come to new conclusions regarding it Levesque [1986].

Extracted information can stored in knowledge bases (KB), which may be implemented using different types of data structures and rules. Knowledge Graphs (KG) are multi-relational graphs that use entities (shown as nodes) and the relations among them (shown as edges) to represent information. These relational facts are usually depicted in the form of triplets consisting of the head entity, the relation itself, and the tail entity. For instance, a KG with information about people and their nationality could contain the triplet *"(Carlos_Silva,Born-in,Brazil)"*, which indicates the relation ("Born-in") between the two entities ("Carlos_Silva" and "Brazil") and is able to answer the question *"Where is Carlos_Silva from?"*. KGs can also be applied for multi-hop question answering, in which the questions require passing through multiple edges of the graph in order to find an answer.

A common issue with KGs is how much of the world's information can be covered by them, since they are often incomplete, with many missing links and entities Dong *et al.* [2014]. Therefore, an important research field is the development of knowledge graph completion (KGC) methods. One common approach is KG embedding, which is the idea of embedding components of a KG such as entities and relations through a mathematical manipulation Wang *et al.* [2017]. The embeddings can later be used for several tasks other than KB completion, such as relation extraction, entity classification, and entity resolution. Previous works proposed different embedding approaches, such as Bordes *et al.* [2013]; Lin *et al.* [2015]; Rocktäschel *et al.* [2015]; Saxena *et al.* [2020]; Toutanova *et al.* [2016]; Wang *et al.* [2015, 2014a,b]; Wei *et al.* [2015]; Xie *et al.* [2016].

Knowledge Graphs are employed in several tools used ubiquitously nowadays. An example is the Google Knowledge Graph Singhal [2012], which is part of Google's search engine and also responsible for answering questions to conversational agents like Google Assistant. Google Knowledge Graph contains information gathered from other well-known KGs such as Freebase Bollacker *et al.* [2008], which was later incorporated into Google Wikidata Google [2014, 2019]. Other examples of large knowledge bases are DBpedia Lehmann *et al.* [2014] and NELL Mitchell *et al.* [2018].

One recent advance in knowledge representation is KE-

PLER Wang *et al*. [2021], which is able to take advantage of the efficiency in capturing factual knowledge from Knowledge Embedding (KE) methods for creating pre-trained language representation models (PLMs). The authors found that KEPLER is able to achieve state-of-the-art performance in KG link prediction and relation extraction tasks, thus helping minimize the aforementioned issues with Knowledge Graphs.

5 Knowledge Base Update

Regardless of the application, but especially crucial in NLP, maintaining a knowledge base is just as important as creating it. As new information becomes available, knowledge bases need to be updated to ensure that the information they contain is accurate and up-to-date. The work of Zhou and Li [2012] examines how existing knowledge base (i.e., knowledge breadth and depth) updates interacts with knowledge integration mechanisms (i.e., external market knowledge acquisition and internal knowledge sharing) to affect radical innovation. For example, a firm that is able to manage the updates on a broad knowledge base is more likely to achieve radical innovation in the presence of internal knowledge sharing rather than market knowledge acquisition.

In a similar manner, the more recent work of Garcia-Olano *et al*. [2021] highlights the importance of knowledge base updates in the context VQA (Visual Question Answering), where a dataset containing questions about images requires accurate and timely information and language and commonsense knowledge to provide a suitable answer.

The dynamic nature of human communication constantly exposes a knowledge base to new information which assume the form of colloquial and regional expressions, new global trends and topics, and more. Thus, it is imperative for maintainers to add these new user-generated patterns to their KBs or they might risk becoming stale. The task of keeping a knowledge base up-to-date can range from simple retraining procedures with a larger dataset to more complex algorithms that optimize and select the best features to prevent an exponential complexity growth, also called update semantics Vajjala *et al*. [2020].

Early work in the area of KBs mostly centered their attention towards the problem of *integrity checking* updates and repairing them Teniente and Olivé [1995]; Kowalski [1992]; Mayol *et al*. [1993]; Mayol and Teniente [1999]; Bry *et al*. [1992]; Olivé [1991]; Nicolas [1982]. With the popularization of Linked Data on the web Bizer *et al*. [2011], several authors have proposed methods based on currently popular knowledge querying language SPARQL Rinne [2012]; Horne *et al*. [2011] as well as knowledge-oriented databases like RDF Neumann and Weikum [2010]; Endris *et al*. [2015].

Knowledge base updates have important applications in various fields, including healthcare, finance, and e-commerce. In the healthcare domain, knowledge base updates can be used to improve clinical decision-making and patient outcomes, for example in the diagnosis and treatment of diabetes, where real-time updates to the knowledge base can improve the accuracy of diagnoses and treatment recommendations Abidi [2007]. For the finance domain, knowledge base updates can be used to improve risk management and decision-making, for example credit risk assessment, where real-time updates to the knowledge base can improve the accuracy of credit risk assessments Jabbari *et al*. [2019]. For instance, an example of knowledge base updates in e-commerce is the use of product catalogs that are updated in real-time to reflect changes in pricing, availability, and other product details, as for the importance of updating product catalogs in real-time to provide customers with accurate and timely information Xu *et al*. [2020].

There are different types of knowledge base updates, including adding new information, modifying existing information, and removing outdated information. A work from Sakama and Inoue [2003] introduces an abductive framework for updating knowledge bases represented by extended disjunctive programs by first providing a simple transformation from abductive programs to update programs which are logic programs specifying changes on abductive hypotheses. Then, extended abduction is introduced as a generalization of traditional abduction, and computed by the answer sets of update programs. Finally, different types of updates, view updates and theory updates are characterized by abductive programs and computed by update programs. The result of this paper provides a uniform framework for different types of knowledge base updates, and each update is computed using existing procedures of logic programming.

In Slota and Leite [2012], the authors introduce an abstract update framework based on viewing a knowledge base as the set of sets of models of its elements and performing updates by introducing additional interpretations to the sets of models of elements of the original knowledge base. This paper shows that the framework can also capture a wide range of both model and formula-based belief update operators which constitute the formal underpinning of existing approaches to ontology updates.

The work of Liang *et al*. [2017] investigated how to keep the freshness of a knowledge base by synchronizing it with its data source, (such as encyclopedia websites), and proposed a set of synchronization principles to build an Update System for knowledge Base (USB) with an update frequency predictor of entities as the core component. The authors designed a set of effective features and realize the predictor and conducted extensive experiments to justify the effectiveness of the proposed system, and finally, the USB was deployed on a Chinese knowledge base to improve its freshness.

A study from Nakashole and Weikum [2012] highlights the importance of updating knowledge bases in real-time to ensure that the information remains relevant and accurate. Nevertheless, despite using one of these mentioned approaches, updating knowledge bases can be challenging due to the volume and complexity of the information contained in the database, specially when conceptualizing cultural heritage information systems (CHIS), as detailed in the work of Kumar and Nair [2021].

6 Platforms, Tools, and Services

As previously mentioned, intelligent systems have become widely present nowadays, introducing the necessity to facilitate their development. Many powerful APIs and platforms have been made available by companies and open-source projects, providing the tools to build complex chatbots in a user-friendly manner. The techniques involved in the development pipeline (data pre-processing, natural language understanding, knowledge extraction and representation) are now transparent to the user, allowing everyone to assemble their own chatbot.

In this section we present some APIs for the development of conversational agents: LUIS by Microsoft Microsoft [2016], Watson Assistant by IBM IBM [2006], and Rasa ras [2024], an open-source platform. The platforms were chosen due to their popularity inside the community, while it is worth mentioning that many others are not included in this paper due to space limitations.

6.1 LUIS – Language Understanding

LUIS is a cloud-based machine learning service that facilitates the embedding of natural language processing into applications, bots, and IoT devices. Many available templates allow the use of LUIS on enterprise-grade conversational bots, chatbots for e-commerce scenarios (e.g., banking, entertainment, food), and for IoT system controlling through voice commands. LUIS is part of Microsoft Cognitive Services and is supported by the Azure infrastructure, easily integrated into any other Microsoft service. The architecture of an information chatbot built using LUIS is depicted in Figure 3, where many Azure services can be identified.

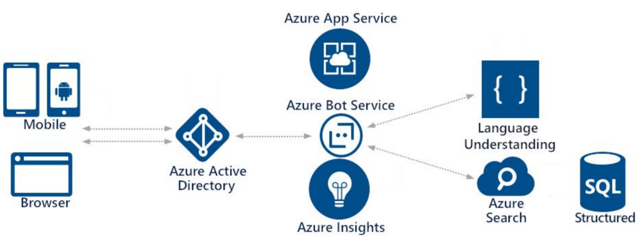

Figure 3. LUIS-based information chatbot.

The Bot Framework Microsoft [2024] is a complete collection of cross-platform command line tools that covers end-to-end bot development pipeline. Some of the available tools are the Bot Framework Emulator, which allows developers to test and debug their bots; the Bot Framework Composer, which is especially geared towards the development itself; and the many Bot Framework Samples, which illustrate examples of key aspects that need to be implemented.

LUIS employs several NLP techniques to provide natural language understanding over the data. Tokenization, speech tagging, segmentation, morphological analysis, translation, and many other methods assist on the language transformation. Once NLU remains a challenging problem given the inherent characteristics of human language, LUIS focuses on intent and entity recognition, identifying what the users want and what they are talking about in order to understand the user's intent and extract relevant information.

One crucial aspect behind the LUIS architecture is the QnA Maker Microsoft [2024b], the module responsible for building the knowledge base based on the available data. The content is imported into the KB and structured in question-answer pairs, extracting information as the entities and the relationship between them. The question-answer pairs include alternate forms for the questions, metadata tags to apply filters on the answer choices, and follow-up prompts to continue the search refinement. Thereby, QnA is able to find the most appropriate answer for a given input question.

LUIS has several features, amongst them, it has pre-built domains, which provides pre-built domains such as calendars, weather, and multimedia that can be customized and used to create language understanding models quickly; Integration with Azure Services, and as a part of the Microsoft Azure cloud platform, and it integrates seamlessly with other Azure services such as Bot Service and Cognitive Services; and Multilingual Support, which enables LUIS to supports multiple languages, including English, Spanish, French, and Chinese.

6.2 IBM Watson Assistant

Watson is an AI-based virtual agent powered by the IBM Cloud Services Qi *et al*. [2021]. Just as Microsoft, IBM also provides cross-platform tools to facilitate the chatbot development. The IBM Cloud API reference IBM [2021] helps in the initial implementation and guides the user through more complex steps to achieve a powerful assistant. Watson also provides pre-trained models that, together with a dialogue builder and content library, accelerate the chatbot development with no code required. Besides using the available knowledge, the IBM platform also looks for up-to-date answers in existing content bases.

Watson is explicitly proactive, suggesting the most relevant options to the user as soon as possible in the conversation. One key aspect is its active learning capabilities, which allow it to improve problem resolution and reduce frustration by asking users for the context in their questions. Its main work mechanism is to use NLU to analyze text and extract relevant information, including entities, keywords, and sentiment.

Watson features include pre-built models for a wide range of industries, including healthcare, finance, and retail. It also has integration with IBM Cloud, making Watson a part of the IBM Cloud platform that integrates with other IBM Cloud services, including Cloud Functions and Cloudant.

6.3 Rasa

Rasa in an open-source framework that embodies natural language understanding and conversation through a machine learning-based dialogue management. Its architecture is highly modular and customizable, employing state-of-the-art NLU research and techniques with the contribution of a global community with over six hundred developers and ten thousand forum members Rasa [2024a]. Instead of dealing with AI systems with hidden internal details, developers can have full access to the entire chatbot pipeline.

Rasa architecture can be observed in Figure 4. The Tracker Store is a module to store the assistant's conversations. The platform provides different storage types implementations,

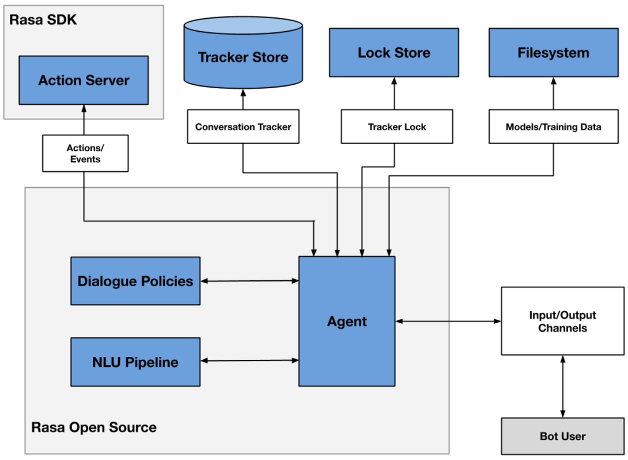

Figure 4. Rasa assistant architecture.

ranging from the default in-memory to SQL-based, Redis-based, Mongo-based, and other databases besides also allowing users to create their own, again demonstrating the flexibility the platform provides. Using a NLP/ML technique similar to Microsoft LUIS, Rasa uses intent recognition and entity recognition to understand the user's intent and extract relevant information.

Rasa also presents a set of methods to interact with APIs, knowledge bases, content management systems, and ready-to-deploy Docker containers, all through its Event Brokers Rasa [2024b]. As well as the other platforms, it can be easily integrated into well-known messaging tools, like Slack, Facebook, Google Home, IVR systems, and custom applications, and a single assistant can be deployed into multiple channels Rasa [2024b].Thereby, Rasa provides a highly customizable product with an agile development and deployment process that can be integrated with a wide range of tools and platforms, including chatbots, voice assistants, and messaging apps.

In opposition to the aforementioned platforms, Rasa allows users to use it simply as a standalone NLU service, easily trained, tested, and executed through command line. It also provides interactive learning, where one can generate new training data to the assistant by talking to it and providing feedback when it makes an error.

6.4 Chatbots' platforms characteristics

Table 1 summarizes the main characteristics of each chatbot platforms presented on this work. It is important to emphasize that all this chatbot plataform use Dialogue Management as one of their main working mechanisms to provide a flexible dialogue management system that allows developers to create custom conversation flows using a rule-based approach

Overall, Microsoft LUIS, IBM Watson Assistant, and Rasa all provide powerful NLP tools and services for developers. The choice of platform will depend on the specific requirements of the project, including the level of customization needed, the complexity of the conversation flows, and the availability of pre-built models and domains.

7 Current Research Trends

Besides strategies based on heuristics and Machine Learning, solutions based on Deep Learning (DL) have become popular in the last few years. Given the fact that language is inherently complex and unstructured, DL can be useful for executing NLP tasks. In this section, we discuss some state-of-the-art DL architectures that have been widely used to understand human language and its nuances. We initially highlight approaches based on Deep Learning used for constructing language models, which are applied successfully to several NLP tasks. We then present and discuss knowledge-based approaches, as they can provide complementary resources.

7.1 Language Model Resources

Recurrent Neural Networks (RNNs) are designed to process data in a progressive and sequential manner, with neural units capable of remembering the recent data processed and learned Karpathy [2015]; Vajjala *et al.* [2020]. Given the inherently sequential nature of language, with sentences flowing from one direction to another, an RNN model can progressively read and interpret any given input. One drawback of the technique is its forgetful memory, which prevents it from remembering the entire context of typical large sentences Vajjala *et al.* [2020]. Long short-term memory networks (LSTMs) aim to mitigate this constraint by discarding the part of the sentence that is irrelevant to understand the context and perform the NLP tasks Hochreiter and Schmidhuber [1997]; Olah [2015]; Vajjala *et al.* [2020]. LSTMs are responsible for some major advances for many NLP tasks Olah [2015]; Vajjala *et al.* [2020].

A recent but prominent technique in solving NLP tasks are transformers Vaswani *et al.* [2017]. Instead of analyzing the sentence sequentially like other approaches propose (e.g., RNNs), transformers look around each word individually to understand its context. This approach is known as self-attention, and it can better represent some of the language nuances mentioned earlier. To build its knowledge, the transformer is pre-trained with very large datasets in an unsupervised fashion, learning context and meanings from the large amount of information. The knowledge acquired can then be transferred to smaller NLP tasks through what is known as transfer learning Vajjala *et al.* [2020], on which the pre-trained model is fine-tuned into downstream NLP tasks such as named-entity recognition (NER), question answering, and text classification.

Some of the advantages transformers have against other DL techniques is its representation capability. The encoder receives a list of word embeddings whose size is the length of the longest sentence on the training dataset, which allows the model to extract context from the entire sentence, regardless of its size Vaswani *et al.* [2017]; Alammar [2018]. Besides that, the transformer also holds a multi-headed attention mechanism that allows it to focus on different positions to capture context and attention, which helps the interpretation of sentences like the aforementioned: *"The man couldn't lift his son because he was so weak"*. By focusing on both "the man" and "so weak", the mechanism can answer *who* was so weak. Therefore, transformers are able to more accurately

Table 1. Main Chatbot's Plataforms Techniques and Features

Platform	Main NLP/ML Techniques	Features
Microsoft LUIS	Intent and Entity Recognition, Dialog Management	Intent and Entity Recognition, Pre-built Domains, Integration with Azure Services, Multilingual Support
IBM Watson	Natural Language Understanding (NLU), Dialog Management, Machine Translation	Natural Language Understanding (NLU), Watson Assistant, Pre-built Models, Integration with IBM Cloud
Rasa	Intent Recognition, Entity Recognition, Dialogue Management	Open Source, Intent and Entity Recognition, Dialogue Management, Customizable

model the context and the subtle language peculiarities that might go unnoticed by other methods, enhancing their representation power.

Transformers also allow a higher level of parallelism comparing to convolutional and recurrent solutions, consequently reducing training time. In recurrent models, for instance, a given hidden state is a function of a previous hidden state, which sequentializes the training process and restrains the performance Vaswani *et al*. [2017]. On the transformer architecture, each word of the sentence flows on its own path in the encoder. The only existing dependency occurs in the self-attention layer, where the score of a word is calculated based on crossing each other word in the sentence against the word in question Alammar [2018].

A promising language representation model is BERT (Bidirectional Encoder Representations from Transformers) Devlin *et al*. [2019]. BERT is pre-trained with massive amounts of data and has improved the transformers original model by implementing a bidirectional pre-training of the data. While typical transformer models use unidirectional self-attention on which each token can only be related to the context on its left, in the BERT model the context can be extracted from both left and right directions in all layers Devlin *et al*. [2019]. This solution enhances the context incorporated and results in a pre-trained model that can be fine-tuned with a single additional output layer for many different tasks Devlin *et al*. [2019]. In that way, BERT is a general-purpose language model that can be tuned for specific tasks and domains.

Larger transformer models are able to provide much better performance for NLP applications. However, they face memory and training time constraints. Some works addressed these aspects through new parallelism levels Shoeybi *et al*. [2020] and through parameter-reduction techniques Lan *et al*. [2020], enhancing scalability and reducing training time without an equivalent loss in performance. Despite the improvements, these models are still too computationally taxing for use in mobile and real-time scenarios Reis *et al*. [2021], which is the case of many conversational agents and chatbots. A potential solution to the inference performance issue is Knowledge distillation (KD), which is a compression technique that transfers the knowledge of a huge model into a lighter representation, while diminishing the performance loss Tang *et al*. [2019]; Reis *et al*. [2021].

An increasingly popular resource in NLP are large language models (LMs) such as GPT-3 Brown *et al*. [2020], which contains billions of parameters and can be used for few-shot learning in various tasks. Despite their aptitude for solving numerous downstream tasks much better than the previous state-of-the-art, concerns have been raised about the consequences of training conversational models and question-answering systems on unfiltered content from the internet Bender *et al*. [2021]. In an effort to better handle toxic conversations and nonfactual answers, some authors are exploring using human feedback as labels in fine-tuning Ouyang *et al*. [2022]. It is evident, however, that factuality in LMs still remains an open issue in the field.

7.2 Knowledge-Based Resources

Despite the progress and the several approaches for context understanding, essentially after the transformers, state-of-the-art language models continue to explore only the statistical distribution of words in the training dataset without embodying their meaning Reis *et al*. [2021]. The outcome is the difficulty of reasoning over common knowledge since the implicit meaning of the sentences is missing. The inability of dealing with common sense might hinder the performance of conversational agents that are required to interact directly with humans. Several approaches to overcome this limitation have been developed recently, such as the integration of graph knowledge resources and even integrating language models and knowledge base resources.

In some works, knowledge graphs are applied within unsupervised pre-training given their explicit interpretation capabilities (by using triples) and contrasts with the interpretability of attention mechanisms Das *et al*. [2018]; Jain and Wallace [2019]; Reis *et al*. [2021]. However, most knowledge bases are often incomplete, as mentioned in the Section *Knowledge Representation and Update*, which further hampers the understanding of common sense. To increase both data diversity and volume, data augmentation methods can be applied to gather more information about common sense and existing relations Reis *et al*. [2021]; Yu *et al*. [2018]. Some researches are dedicated to identifying the best possible way to integrate external knowledge and make it available to common sense reasoning tasks. This is a critical task given the importance of having a knowledge graph that is correctly-aligned with the given task's objectives Bauer [2021].

The use of knowledge bases and knowledge graphs within the scope of chatbots allow for the dialogue planning to cope with more complex situations. These are the situations in which context elements are needed by the system to obtain the right answer. Some works, such as Kacupaj *et al*. [2021], innovate in this field by introducing the use of both graph knowledge and transformer-based resources.

The flexibility of pre-trained language models in natural language understanding and supporting activities such as question formulation is regarded as a important asset in the field. Some of the work found in the literature integrated these models as a possible alternative or complement to query operations in structured knowledge bases Danushka Bollegala and ichi Kawarabayashi [2021]. In a related initiative, embedding entities and relations from a knowledge graph has been experimented as a way to improve performance in predicting entity relations over large knowledge bases Heinzerling and Inui [2021].

In the past few years, knowledge bases have become more robust and harder to navigate Zaveri *et al*. [2016]. As a result, attempts at reducing their complexity and redundancy have emerged Tanon *et al*. [2020]; Wang and Hao [2020]; Xie *et al*. [2017]. One example is the latest major version of YAGO (version 4) which simplifies the taxonomy structure by merging instances of Wikidata, the most complex source, into the simpler structure of schema.org, which provides a great reduction in output complexity Tanon *et al*. [2020].

8 Concluding Remarks

In this paper, we presented the main aspects regarding conversational agents development and implementation. We aimed to highlight the prominence these systems are gaining in automating human-depending activities, their most important software platforms, and their main known challenges. Some of these challenges are related directly to the natural language ambiguity and diversity, preventing some interacting tasks from being automatized. This is the case of natural language understanding, information extraction, and knowledge integration.

Although these examples can carry many open questions, the literature also shows promising initiatives discussed in the paper. In particular, the paper discussed the state-of-the-art resources dedicated to fostering the possibilities of automatic content update. These alternatives can be observed in the case of language models based on NLP and DL resources, with the potential to support question-answering contexts. Complementarily, the knowledge base resources provide the chatbots with important new features, closing the gap in natural language and context understanding.

Declarations

Acknowledgements

This study was financed in part by the São Paulo Research Foundation (FAPESP) (Grant #2022/15816-5)[1]

Authors' Contributions

EV and JR contributed to the conception of this research approach, planned the experiments, and conducted the workshops. EV analyzed the results and is the writer of this manuscript. JR reviewed the manuscript. All authors read and approved the final manuscript.

[1]The opinions expressed in this work do not necessarily reflect those of the funding agencies.

Competing interests

The authors declare that they have no competing interests.

References

(2024). Rasa. Available at: `https://rasa.com/` Accessed in: Accessed in: 16th July 2021.

Abidi, S. S. R. (2007). Healthcare knowledge management: The art of the possible. In *K4CARE*. DOI: 10.1007/978-3-540-78624-5.

Ahmed, M. and Pathan, A.-S. K. (2018). *Data analytics: concepts, techniques, and applications*. Crc Press. Book.

Alammar, J. (2018). The illustrated transformer. Available at: `https://jalammar.github.io/illustrated-transformer/`Accesed in: 22th July 2021.

Bagwan, F., Phalnikar, R., and Desai, S. (2021). Artificially intelligent health chatbot using deep learning. In *2021 2nd International Conference for Emerging Technology (INCET)*, pages 1–5. DOI: 10.1109/INCET51464.2021.9456195.

Bauer, L. (2021). Identify, align, and integrate: Matching knowledge graphs to commonsense reasoning tasks. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, page 2259–2272, Online. Association for Computational Linguistics. DOI: 10.48550/arXiv.2104.10193.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623. DOI: 10.1145/3442188.3445922.

Bizer, C., Heath, T., and Berners-Lee, T. (2011). Linked data: The story so far. In *Semantic services, interoperability and web applications: emerging concepts*, pages 205–227. IGI global. DOI: 10.4018/978-1-60960-593-3.ch008.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, page 1247–1250, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/1376616.1376746.

Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., and Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 2787–2795, Red Hook, NY, USA. Curran Associates Inc. Available at:`https://proceedings.neurips.cc/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html`.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., *et al*. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901. Available at:`https://proceedings.neurips.cc/paper/`

2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

Bry, F., Manthey, R., and Martens, B. (1992). Integrity verification in knowledge bases. In *Logic Programming*, pages 114–139. Springer. DOI: 10.1007/3-540-55460-2$_9$.

Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., and Jatowt, A. (2020). Yake! keyword extraction from single documents using multiple local features. *Information Sciences*, 509:257–289. DOI: 10.1016/j.ins.2019.09.013.

Damerau, Fred J. Indurkhya, N. (2010). *Handbook of natural language processing*. Chapman & Hall/ CRC Press. DOI: 10.1201/9781420085938.

Danushka Bollegala, Huda Hakami, Y. Y. and ichi Kawarabayashi, K. (2021). Relwalk – a latent variable model approach to knowledge graph embedding. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, page 1551–1565, Online. Association for Computational Linguistics. DOI: 10.48550/arXiv.2101.10070.

Das, R., Munkhdalai, T., Yuan, X., Trischler, A., and McCallum, A. (2018). Building dynamic knowledge graphs from text using machine reading comprehension. DOI: 10.48550/arXiv.1810.05682.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., and Zhang, W. (2014). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 601–610, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/2623330.2623623.

Endris, K. M., Faisal, S., Orlandi, F., Auer, S., and Scerri, S. (2015). Interest-based rdf update propagation. In *International Semantic Web Conference*, pages 513–529. Springer. DOI: 10.1007/978-3-319-25007-6$_3$0.

Firoozeh, N., Nazarenko, A., Alizon, F., and Daille, B. (2020). Keyword extraction: Issues and methods. *Natural Language Engineering*, 26(3):259–291. Available at: https://www.cambridge.org/core/journals/natural-language-engineering/article/abs/keyword-extraction-issues-and-methods/84BFD5221E2CA86326E5430D03299711#article.

Frank, E., Paynter, G. W., Witten, I. H., Gutwin, C., and Nevill-Manning, C. G. (1999). Domain-specific keyphrase extraction. Available at: https://www.ijcai.org/Proceedings/99-2/Papers/002.pdf.

Garcia-Olano, D., Onoe, Y., and Ghosh, J. (2021). Improving and diagnosing knowledge-based visual question answering via entity enhanced knowledge injection. *Companion Proceedings of the Web Conference 2022*. DOI: 10.1145/3487553.3524648.

Google (2014). Google freebase. . Available at: https://developers.google.com/freebase Accessed in: 7th July 2021.

Google (2019). Google wikidata. https://www.wikidata.org/wiki/Wikidata:WikiProject_Freebase. Available at: https://www.wikidata.org/wiki/Wikidata:WikiProject_Freebase Accessed in: 7th July 2021.

Heinzerling, B. and Inui, K. (2021). Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, page 1772–1791, Online. Association for Computational Linguistics. DOI: 10.48550/arXiv.2008.09036.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9:1735–80. DOI: 10.1162/neco.1997.9.8.1735.

Horne, R., Sassone, V., and Gibbins, N. (2011). Operational semantics for sparql update. In *Joint International Semantic Technology Conference*, pages 242–257. Springer. DOI: 10.1007/978-3-642-29923-0$_1$6.

IBM (2006). Watson assistant. https://www.ibm.com/cloud/watson-assistant. Available at: https://cloud.ibm.com/apidocs/assistant-v2 Accesed in: 16th July 2021.

IBM (2021). Watson assistant v2 api reference. Available at: https://cloud.ibm.com/apidocs/assistant-v2 Accessed in: 20th July 2021.

Insa, D., Silva, J., and Tamarit, S. (2013). Using the words/leafs ratio in the dom tree for content extraction. *The Journal of Logic and Algebraic Programming*, 82(8):311–325. Automated Specification and Verification of Web Systems. DOI: 10.1016/j.jlap.2013.01.002.

Jabbari, A., Sauvage, O., and Cabioch, N. (2019). Towards a knowledge base of financial relations: Overview and project description. In *2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, pages 313–316. IEEE. DOI: 10.1109/AIKE.2019.00063.

Jain, S. and Wallace, B. C. (2019). Attention is not explanation. DOI: 10.48550/arXiv.1902.10186.

Kacupaj, E., Plepi, J., Singh, K., Harsh Thakkar, J. L., and Maleshkova, M. (2021). Conversational question answering over knowledge graphs with transformer and graph attention networks. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, page 850–862, Online. Association for Computational Linguistics. DOI: 10.48550/arXiv.2104.01569.

Karpathy, A. (2015). The unreasonable effectiveness of recurrent neural networks. http://karpathy.github.io/2015/05/21/rnn-effectiveness/. Available at: http://karpathy.github.io/2015/05/21/rnn-effectiveness/ Accessed in: 23th July 2021.

Kowalski, R. (1992). Database updates in the event calculus. *The Journal of Logic Programming*, 12(1-2):121–146. DOI: 10.1016/0743-1066(92)90041-Z.

Kumar, T. G. and Nair, R. R. (2021). Conserving knowledge heritage: opportunities and challenges in conceptualizing cultural heritage information system (chis) in the indian context. *Global Knowledge, Memory and Communication*.

DOI: 10.1108/GKMC-02-2021-0020.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2020). Albert: A lite bert for self-supervised learning of language representations. DOI: 10.48550/arXiv.1909.11942.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., and Bizer, C. (2014). Dbpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, 6.

Levesque, H. J. (1986). Knowledge representation and reasoning. *Annual review of computer science*, 1(1):255–287. Book.

Levesque, H. J., Davis, E., and Morgenstern, L. (2012). The winograd schema challenge. KR'12, page 552–561. AAAI Press. Available at: `https://cdn.aaai.org/ocs/4492/4492-21843-1-PB.pdf` Accessed in: July 2021.

Liang, J., Zhang, S., and Xiao, Y. (2017). How to keep a knowledge base synchronized with its encyclopedia source. In *International Joint Conference on Artificial Intelligence*. Available at: `https://www.ijcai.org/Proceedings/2017/0524.pdf` Accessed in: July 2021.

Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. (2015). Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2181–2187. AAAI Press. DOI: 10.1609/aaai.v29i1.9491.

Litvak, M. and Last, M. (2008). Graph-based keyword extraction for single-document summarization. In *Coling 2008: Proceedings of the workshop Multi-source Multilingual Information Extraction and Summarization*, pages 17–24. Available at: `https://aclanthology.org/W08-1404.pdf` Accessed in: July 2021.

Liu, Q., Shao, M., Wu, L., Zhao, G., Fan, G., and Li, J. (2017). Main content extraction from web pages based on node characteristics. *Journal of Computing Science and Engineering*, 11(2):39–48. DOI: 10.5626/JCSE.2017.11.2.39.

Lou, Y. F., Zhang, Y. C., and Yuan, Z. J. (2013). Website information extraction based on dom-model. In *Applied Mechanics and Materials*, volume 347, pages 2889–2893. Trans Tech Publ. DOI: 10.4028/www.scientific.net/AMM.347-350.2889.

Mayol, E. and Teniente, E. (1999). A survey of current methods for integrity constraint maintenance and view updating. In *International Conference on Conceptual Modeling*, pages 62–73. Springer. DOI: 10.1007/3-540-48054-4_6.

Mayol, E., Teniente, E., and Gargallo, P. (1993). Incorporating modification requests in updating consistent knowledge bases. In *DAISD*, pages 335–359. Citeseer. Available at: `https://upcommons.upc.edu/handle/2117/375255` Accessed in: July 2021.

Microsoft (2016). Language understanding (luis). Available at: `https://www.luis.ai/` Accessed in: Accessed in: 16th July 2021.

Microsoft (2024). Microsoft bot framework. Available at: `https://github.com/microsoft/botbuilder-tools` Accessed in: Accessed in: 19th July 2021.

Microsoft (2024a). Project answer search. Available at: `https://www.microsoft.com/en-us/research/project/answer-search/` Accessed in: Accessed in: 19th July 2021.

Microsoft (2024b). Qna maker. Available at: `https://docs.microsoft.com/en-us/azure/cognitive-services/QnAMaker/` Accessed in: Accessed in: 19th July 2021.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411. Available at: `https://aclanthology.org/W04-3252.pdf` Accessed in: Accessed in: July 2021.

Mitchell, T., Cohen, W., Hruschka, E., Talukdar, P., Yang, B., Betteridge, J., Carlson, A., Dalvi, B., Gardner, M., Kisiel, B., Krishnamurthy, J., Lao, N., Mazaitis, K., Mohamed, T., Nakashole, N., Platanios, E., Ritter, A., Samadi, M., Settles, B., Wang, R., Wijaya, D., Gupta, A., Chen, X., Saparov, A., Greaves, M., and Welling, J. (2018). Never-ending learning. *Commun. ACM*, 61(5):103–115. DOI: 10.1145/3191513.

Nakashole, N. and Weikum, G. (2012). Real-time population of knowledge bases: opportunities and challenges. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-Scale Knowledge Extraction (AKBC-WEKEX)*, pages 41–45. Available at: `https://aclanthology.org/W12-3008.pdf` Accessed in: Accessed in: July 2021.

Neumann, T. and Weikum, G. (2010). x-rdf-3x: Fast querying, high update rates, and consistency for rdf databases. *Proceedings of the VLDB Endowment*, 3(1-2):256–263. DOI: 10.14778/1920841.1920877.

Ngai, E. W., Lee, M. C., Luo, M., Chan, P. S., and Liang, T. (2021). An intelligent knowledge-based chatbot for customer service. *Electronic Commerce Research and Applications*, 50:101098. DOI: 10.1016/j.elerap.2021.101098.

Nicolas, J.-M. (1982). Logic for improving integrity checking in relational data bases. *Acta Informatica*, 18(3):227–253. DOI: 10.1007/BF00263192.

Olah, C. (2015). Understanding lstm networks. Available at: `https://colah.github.io/posts/2015-08-Understanding-LSTMs/` Accessed in: Accessed in: 23th July 2021.

Olivé, A. (1991). Integrity constraints checking in deductive databases. In *VLDB*, pages 513–523. Citeseer. Available at: `https://vldb.org/conf/1991/P513.PDF` Accessed in: Accessed in: July 2021.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., *et al.* (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744. Available at: `https://proceedings.neurips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html` Accessed in: Accessed in: July 2021.

Pandya, S. S. and Kalani, N. B. (2021). Preprocessing phase of text sequence generation for gujarati language. In *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, pages 749–752. DOI:

110.1109/ICCMC51019.2021.9418046.

Qi, H., Pan, L., Sood, A., Shah, A., Kunc, L., Yu, M., and Potdar, S. (2021). Benchmarking commercial intent detection services with practice-driven evaluations. *CoRR*, abs/2012.03929. DOI: 10.18653/v1/2021.naacl-industry.38.

Rasa (2024a). The future of conversational ai is open. Available at: https://rasa.com/open-source/ Accessed in: Accessed in: 20th July 2021.

Rasa (2024b). Let your business start a conversation. Available at: https://rasa.com/product/features/ Accessed in: Accessed in: 20th July 2021.

Reis, E. S. D., Costa, C. A. D., Silveira, D. E. D., Bavaresco, R. S., Righi, R. D. R., Barbosa, J. L. V., Antunes, R. S., Gomes, M. M., and Federizzi, G. (2021). Transformers aftermath: Current research and rising trends. *Commun. ACM*, 64(4):154–163. DOI: 10.1145/3430937.

Rinne, M. (2012). Sparql update for complex event processing. In *International Semantic Web Conference*, pages 453–456. Springer. DOI: 10.1007/978-3-642-35173-$0_3$8.

Rocktäschel, T., Singh, S., and Riedel, S. (2015). Injecting logical background knowledge into embeddings for relation extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1119–1129, Denver, Colorado. Association for Computational Linguistics. DOI: 10.3115/v1/N15-1118.

Rossi, R. G., Marcacini, R. M., and Rezende, S. O. (2014). Analysis of domain independent statistical keyword extraction methods for incremental clustering. *Learning and Nonlinear Models*, 12(1):17–37. Available at: https://www.researchgate.net/profile/Ricardo-Marcacini-2/publication/307782161_Analysis_of_Domain_Independent_Statistical_Keyword_Extraction_Methods_for_Incremental_Clustering/links/57e03cc108aece48e9e1f1e5/Analysis-of-Domain-Independent-Statistical-Keyword-Extraction-Methods-for-Incremental-Clustering.pdf Accessed in: Accessed in: July 2021.

Russel, Stuart J.; Norvig, P. (2021). *Artificial Intelligence - A Modern Approach*. Pearson Education. Book.

Sakama, C. and Inoue, K. (2003). An abductive framework for computing knowledge base updates. *Theory and Practice of Logic Programming*, 3:671 – 715. Available at: https://api.semanticscholar.org/CorpusID:3266512 Accessed in: Accessed in: July 2021.

Saxena, A., Tripathi, A., and Talukdar, P. (2020). Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics. DOI: 10.18653/v1/2020.acl-main.412.

Shoeybi, M., Patwary, M., Puri, R., LeGresley, P., Casper, J., and Catanzaro, B. (2020). Megatron-lm: Training multi-billion parameter language models using model parallelism. DOI: 10.48550/arXiv.1909.08053.

Singhal, A. (2012). Introducing the knowledge graph: things, not strings. Available at: https://blog.google/products/search/introducing-knowledge-graph-things-not/ Accessed in: Accessed in: 14th July 2021.

Slota, M. and Leite, J. (2012). A unifying perspective on knowledge updates. In *European Conference on Logics in Artificial Intelligence*. DOI: 10.1007/978-3-642-33353-$8_2$9.

Tang, R., Lu, Y., Liu, L., Mou, L., Vechtomova, O., and Lin, J. (2019). Distilling task-specific knowledge from bert into simple neural networks. DOI: 10.48550/arXiv.1903.12136.

Tanon, T. P., Weikum, G., and Suchanek, F. (2020). Yago 4: A reason-able knowledge base. In *European Semantic Web Conference*, pages 583–596. Springer. DOI: 10.1007/978-3-030-49461-2.

Teniente, E. and Olivé, A. (1995). Updating knowledge bases while maintaining their consistency. *The VLDB Journal*, 4(2):193–241. DOI: 10.1007/BF01237920.

Toutanova, K., Lin, V., Yih, W.-t., Poon, H., and Quirk, C. (2016). Compositional learning of embeddings for relation paths in knowledge base and text. pages 1434–1444. DOI: 10.18653/v1/P16-1136.

Tumey, P. D. (1999). Learning to extract keyphrases from text. *NRC Technical Report ERB-l 057. National Research Council, Canada*, pages 1–43. DOI: 10.48550/arXiv.cs/0212013.

Unbehauen, J., Hellmann, S., Auer, S., and Stadler, C. (2012). Knowledge extraction from structured sources. *Search Computing: Broadening Web Search*, pages 34–52. DOI: 10.1007/978-3-642-34213-4_3.

UNESCO (1975). Unisist - indexing principles. Book.

Vajjala, S., Majumder, B., Gupta, A., and Surana, H. (2020). *Practical Natural Language Processing: A Comprehensive Guide to Building Real-world NLP Systems*. O Reilly Media. Book.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. DOI: 10.48550/arXiv.1706.03762.

Vega-Oliveros, D. A., Gomes, P. S., Milios, E. E., and Berton, L. (2019). A multi-centrality index for graph-based keyword extraction. *Information Processing & Management*, 56(6):102063. DOI: 10.1016/j.ipm.2019.102063.

Wang, Q. and Hao, Y. (2020). Alstm: An attention-based long short-term memory framework for knowledge base reasoning. *Neurocomputing*, 399:342–351. DOI: 10.1016/j.neucom.2020.02.065.

Wang, Q., Mao, Z., Wang, B., and Guo, L. (2017). Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743. DOI: 10.1109/TKDE.2017.2754499.

Wang, Q., Wang, B., and Guo, L. (2015). Knowledge base completion using embeddings and rules. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 1859–1865. AAAI Press. Available at:https://www.ijcai.org/Proceedings/15/Papers/264.pdf.

Wang, X., Gao, T., Zhu, Z., Zhang, Z., Liu, Z., Li, J., and Tang, J. (2021). Kepler: A unified model for knowledge embedding and pre-trained language representation.

Transactions of the Association for Computational Linguistics, 9:176–194. DOI: 10.48550/arXiv.1911.0613.

Wang, Z., Zhang, J., Feng, J., and Chen, Z. (2014a). Knowledge graph and text jointly embedding. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1591–1601, Doha, Qatar. Association for Computational Linguistics. DOI: 10.3115/v1/D14-1167.

Wang, Z., Zhang, J., Feng, J., and Chen, Z. (2014b). Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, page 1112–1119. AAAI Press. DOI: 10.1609/aaai.v28i1.8870.

Wei, Z., Zhao, J., Liu, K., Qi, Z., Sun, Z., and Tian, G. (2015). Large-scale knowledge base completion: Inferring via grounding network sampling over selected instances. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, page 1331–1340, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/2806416.2806513.

Xie, Q., Ma, X., Dai, Z., and Hovy, E. (2017). An interpretable knowledge transfer model for knowledge base completion. *arXiv preprint arXiv:1704.05908*. DOI: 10.48550/arXiv.1704.05908.

Xie, R., Liu, Z., Jia, J., Luan, H., and Sun, M. (2016). Representation learning of knowledge graphs with entity descriptions. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, page 2659–2665. AAAI Press. DOI: 10.1609/aaai.v30i1.10329.

Xu, F., Pan, Z., and Xia, R. (2020). E-commerce product review sentiment classification based on a naïve bayes continuous learning framework. *Information Processing & Management*, 57(5):102221. DOI: 10.1016/j.ipm.2020.102221.

Yu, A. W., Dohan, D., Luong, M.-T., Zhao, R., Chen, K., Norouzi, M., and Le, Q. V. (2018). Qanet: Combining local convolution with global self-attention for reading comprehension. DOI: 10.48550/arXiv.1804.09541.

Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., and Auer, S. (2016). Quality assessment for linked data: A survey. *Semantic Web*, 7(1):63–93. DOI: 10.3233/SW-150175.

Zhao, L., Alhoshan, W., Ferrari, A., Letsholo, K. J., Ajagbe, M. A., Chioasca, E.-V., and Batista-Navarro, R. T. (2021). Natural language processing for requirements engineering: A systematic mapping study. *ACM Computing Surveys (CSUR)*, 54(3):1–41. DOI: 10.1145/3444689.

Zhou, K. Z. and Li, C. B. (2012). How knowledge affects radical innovation: Knowledge base, market knowledge acquisition, and internal knowledge sharing. *Southern Medical Journal*, 33:1090–1102. DOI: 10.1002/smj.1959.