Received: 21 December 2023 | Revised: 4 June 2024 | Accepted: 31 July 2024

DOI: 10.1111/exsy.13712

REVIEW

Expert Systems WILEY

Intent detection for task-oriented conversational agents: A comparative study of recurrent neural networks and transformer models

Mourad Jbene[1] | Abdellah Chehri[2] | Rachid Saadane[1] | Smail Tigani[3] | Gwanggil Jeon[4]

[1]SIRC-LaGeS, Hassania School of Public Works, Casablanca, Morocco

[2]Department of Mathematics and Computer Science, Royal Military College of Canada, Kingston, Ontario, Canada

[3]AAIR Lab, Digital Engineering and Artificial Intelligence Systems High Private School, Casablanca, Morocco

[4]Department of Embedded Systems Engineering, College of Information Technology, Incheon National University, Incheon, Korea

Correspondence
Abdellah Chehri, Department of Mathematics and Computer Science, Royal Military College of Canada, Kingston, Ontario, Canada.
Email: chehri@rmc.ca

Abstract

Conversational assistants (CAs) and Task-oriented ones, in particular, are designed to interact with users in a natural language manner, assisting them in completing specific tasks or providing relevant information. These systems employ advanced natural language understanding (NLU) and dialogue management techniques to comprehend user inputs, infer their intentions, and generate appropriate responses or actions. Over time, the CAs have gradually diversified to today touch various fields such as e-commerce, healthcare, tourism, fashion, travel, and many other sectors. NLU is fundamental in the natural language processing (NLP) field. Identifying user intents from natural language utterances is a sub-task of NLU that is crucial for conversational systems. The diversity in user utterances makes intent detection (ID) even a challenging problem. Recently, with the emergence of Deep Neural Networks. New State of the Art (SOA) results have been achieved for different NLP tasks. Recurrent neural networks (RNNs) and Transformer architectures are two major players in those improvements. RNNs have significantly contributed to sequence modelling across various application areas. Conversely, Transformer models represent a newer architecture leveraging attention mechanisms, extensive training data sets, and computational power. This review paper begins with a detailed exploration of RNN and Transformer models. Subsequently, it conducts a comparative analysis of their performance in intent recognition for Task-oriented (CAs). Finally, it concludes by addressing the main challenges and outlining future research directions.

KEYWORDS
conversational systems, intent detection, natural language understanding, recurrent neural networks, transformer models

1 | INTRODUCTION

Personalization is a research subject that dates back to the late 1990s and is considered to be one of the more established research fields. The concept of personalization, which is also known in a broader sense as customization, refers to the act of modifying a service or a product in such a way that it fits to the preferences, cognition, requirements, or capabilities of specific persons within the confines of a specific setting.

A bot is a virtual agent that uses more or less advanced artificial intelligence to communicate with a user on a given domain. Human-machine communication must be as natural as possible to approach human touch as closely as possible. The assistant can recognize your request, whether written or oral and then provide you with a suitable answer witch could also be personalized based on the users' profile of interactions. In addition, the robot can be represented by a totemic character or an avatar so that users feel accompanied at best in their experience.

We can specify several types of assistants, depending on their task. The chatbots respond in writing to user requests. Callbots, on the other hand, answer phone calls. Finally, voice assistants, or conversational agents, called also voice bots like Siri or Alexa, are dedicated to oral communication. They are a powerful tool to free up time for sellers and agents dedicated to customer relations on redundant aspects of their daily lives. But even more, there are many prominent areas where virtual personalization has gained traction, including virtual assistants on devices such as Alexa, Siri, Cortana, chat-bots, and online suggestions for e-commerce and entertainment.

In both verbal and written interactions, the function of communicative labour can be automated through the use of conversational agents, often known as chatbots. The widespread availability of voice assistant chatbots like Siri, Alexa, Cortana, and Google Assistant, as well as the numerous chatbot features in online retail, have made the utility, interaction, and operation of a chatbot more familiar to a large portion of modern society. For example, chatbots are used in industrial settings to provide information, instructions, detect fatigue, and address exceptions.

1. **Siri:** In 2011, Apple offered an assistant named Siri to its devices (computers, tablets, and even smartphones). Innovation was born since the assistants offered until then were only accessible on computers. When it started, Siri could answer basic questions given by users. Nowadays, he is recognized as a real virtual assistant since he is recognized for advanced human properties such as humour and the answer to more complex questions.
2. **Google Assistant:** Formerly Google Now, this conversational assistant, presented in the form of a downloadable application, allows not only to answer the questions asked by users but also to provide personalized recommendations to them. It is based on speech recognition, synthesis, and automatic natural language processing (NLP). It, therefore, includes both written and oral requests.
3. **Alexa:** Launched by the giant Amazon, Alexa is not content to be only an application, but it also appears in the form of a connected speaker. In addition to multilingual voice communication (English, French, Spanish, German, and Japanese) and distinguishing several voice profiles, Alexa can play a piece of music, set alarms, serve as a calculator and provide information such as weather or traffic. Finally, she can also control some so-called intelligent devices inside the house.
4. **Cortana:** Cortana, the Windows personal assistant, offers, just like Siri and Alexa, to answer users' oral questions. It is based on the Bing search engine, specifically Microsoft, and on the data provided within the users' smartphones.

Conversational agents, commonly referred to as conversational assistants (CAs), are computer systems designed to emulate human conversation through various communication channels. These channels include speech, text, facial expressions, and gestures (Laranjo et al., 2018). Task-oriented dialogue systems, in particular, are conversational agents that aim to assist users in accomplishing a specific task. For example, they could help users make a restaurant reservation, (book a flight, recommend a movie to watch, etc.) through dialogue in natural language, either in spoken or written form. These systems have been subject to growing interest in the last decade with applications in various industrial sectors. For instance, travel bots facilitate hotel and flight booking in the tourism industry. In contrast, chatbots support tasks such as checking account balances, money transfers, and bill payments in the banking industry.

Dialogue systems are built-in generally as a pipeline of components (Iovine et al., 2020). The user message, also called utterance or act of speech, is first analysed by the NLU feature, which interprets the user's needs. Then the Dialogue State Tracker (DST) module remembers all the information that was exchanged between the user and the system. It also updates its dialogue based on the user's message and the previous form. A dialogue policy manager block is then used to choose the action that will be performed by the system based on the current input and the state of the conversation. Finally, the natural language generation (NLG) component produces the actual response to the user.

The NLU component is the first step in building effective dialogue systems. The two main key problems in NLU are identifying the user intention (ID) and extracting attribute values i.e Slot-Filling (SF) from the user utterance (Bhathiya & Thayasivam, 2020). The present paper centres on the task of ID. The task at hand involves the identification of user input through a process known as text classification. This process entails categorizing the user's input into one or more intent categories that have been predefined. For instance, in the context of conversational recommendation scenarios, it is observed that users often have specific goals in mind when engaging with the system. These goals typically revolve around expressing their preferences or requesting recommendations (Epure et al., 2018).

Two widely used architectures that were used for ID in the last decade are RNN-based and Transformer-based models. RNNs are a popular category of neural network models successfully utilized for various sequential data (text streams, audio clips, video clips, time-series data, etc.) modelling. For ID, in particular, RNNs have proven a good performance on multiple benchmark data sets. On the other hand, Transformer models (Vaswani et al., 2017) are pretty new architectures that were proposed so that they can be trained on massive data sets and benefit from parallelism. As a result, transformer models achieved state-of-the-art results in various NLP tasks. Moreover, recent studies demonstrate their competitive performance against RNN-based models for ID. Thus, it is an excellent opportunity to review recent works and to understand the latest progress, challenges, and frontiers for ID.

Briefly, this paper's significant contributions are summarized below:

1. An overview of RNNs, and Transformer models, including historical advancements, types, and variants.
2. Presentation of the main RNN and Transformer-based models proposed for the ID task.
3. Comparative analysis of different models on widely used data sets, encompassing performance and efficiency evaluations.
4. Summarization of the main challenges faced by current ID models, along with proposed avenues for future research.

The rest of the survey is organized as follows. Section 2 presents the theoretical background of RNNs and commonly used variants, including GRUs and LSTMs. Section 3 describes the building blocks of Transformer models. A review of recent studies about ID using RNNs and Transformers is discussed in Section 4. Section 5 presents a comparative analysis of different architectures on widely used NLU benchmark data sets. Finally, conclusion and feature works are given in Section 6.

2 | RECURRENT NEURAL NETWORKS

Recurrent versus feed-forward: A neural network can be defined as a complex interconnected system composed of individual nodes, wherein each node is responsible for receiving input signals and generating corresponding output signals. The connectivity pattern within a neural network is determined by its architectural design, which governs the interconnections between individual nodes or neurons. In the architecture of a feed-forward neural network (FFNNs), it is observed that the outputs of each node exclusively influence the nodes located in the subsequent layers. FFNNs encompass a variety of neural network architectures that are widely employed in various domains. Prominent examples of FFNNs include multi-layered perceptrons (MLPs), Convolutional Neural Networks (CNNs), and Transformers (Vaswani et al., 2017). These architectures have demonstrated remarkable success in tasks such as image classification, NLP, and sequence modelling. MLPs, the most basic form of FFNNs, consist of multiple layers of interconnected neurons, where information flows in a unidirectional manner from the input layer to the output layer. On the other hand, for recurrent neural networks (RNNs), a node's output depends on its inputs' entire history, which results in a temporal dynamic behaviour. This makes RNNs suitable for processing several sequential data such as videos, audio, text, time series, and so forth (Sarker, 2021).

RNN mechanism: RNNs were mentioned many times in literature, one of which could be the most similar to today's vanilla RNN can be referenced by Elman (1990). In a schematic representation, it can be observed that an RNN layer employs a "for" loop to sequentially process each timestep within a given sequence. Simultaneously, the layer maintains an internal state that effectively encodes pertinent information regarding the timesteps it has encountered thus far.

Figure 1a illustrates the structure of a vanilla RNN, while its unrolled version is depicted in Figure 1b. The equations involved in vanilla RNN are:

$$H_t = f_h\left(U \cdot X_t + W \cdot H_{t-1}\right), \tag{1}$$

$$O_t = S \cdot H_t \tag{2}$$

where at each timestep t, Equation (1) calculates the hidden state value H_t using previous hidden state H_{t-1} and input X_t, and Equation (2) calculates the output O_t. U, W, and S as weight matrices learned during the training of the network. The function f_h is a type of smooth, bounded function commonly used in various fields of research. Examples of such functions include the logistic *sigmoid* function and the hyperbolic *tangent* function. These functions exhibit desirable properties such as smoothness and boundedness, making them suitable for a wide range of applications.

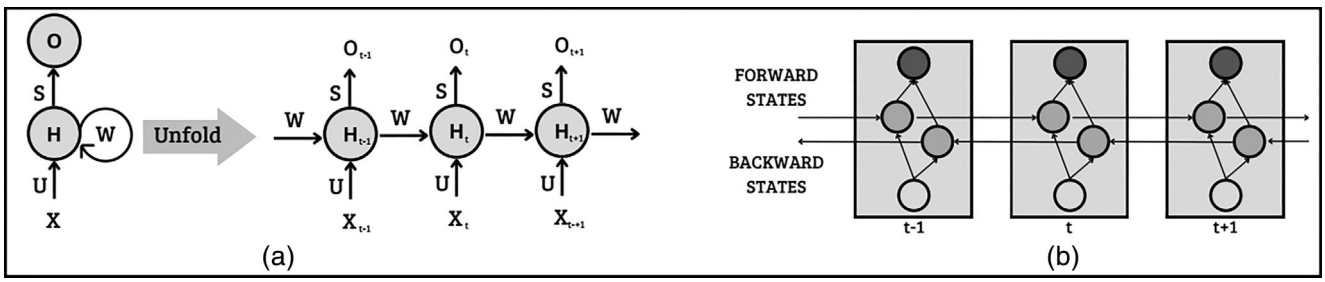

FIGURE 1 (a) Simple diagram of a recurrent layer. (b) The general structure of BRNN shown unfolded in time for three time steps.

RNN problems: Vanilla RNNs pose challenges during training due to issues such as vanishing gradients, where gradients become extremely small (go to zero) and may cause the network to effectively stop learning, or exploding gradients (go to infinity), where gradients become exceptionally large and lead to unstable training. These problems arise from the recurrent application of the same parameters throughout the training process. However, these challenges were mitigated with the development of more sophisticated architectures such as long short-term memory (LSTM) networks and gated recurrent units (GRUs). LSTM, introduced in (Hochreiter & Schmidhuber, 1997), and GRU, proposed in (Chung et al., 2014), address these issues by incorporating mechanisms to regulate the flow of information and gradients within the network. These architectures are designed to better retain long-term dependencies and prevent the vanishing or exploding gradient problem. Further details on LSTM and GRU will be provided in subsequent sections.

2.1 | Long short-term memory

The LSTM unit was originally introduced by Hochreiter and Schmidhuber in their pioneering paper (Hochreiter & Schmidhuber, 1997). Subsequent to its inception, a number of minor adjustments have been implemented to the initial LSTM unit. One of the well-documented implementations was initially introduced in an important paper by Graves (2013).

In contrast to a conventional RNN unit that calculates a weighted sum of the input signal and applies a nonlinear function, an LSTM unit distinguishes itself by preserving a memory C_t at each time step t. The schematic representation of a typical LSTM cell is depicted in Figure 2a.

A typical LSTM unit is composed of several components, namely an input gate, a forget gate, an output gate, and a cell state. These components are calculated using the following equations:

$$f_t = \sigma(W_f.[h_{t-1},x_t] + b_f), \tag{3}$$

$$i_t = \sigma(W_i.[h_{t-1},x_t] + b_i), \tag{4}$$

$$\tilde{C}_t = \tanh(W_C.[h_{t-1},x_t] + b_C), \tag{5}$$

$$C_t = f \circ C_{t-1} + i_t \circ \tilde{C}_t, \tag{6}$$

$$o_t = \sigma(W_o.[h_{t-1},x_t] + b_o), \tag{7}$$

$$h_t = o_t \circ \tanh(C_t). \tag{8}$$

The weights matrices, denoted as W_f, W_i, W_C, W_o along with the bias terms b_f, b_i, b_C, b_o are essential components of the LSTM unit that are iteratively adjusted during the training process.

Forget gate: The gate in question serves the purpose of determining the degree to which the pre-existing memory should be either discarded or preserved. To compute f_t at timestep t, the sigmoid function σ is applied to the preceding hidden state h_{t-1} and present input x_t as shown in Equation (3). The resulting values come out in the range [0, 1], closer to 0 means data should be lost, and closer to 1 suggests it should be kept.

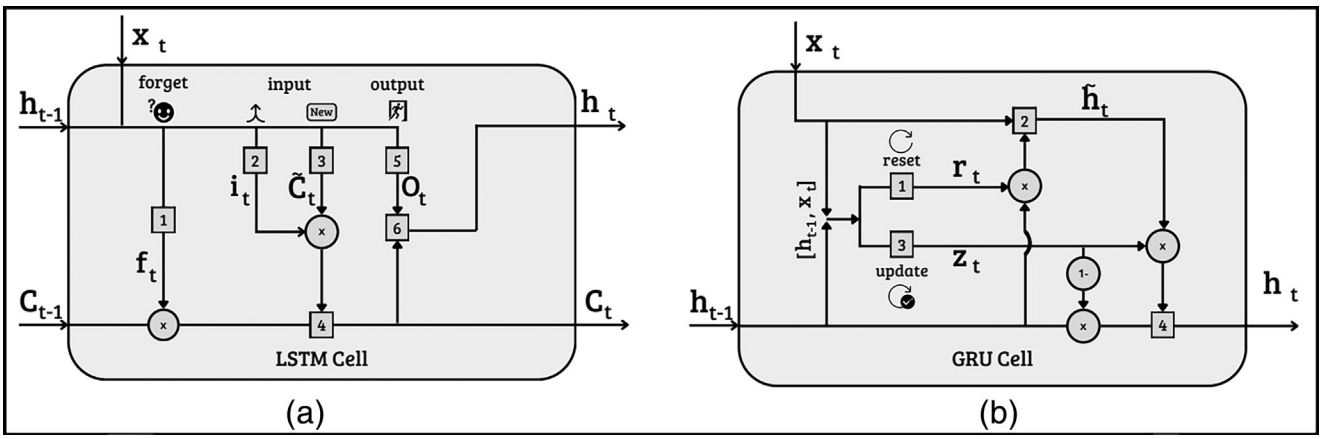

FIGURE 2 Simple diagram of (a) a single standard LSTM cell, (b) a single GRU cell.

Input gate: The gate in question serves as a modulator, regulating the extent to which the recently acquired memory content is incorporated into the memory cell. To compute i_t at timestep t, the previous hidden state h_{t-1} and the current input x_t are passed into a sigmoid function σ that decides which values will be updated and filter out unwanted information, by transforming the values to the range [0, 1] as shown in Equation (4). The previous hidden state h_{t-1} and the current input x_t are also passed into a *tanh* function to compress values into the range [−1, 1] as shown in Equation (5).

Cell state: C acts as the networks' memory. At each timestep t, the memory cell C_t is updated by selectively incorporating new memory content while also considering the existing memory. To compute C_t, it is necessary to perform a point-wise multiplication between the cell state and the forget vector f_t. There is a potential risk of the cell state values being affected if they are multiplied by values that are close to zero. Then the output of a point-wise multiplication of i_t and $\tilde{C}_t$ is added using a point-wise addition. That gives the new cell state as shown in Equation (6).

Output gate: The output gate plays a crucial role in determining the subsequent hidden state. The computation of o_t involves the utilization of the previous hidden state h_{t-1} and the current input x_t, which are both fed into the sigmoid function σ, as depicted in Equation (7). The altered state of cell C_t, which has undergone recent modifications, is subsequently conveyed to the hyperbolic tangent *tanh* function. In accordance with the mathematical formulation presented in Equation (8), the output of the *tanh* is combined with the output of the sigmoid function σ to determine the specific information that the hidden state should encompass. The subsequent time step involves the propagation of the newly updated cell state C_t and the revised hidden state h_t.

2.2 | Gated recurrent unit

The gated recurrent unit (GRU) (Chung et al., 2014) is a variant of the RNN that was proposed following the LSTM unit. The GRU is a simplified version of the LSTM model. While both models are similar, the GRU eliminates the need for a separate cell state and instead uses the hidden state to transfer information. A GRU unit has only two internal gates, namely, the update gate z_t and the rest gate r_t. The structure of a standard GRU cell is shown in Figure 2b.

The computations involved in the update and reset gates in a typical GRU unit are presented in the following equations:

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]), \tag{9}$$

$$\tilde{h}_t = \tanh(W_h \cdot [r_t \circ h_{t-1}, x_t]), \tag{10}$$

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]), \tag{11}$$

$$h_t = (1 - z_t) \circ h_{t-1} + z_t \circ \tilde{h}_t, \tag{12}$$

with W_r, W_h, and W_z weights matrices that are learned during the training.

Reset gate: The reset gate is utilized to determine the quantity of previous data that needs to be retained. r_t is computed by passing the previous steps' memory h_{t-1} and the current steps' input x_t to a sigmoid function σ as shown in Equation (9).

The reset gate is element-wise multiplied '$\circ$' to the hidden state h_{t-1}, and concatenated with the current input x_t. The resulting values are then fed to a *tanh* activation function to generate a vector $\tilde{h}_t$, that stores values in the range [−1, 1] as the new beliefs of the cell as shown in Equation (10).

Update gate: The Update gate exhibits similar functionality to the forget and input gates observed in a LSTM network. z_t is computed by passing the previous steps' memory h_{t-1} and the current steps' input x_t to a sigmoid function σ as shown in Equation (11). This determines which data should be discarded and what new should be added. The updated hidden state, denoted as h_{t-1}, is a result of blending the new beliefs of the cell, $\tilde{h}_t$, and the current hidden state, $\tilde{h}_t$, in a proportion determined by the update gate, z_t. This process is mathematically expressed in Equation (12).

2.3 | Types of RNN

Various forms of RNN architectures have been proposed in order to effectively tackle a diverse range of problems that are based on sequential data. Figure 3a depicts a **one-to-one** RNN architecture, which can be considered as a neural network with weight sharing. The model in question, commonly referred to as the Vanilla Neural Network, is a versatile tool applicable to a wide range of machine learning tasks characterized by a singular input and output. In Figure 3b a **one-to-many** RNN, which generates a series of outputs given one input, such as in the image captioning task (Xu et al., 2015). The third type depicted in Figure 3c is the **many-to-one**. In this scenario, the model takes a sequence of inputs and generates a single output. A good example that uses this scheme is text classification when the objective is to assign a single label for an entire text

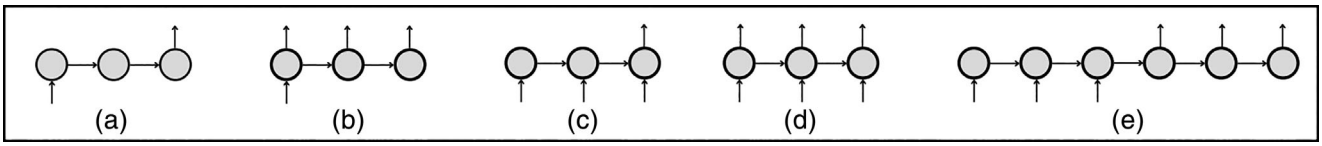

FIGURE 3 RNN types. (a) One-to-one, (b) one-to-many, (c) many-to-one, (d) many-to-many with alignment, (e) many-to-many without alignment.

sequence as in sentiment analysis (Jbene, Raif, et al., 2022), or intent detection (ID) (Jbene, Tigani, et al., 2022; Ravuri & Stolcke, 2015). Figure 3d,e shows **many-to-many** RNNs. The first one is with a one-to-one alignment between the number of input and output timesteps. And the second one is without a specific alignment between the input and output. The input sequence comprises numerous inputs, yielding multiple outputs. Both the input and output are sequences of arbitrary length. This configuration is commonly known as a **sequence-to-sequence** or **encoder-decoder** framework. It has found successful applications in various fields, such as machine translation, where the input text is translated from one language to another (Sutskever et al., 2014).

2.4 | Variants of RNN

Throughout recent years, more complex variants of RNNs were proposed as the complexity of the tasks and the requirement for more performance increase. This includes Bidirectional RNNs, Deeper RNNs, and attention-based RNNs, which we discuss in the following sections.

2.4.1 | Bidirectional recurrent neural networks

Conventional RNN architectures make conclusions about a current state taking into consideration only the previously seen inputs. A bidirectional recurrent neural network (BRNN) with two hidden layers running in opposite directions was proposed by Schuster and Paliwal (1997) to tackle this issue. BRNN consists of a forward and backward RNN layer as shown in Figure 1b. Neuron states are divided into two parts: one that governs the positive time direction (ahead states) and another that governs the negative time direction (backward states).

BRNN encoder reads the input vectors $x=(x_1,x_2,\ldots,x_T)$ and generates T hidden states by concatenating the forward pass $\overrightarrow{h}$ and backward pass $\overleftarrow{h}$ hidden states. Thus, the last state of the BRNN carries information about the entire source sequence. The computations involved in the BRNN cell are presented in the following equations (Sutskever et al., 2014):

$$\overrightarrow{h}_t = f\left(\overrightarrow{W}X_t + \overrightarrow{V}\overrightarrow{h}_{t-1} + \overrightarrow{b}\right), \tag{13}$$

$$\overleftarrow{h}_t = f\left(\overleftarrow{W}X_t + \overleftarrow{V}\overleftarrow{h}_{t-1} + \overleftarrow{b}\right), \tag{14}$$

$$y_t = g\left(\overrightarrow{U}\overrightarrow{h} + \overleftarrow{U}\overleftarrow{h} + c\right). \tag{15}$$

Using non-linear functions f and g, along with weight matrices $\overrightarrow{W}$, $\overleftarrow{W}$, $\overrightarrow{U}$, $\overleftarrow{U}$, $\overleftarrow{V}$, and $\overrightarrow{V}$, and bias terms $\overrightarrow{b}$, $\overleftarrow{b}$, and c, all of which are learned during network training, the output signal is represented by y_t.

2.4.2 | Deep recurrent neural networks

The utilization of deep architectures, specifically deep recurrent neural networks (DRNNs), has played a pivotal role in the notable achievements observed in neural network systems as of late. According to previous research (Graves et al., 2013), it has been observed that deep architectures possess the ability to construct increasingly sophisticated representations of data.

DRNNs can be characterized by various configurations, but a more straightforward approach involves the stacking of multiple hidden layers of RNNs. In this setup, the output sequence of one layer serves as the input sequence for the subsequent layer.

If we assume that the same hidden layer function is employed across all N layers within the stack. Then, the hidden vector sequences, denoted as h^n, are computed iteratively for each layer, starting from $n=1$ up to N, and for each time step, denoted as t, ranging from 1 to T as in Equation (16):

$$h_t^n = \phi_h\left(W_{h^{n-1}h^n}h_t^{n-1} + W_{h^nh^n}h_{t-1}^n + b_t^n\right), \tag{16}$$

where $h^0 = x$, W, and b are weight matrices and ϕ_h is a non-linear activation function. The network outputs y_t are calculated following Equation (17):

$$y_t = W_{h^Ny}h_t^N + b_y. \tag{17}$$

Deep bidirectional recurrent neural networks (DBRNNs) can be effectively realized by substituting each hidden sequence h^n with two distinct sequences, namely the forward sequence $\overrightarrow{h}_t$ and the backward sequence $\overleftarrow{h}_t$. To ensure comprehensive information flow, it is crucial that every hidden layer in the network receives input from both the forward and backward layers at the preceding level.

2.4.3 | Attention-based RNN model

The attention mechanism is a computational technique that is inspired by the cognitive processes observed in human psychology. The prevailing hypothesis posits that human cognitive processes exhibit a tendency to allocate priority to particular components within the perceptual domain, thereby disregarding the remaining visible information. In the realm of text streams, the concept of attention pertains to the cognitive capacity to focus one's mental resources on particular components within a given sequence, while simultaneously disregarding extraneous information. This cognitive process plays a pivotal role in facilitating optimal learning outcomes.

The efficacy of the attention mechanism has been empirically validated across a range of machine learning (ML) tasks, including but not limited to machine translation, video captioning, and image captioning (Xu et al., 2015; Chen, Zhang, et al., 2017; You et al., 2016). The attention mechanism was initially introduced by Bahdanau et al. (2015) in the context of neural machine translation, a task that entails a many-to-many mapping. The utilization of an encoder-decoder model was employed by the researchers, alongside the implementation of a novel approach to integrate weights onto the intermediate hidden values. The weights play a crucial role in determining the allocation of attention by the model towards individual elements within the input sequence at each decoding step.

There are many variants of the attention mechanism in the literature, we find Content-based attention (Graves et al., 2014), additive attention (Bahdanau et al., 2015), location-based attention (Luong et al., 2015), general attention (Luong et al., 2015), dot-product (Luong et al., 2015), and finally scaled dot-product attention (Vaswani et al., 2017). To illustrate the concept of attention we consider two examples. The global additive attention of Bahdanau et al. (2015) that we depict in Figure 4a. And the Global multiplicative attention version of Luong et al. (2015) that we depict in Figure 4b.

The formulas used to calculate the attention vectors in Figure 4a,b are presented below in Equations 18 to 21.

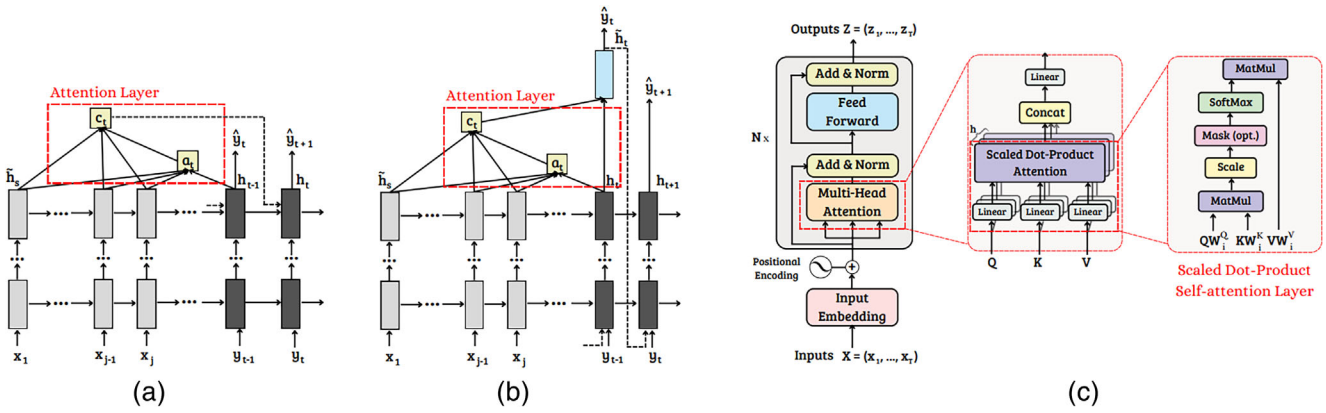

FIGURE 4 Graphical illustration of different versions of the attention mechanism. (a) Bidirectional RNN-based encoder-decoder with Bahdanou Attention. (b) Bidirectional RNN-based encoder-decoder with Luong (Global) attention (c) transformer encoder with scaled dot-product self-attention.

$$a_t(s) = \frac{\exp(score(h_t, \bar{h}_s))}{\sum_{s'=1}^{s} \exp(score(h_t, \bar{h}_{s'}))}, \tag{18}$$

$$c_t = \sum_{s=1} a_t(s)\, \bar{h}_{s'}, \tag{19}$$

$$a_t = \tanh(W_c[c_t; h_t]), \tag{20}$$

$$score(h_t, \bar{h}_s) = \begin{cases} h_t^T W \bar{h}_s & [Luongs' multiplicative\ style] \\ v_a^T \tanh(W_1 h_t + W_2 \bar{h}_s) & [Bahdanaus' additive\ style] \end{cases}, \tag{21}$$

where $a_t(s)$: attention weight for encoder hidden state $\bar{h}\ s$ at decoder timestep t; h_t: decoder hidden state at timestep t; $\bar{h}_s$: encoder hidden state at position s; c_t: context vector at decoder timestep t; a_t: attention vector at decoder timestep t; $score(h_t, \bar{h}_s)$: score function between ht and $\bar{h}\ s$; W, W_1, W_2: weight matrices; v_a^T: weight vector.

For recent studies in NLP, RNNs based on the attention mechanism have become a major trend in various text-processing research fields, such as question answering (Chen, Hu, et al., 2017), text classification (Liu & Lane, 2016), recommendation systems (Ying et al., 2018) and so on.

3 | TRANSFORMER MODELS

Transformer models (Vaswani et al., 2017) have demonstrated success across a wide range of tasks. In this section, we describe its essential components and working mechanisms.

3.1 | Self-attention

Self-attention is known to be the central and indispensable component of Transformer models. Self-attention is a very effective method of leveraging context-aware features over variable-length sequences for NLP tasks (Tan et al., 2018; Zhong et al., 2018).

Given a matrix of input vectors $X \in R$, self-attention maps it to queries Q, keys K, and values V matrices using different linear projections. The output matrix is a weighted sum of values as shown in Equation (22).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{22}$$

where d_k is the key dimensionality.

3.2 | Transformer architecture

The transformer, as described by Vaswani et al. (2017), is a model that employs an encoder-decoder architecture. It leverages stacked self-attention and fully connected layers in both the encoder and decoder components. The encoder is comprised of N layers, with each layer consisting of two sub-layers: a Multi-head self-attention mechanism, and a Feed-forward network, which is illustrated in Figure 4c.

The multi-head attention mechanism is a technique that allows for the extraction of multiple representations, denoted as h, for a given input (Q, K, V). Each representation is obtained by applying scaled dot-product attention to the input. The attention mechanism computes the similarity between the query Q and key K vectors, and uses this similarity to weight the values V. This process is repeated for each representation, resulting in h different sets of attention weights. These sets are then concatenated together, and the concatenated result is projected through a feed-forward layer. This multi-head attention mechanism enables the model to capture different aspects of the input and incorporate them into the final representation. This can be expressed in the same notation as Equation (22):

$$\text{Head}_i = \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right), \tag{23}$$

$$\text{MultiHead}_i(Q, K, V) = \text{Contact}_i(\text{Head}_i)W^0, \tag{24}$$

where the W_i and W^0 are parameter matrices.

The feed-forward network serves as the second component within each layer of the Transformer network. The use of a two-layer network with a Rectified Linear Unit (ReLU) activation is suggested by Vaswani et al. (2017). In a similar manner, it can be observed that every layer of the decoder architecture comprises the aforementioned two sub-layers, along with an extra sub-layer dedicated to multi-head attention. This supplementary sub-layer is responsible for receiving the output from the encoder as both its keys and values.

In recent years, there has been a proliferation of proposed iterations of the Transformer model. Multi-head attention is a technique employed in various ways within the context of neural network architectures. Specifically, it is utilized in three distinct manners: *encoder self-attention*, *decoder self-attention*, *and encoder-decoder attention*. Several Transformer models have been proposed in recent literature, as evidenced by the works cited in references (Brown et al., 2020; Devlin et al., 2019; Liu, Ott, et al., 2019).

4 | INTENT DETECTION FOR DIALOGUE SYSTEMS

Natural language understanding (NLU) serves as an essential and foundational element within conversational systems. ID and SF are fundamental components of NLU. An illustrative example for these tasks is shown in Figure 5.

The primary emphasis of this paper lies in the investigation of the ID task, whereas certain recent studies have approached the two tasks in a combined manner. Assuming a strong relationship exists between the two tasks, they propose to train models in a multi-task fashion to achieve promising results.

4.1 | Defining the intent detection task

Intent detection can be considered as a sentence classification task. Given an input utterance $X = (x_1,...,x_n)$, where n denotes the length of X, the model should assign one of the N pre-defined set of intent labels y_i based on the components of the user utterance, which can be formulated as:

$$y_i = \arg\max_{i \in N} P(y_i/x). \tag{25}$$

4.2 | Intent detection data sets

Over the past few years, numerous data sets have been introduced for the ID task. In the following section, we provide a concise overview of 17 widely utilized data sets. The majority of these data sets are in English, while some are multilingual. Additionally, they span a diverse array of domains, including travel, hotels (Price, 1990), entertainment (Coucke et al., 2018), banking (Casanueva et al., 2022), and agriculture (Hao et al., 2023). Furthermore, Table 1 provides additional statistics about each data set.

1. **ATIS**: the ATIS (Air Travel Information System) corpus (Price, 1990) is the most used data set for Spoken Language Understanding (SLU) research. It was collected to build a spoken dialogue system to provide information on US flights. There are multiple variants of the data set. In this paper, we've only considered papers that used the most common version in (Tür et al., 2010). The data set consists of sentences of people making flight reservations. There are 4978 sentences for training and 893 sentences for testing. The number of distinct intents is 18, as shown in Table 1.
2. **Ask ubuntu, Chatbot**, and **Web Applications**: The three data sets were proposed in reference (Braun et al., 2017). The Web Applications and *askubuntu* data sets are derived from queries scraped from *StackExchange* forums like askubuntu.com and webapps.stackexchange.com. Despite not being originally designed for dialogue systems, they consist of dialogue-style utterances targeting various intents related to software support. Both data sets are in English. The Chatbot Corpus, on the other hand, comprises real user utterances from a public transit query dialogue system in Munich, Germany. While primarily in English, these utterances contain numerous German place names.

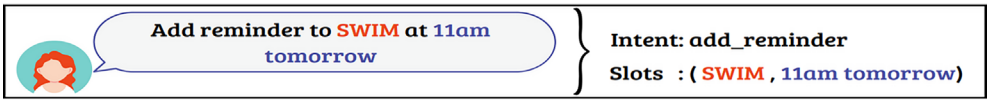

FIGURE 5 An Illustrative Instance from the xSID data set Featuring the Intent 'add_reminder'.

TABLE 1 Intent detection data sets primarily comprising English utterances are arranged in the table from oldest to most recent. (Note that data sets labelled with 'Eng*' also include versions available in other languages).

Data set	Domain	Intents	Utterances	Source	Lang	Licence
ATIS (Price, 1990)	Travel & hotels	24	5871	Crowd	Eng	LDC
Ask Ubuntu (Braun et al., 2017)	Software support	5	162	Users	Eng	CC-BY-SA 3.0
Chatbot (Braun et al., 2017)	Transport	2	206	Users	Eng	CC-BY-SA 3.0
Web Applications (Braun et al., 2017)	Software support	8	89	Users	Eng	CC-BY-SA 3.0
SNIPS (Coucke et al., 2018)	Restaurant & Entertainment	7	14,484	Crowd	Eng	CC0 1.0
TOP (Gupta et al., 2018)	-	25	44,783	Crowd	Eng	-
HWU-64 (Liu, Eshghi, et al., 2019)	Home automation	64	25,716	Crowd	Eng	CC-BY-SA 3.0
Facebook (Schuster et al., 2019)	Multiple domains	12	43,323	Crowd	Eng*	CC-BY-SA
TOPv2 (Chen et al., 2020)	Multiple domains	80	181,000	Crowd	Eng	-
Leyzer (Sowanski & Janicki, 2020)	Home automation	186	3892	Generated	Eng*	CC-BY-NC-ND 4.0
MixATIS (Qin et al., 2020)	Travel & hotels	24	20,000	Derived	Eng	-
MixSNIPS (Qin et al., 2020)	-	7	50,000	Derived	Eng	-
CSTOP (Einolghozati et al., 2021)	Multiple domains	19	5803	Expert	Eng	-
MTOP (Li et al., 2021)	Multiple domains	117	22,288	Crowd	Eng*	-
xSID (Van der Goot et al., 2021)	Multiple domains	16	44,405	Derived	Eng*	CC-BY-SA 4.0
NLU++ (Casanueva et al., 2022)	Banking	62	3080	Users	Eng	-
NLU++ (Casanueva et al., 2022)	Agriculture	22	11,976	Users	zh-CN	-

3. **SNIPS**: the SNIPS data set (Coucke et al., 2018) was collected by Snips voice assistant. The data set under consideration comprises a total of seven distinct intent types. The distribution of samples across intention labels exhibits a near-equilibrium, with each label being represented by a comparable number of samples. The data set contains comprehensive information, which can be accessed in Table 1. The complexity of the SNIPS data set surpasses that of the ATIS data set due to the presence of multi-domain intents and a comparatively extensive vocabulary.
4. **TOP**: data set, introduced by (Gupta et al., 2018), employs a hierarchical annotation scheme to address the challenge of utterances associated with multiple intent labels. Every utterance in TOP is assigned a top-level intent. Notably, 35% of all utterances exhibit multiple intents.
5. **HWU-64**: joint corpus encompasses an extensive array of intent categories, including but not limited to home automation, travel, and general inquiries such as weather queries. It comprises a total of 64 intent categories spanning 25, 716 crowdsourced utterances (Liu, Eshghi, et al., 2019).
6. **Facebook**: this task Oriented Dialogue data set was specifically designed to assess multilingual transfer learning capabilities (Schuster et al., 2019). It comprises 12 intents categorized across three domain areas related to setting alarms, reminders, and querying weather information. Initially, English language utterances were collected by instructing crowd workers to provide sample commands or inquiries they might address to a device capable of handling the three intent categories. Subsequently, separate crowd workers annotated both intent labels for each utterance. A portion of the English utterances was then translated into Spanish and Thai by native speakers.
7. **TOPv2**: Chen et al. (2020) expanded upon the TOP corpus by introducing 72 additional intents, resulting in the creation of TOPv2. This corpus encompasses approximately 180, 000 utterances distributed across 80 intents. Similar to TOP, TOPv2 was compiled through crowdsourcing utterances. Its annotation structure mirrors that of TOP, employing a hierarchical style. It is estimated that roughly 16% of utterances in TOPv2 are multi-intent.
8. **Leyzer**: a multilingual corpus introduced by (Sowanski & Janicki, 2020), encompasses English, Spanish, and Polish languages. It sets new standards with its imbalanced data sets, containing between one and 672 samples per intent class. With 186 intents covering various tasks like news, weather, calendar, and web search, Leyzer stands as the largest data set in this survey. Unlike most data sets discussed in this article, Leyzer's sample utterances are not human-generated through crowdsourcing or user queries; instead, they are generated using predefined grammars.
9. **MixATIS** and **MixSNIPS**: were introduced by Qin et al. (2020) to address the scarcity of multi-intent data sets for evaluating intent classification and slot-filling models. They achieve this by synthesizing multi-intent queries from single-intent data sets, specifically combining queries from the **Snips** and **ATIS** data sets to create MixSnips and MixATIS, respectively. These queries are formed by connecting single-intent queries using conjunctions such as 'and', ',' (comma), 'and also', and 'and then'.

10. **CSTOP**: corpus (Einolghozati et al., 2021) is a bilingual data set combining Spanish and English, featuring code-switched ('Spanglish') queries. These queries focus on weather and device domains, encompassing 19 intents. CSTOP was meticulously crafted by proficient bilingual workers skilled in code-switched Spanish and English. Although some utterances in CSTOP have multiple intent annotations, it is estimated that only about 6% include such annotations, predominantly within the weather domain.
11. **MTOP**: inspired by the nested structure of queries in various TOP data sets, is a comprehensive benchmark comprising nested queries in six languages: English, Spanish, French, German, Hindi, and Thai (Li et al., 2021). Encompassing 11 domains and 117 intents (with each domain containing between 3 and 27 intents), MTOP was meticulously crafted in multiple phases. Initially, crowd workers provided English utterances for hypothetical scenarios within specific domains. Then, professional translators translated these utterances into each target language.
12. **xSID**: corpus serves as a benchmark for cross-lingual transfer, featuring abundant training data in English and limited evaluation data in 13 languages: Arabic, Chinese, Danish, Dutch, English, German, Indonesian, Italian, Japanese, Kazakh, Serbian, Turkish, and South Tyrolean (each language comprising 800 evaluation utterances) (Van der Goot et al., 2021). Constructed by sampling data from **SNIPS** and **Facebook**, the data set underwent expert translation from English to each target language. xSID encompasses 16 intents and 41 slot types.
13. **NLU++**: benchmark, introduced by (Casanueva et al., 2022), comprises two joint-task data sets: Banking (with 48 intents and 13 slots) and Hotels (with 40 intents and 14 slots). Combined, these data sets offer 62 intents, 17 unique slots, and a total of 3, 080 user-generated utterances, many of which involve multiple intents. Unlike traditional annotation methods, NLU++ applies intent labels directly to each utterance instead of annotating spans for separate intent segments. These intent labels vary in granularity, allowing broad categories like 'cancel' to apply across both Banking and Hotel domains, while more specific categories like 'account' are domain-specific.
14. **AGIS**: was developed and annotated by the authors as referenced in (Hao et al., 2023), featuring 22 intent categories, 10 slot types, and a total of 11, 976 samples. Notably, it stands as the first data set specifically tailored for Chinese agricultural joint ID and SF tasks.

4.3 | RNN-based models

The specific properties of RNNs make them suitable for sequence modelling applications. LSTM, GRU, and their bidirectional variants, with the attention mechanism, have been shown to produce a good performance for ID. For instance, (Ravuri & Stolcke, 2015) successfully applied LSTM for ID, indicating that sequential features benefit intent detection. The authors in (Liu & Lane, 2016) employed an attention-based encoder-decoder BRNN model for joint ID and SF, allowing the network to learn the relationship between slot and intent. In a recent study, Liu, Meng, et al. (2019) introduced a groundbreaking concept known as the collaborative memory network (CM-Net). This innovative approach aims to effectively capture the semantic correlations between words by leveraging a collaborative framework. By enhancing the information flows within the model, the researchers observed a notable improvement in its overall performance.

4.4 | Transformer-based models

Recent progress in pre-trained neural language models based on Transformer architecture has significantly improved the performance of many NLP tasks. For ID, (Qin et al., 2019) used a pre-trained embedding encoder to replace its attention encoder (Stack-Propagation + BERT), which boosts the model's performance. In a previous study, the BERT-Cap model was introduced by (Liu et al., 2020). This model utilizes a pre-trained BERT transformer to encode sentence sequences. Subsequently, a capsule network with a dynamic routing mechanism is employed to capture higher-level features. The efficacy of the PMM-Att model, as proposed in reference (Yang et al., 2021), has been further substantiated through experiments involving BERT. In a recent study, a groundbreaking non-autoregressive SLU model called Layered-Refine Transformer was developed by Cheng et al. (2021). This innovative model incorporates a slot label generation (SLG) task and a layered refine mechanism (LRM), resulting in significant improvements in performance.

4.5 | Hybrid models

Some recent studies tried to benefit from both worlds and propose hybrid RNN-Transformer architectures. For instance, Huang et al. (2020) claim that adopting LSTM as an intent decoder and leveraging BERT as an additional encoder further improves their multi-view encoder (MV-Encoder). Qin, Liu, et al. (2021) proposed a co-interactive Transformer that consists of a shared encoder based on BiLSTM and a co-interactive Module based on the Transformer. Recently, Ni et al. (2020) proposed BERT-BACBC, which is a hybrid BERT, BiGRU model, achieving the best performance compared to a variety of baselines.

4.6 | Other ID models

The field of ID has historically been dominated by RNN-based models, but recent advancements have seen the emergence of transformer-based approaches. However, notable works have explored alternative methodologies. For instance, by Zhang et al. (2019), the authors introduced a model based on the Capsule neural network architecture, employing a dynamic routing-by-agreement schema. Their CAPSULE-NLU model achieved competitive performance, achieving 97.3% accuracy on SNIPS and 95.0% accuracy on ATIS data sets. Furthermore, Xue and Ren (2021) proposed an intention-enhanced attentive Bert Capsule network, leveraging the capabilities of pre-trained language models to encapsulate contextual information from utterances. By jointly learning label embeddings, they generated intent-concentrated utterance features and guided the aggregation process of the capsule network.

Additionally, other works have explored the utilization of Graph neural networks. For example, by Qin, Che, et al. (2021), the CGCN-AF framework was introduced. This context-aware GCN automatically encapsulates relevant contextual information, with an adaptive fusion layer applied to each token dynamically incorporating pertinent contextual details.

5 | COMPARATIVE STUDY OF ID MODELS

In the subsequent sections, we conduct a comparative analysis of various ID models, considering multiple facets across four distinct data sets. This comparison encompasses the models' performance, the diversity of tasks they were trained on, their efficiency, and their adaptability to low-resource environments. Ultimately, we conclude by synthesizing the key findings and outlining potential avenues for further research.

5.1 | Evaluation metrics

In the domain of Intent identification, accuracy, and F1-score have emerged as the prevailing metrics of choice. Accuracy is a metric that quantifies the proportion of sentences for which the intended meaning was accurately predicted. On the other hand, the F1-score is a measure that combines precision and recall by taking their harmonic mean. The procedure for calculation is depicted as follows:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}, \tag{26}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall}, \tag{27}$$

$$Precision = \frac{TP}{TP + FP}, \tag{28}$$

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{29}$$

True positives (TP) represent the instances where the model correctly predicts the presence of a specific intent. True negatives (TN) signify the cases where the model accurately identifies the absence of that intent. False positives (FP) occur when the model incorrectly predicts the presence of an intent that is not actually present. Finally, False negatives (FN) occur when the model fails to detect an intent that is actually present.

5.2 | Performance comparison across data sets

To assess the performance of RNN-based models relative to Transformer-based models, Table 2 provides a comparative analysis across four widely used data sets for the ID task, drawing from various previous studies. We categorize methods into five classes based on their backbone architecture: those built on GRU, LSTM, transformer encoder, transformer decoder, and hybrid transformer-RNN architectures. The table indicates a marginally superior performance of Transformer-based models, consistent with findings from prior research (Wei et al., 2022). This advantage can be attributed to Transformer models being pre-trained on extensive data sets (Liu, Ott, et al., 2019), whereas RNN-based models typically rely solely on pre-trained word embeddings for word and sentence representations. Using pre-trained word representations still does not give the RNN-based models the comprehensive understanding achieved by Transformer architectures. Additionally, hybrid Transformer-RNN

TABLE 2 Performance analysis of various model types on intent detection benchmark data sets.

Data set	Backbone	Model name	Year	Pre-training (Y/N)	F1 (%)	Acc (%)
ATIS	GRU	BiGRU-CRF (Daha & Hewavitharana, 2019)	2019	N	-	95.60
		Wheel-GAT (Wei et al., 2022)	2022	N	-	97.50
	LSTM	TM + SAN + Bi-LSTM (Yolchuyeva et al., 2019)	2019	N	-	96.81
		BiLSTM+ID+SD (Yang et al., 2021)	2021	N	-	97.00
	Transformer-encoder	RoBERTa + DRM (Shen et al., 2021)	2021	Y	-	98.31
		WFST-BERT (Abro et al., 2022)	2022	Y	98.12	-
	Transformer-decoder	gpt-3.5-turbo (Yoon et al., 2024)	2024	Y	-	40.30
	Transformer and RNN	ICN + MTL (OIR) + BERT (Huang et al., 2021)	2021	Y	-	98.20
		Wheel-GAT + BERT (Wei et al., 2022)	2022	Y	-	98.00
SNIPS	GRU	BiGRU-CRF (Daha & Hewavitharana, 2019)	2019	N	-	97.00
		Wheel-GAT (Wei et al., 2022)	2022	N	-	98.40
	LSTM	DBLC-model (Li et al., 2022)	2022	N	-	96.99
		BiLSTM+ID+SD (Yang et al., 2021)	2021	N	-	98.70
	Transformer-encoder	RoBERTa + DRM (Shen et al., 2021)	2021	Y	-	98.87
		Albert (xxl) (Louvan & Magnini, 2020)	2020	Y	-	99.20
	Transformer-decoder	gpt-3.5-turbo (Yoon et al., 2024)	2024	Y	-	81.68
	Transformer and RNN	ICN + MTL (OIR) + BERT (Huang et al., 2021)	2021	Y	-	99.30
		Wheel-GAT + BERT (Wei et al., 2022)	2022	Y	-	99.30
MixATIS	LSTM	GL-GIN (Qin, Wei, et al., 2021)	2021	N	-	76.30
		SDJN (Chen et al., 2022)	2022	N	-	77.10
	Transformer-encoder	SSRAN (Cheng et al., 2023)	2023	Y	-	77.90
		BiSLU (Tu et al., 2023)	2023	Y	-	81.50
	Transformer-decoder	EN-Llama-2 (Yin, Huang, Xu, Huang, & Chen, 2024)	2024	Y	-	80.60
		EN-Mistral (Yin, Huang, Xu, Huang, & Chen, 2024)	2024	Y	-	82.40
	Transformer and RNN	Uni-MIS (Yin, Huang, & Xu, 2024)	2024	Y	-	78.50
MixSNIPS	LSTM	GL-GIN (Qin, Wei, et al., 2021)	2021	N	-	95.60
		SDJN (Chen et al., 2022)	2022	N	-	96.50
	Transformer-encoder	SSRAN (Cheng et al., 2023)	2023	Y	-	98.40
		BiSLU (Tu et al., 2023)	2023	Y	-	97.80
	Transformer-decoder	EN-Llama-2 [70]	2024	Y	-	96.60
		EN-Mistral (Yin, Huang, Xu, Huang, & Chen, 2024)	2024	Y	-	97.60
	Transformer and RNN	Uni-MIS (Yin, Huang, & Xu, 2024)	2024	Y	-	97.20

models demonstrate slightly improved performance, which is aligned with previous studies (Wei et al., 2022), particularly evident in models like Wheel-GAT and Wheel-GAT+BERT.

5.3 | Evaluation of single-task versus multi-task training

Upon reviewing the literature on ID, we observed a trend towards training models on multiple tasks simultaneously. Proponents of this approach argue that it enhances performance, particularly when tasks are interrelated. To delve deeper into this phenomenon, we present a comparative analysis in Table 3 of state-of-the-art approaches based on their training regimen. We classify these approaches into three main categories: models exclusively trained for ID, models trained jointly on ID and slot filling (SF), and models trained on a broader array of tasks. Our findings indicate that SF emerges as the primary task paired with ID, reflecting their complementary roles in NLU. The close association between ID and SF has prompted the development of models capable of jointly addressing both tasks. However, a preliminary assessment of model performance reveals no discernible difference between models trained solely for ID and those trained for both ID and SF.

TABLE 3 Performance analysis of various model types on ATIS and SNIPS ID benchmark data sets based on the training tasks.

Data set	Training tasks	Model name	Backbone	Year	F1 (%)	Acc (%)
ATIS	ID	Char-Rep-ID (Shivnikar et al., 2021)	RNN	2020	-	99.53
		AgHA-IDSF (Hao et al., 2023)	RNN	2023	-	97.76
		NIDAL (Mullick, 2023)	Transformer	2022	88.90	92.10
		P CMlDlarge mean (Song et al., 2023)	Transformer	2023	-	98.00
	ID and SF	BiGRU-CRF (Daha & Hewavitharana, 2019)	RNN	2019	-	95.60
		Wheel-GAT (Wei et al., 2022)	RNN	2022	-	97.50
		Co-Interactive Transformer (Qin, Liu, et al., 2021)	Transformer	2021	-	97.70
		BERT-FAN (Huang et al., 2024)	Transformer	2024	-	97.80
	Multitasks	gpt-3.5-turbo (Yoon et al., 2024)	Transformer	2024	-	40.30
SNIPS	ID	Char-Rep-ID (Shivnikar et al., 2021)	RNN	2020	-	98.95
		AgHA-IDSF (Hao et al., 2023)	RNN	2023	-	98.29
		NIDAL (Mullick, 2023)	Transformer	2022	95.50	97.90
		P CMlDlarge mean (Song et al., 2023)	Transformer	2023	-	98.20
	ID and SF	BiGRU-CRF (Daha & Hewavitharana, 2019)	RNN	2019	-	97.00
		Wheel-GAT (Wei et al., 2022)	RNN	2022	-	98.40
		Co-Interactive Transformer (Qin, Liu, et al., 2021)	Transformer	2021	-	98.80
		BERT-FAN (Huang et al., 2024)	Transformer	2024	-	98.30
	Multitasks	Deep Multi-task model (Firdaus et al., 2021)	RNN	2020	-	99.60
		gpt-3.5-turbo (Yoon et al., 2024)	Transformer	2024	-	81.68

Moreover, we observe a divergence in training strategies between RNN-based and Transformer-based models. RNN-based models undergo training for at most three tasks concurrently, exemplified by the deep multi-task model (Firdaus et al., 2021). In contrast, Transformer-based models, often pre-trained on extensive data sets, exhibit versatility across various NLU tasks, as demonstrated in the BERT paper (Devlin et al., 2019). Fine-tuning a transformer model further enhances its performance on specific tasks. Additionally, while transformer-decoder models are primarily designed for text generation, recent studies (Yoon et al., 2024) showcase their adaptability for classification tasks like ID through straightforward fine-tuning with the appropriate prompt-output design.

5.4 | Efficiency analysis of ID models

The efficiency of models emerges as a crucial factor in model selection, particularly for deployment on edge devices or low-resource systems. Table 4 presents a comparative analysis of various models based on factors such as model size, number of parameters, training time, and model latency. However, for a fair assessment, our focus lies primarily on model size, parameter count, and corresponding accuracy, as other efficiency metrics can fluctuate based on hardware and experimental setups used during training. Notably, RNN-based models exhibit lighter parameter counts and smaller model sizes compared to their transformer-based counterparts. Additionally, we observe a clear correlation between increased model size and enhanced accuracy among transformer-based models.

In conclusion, while Transformer-based models demonstrate superior performance, their tens of millions of parameters render them impractical for on-device deployment and constrained environments, contrasting with the lighter RNN-based models (Agarwal et al., 2021).

5.5 | Challenges and research directions

Much of the research in the field has been focused on enhancing performance, with relatively less attention given to efficiency. As demonstrated in the comparative analysis conducted in the preceding sections, Transformer-based models exhibit slightly superior performance compared to RNN-based approaches across benchmark data sets. This advantage can be attributed to their comprehensive pre-training on extensive data sets, providing them with a broad understanding of diverse domains (Zhao et al., 2021). Consequently, fine-tuning models like BERT (Devlin et al., 2019) often yields better results than training RNN-based models from scratch, particularly for specific tasks like identity (ID) recognition.

TABLE 4 Efficiency of different models on the ATIS and SNIPS data sets.

Data set	Backbone	Model name	Model size (MB)	Parameters (M)	Training time (s)	Latency (ms)	Acc (%)
ATIS	Transformer	Stack-Prop+BERT (Agarwal et al., 2021; Qin et al., 2019)	>1200	-	-	-	97.50
		BERT-FAN76	-	116.6	1456	-	97.80
		$BERT_{Base}$ (Agarwal et al., 2021; Devlin et al., 2019)	-	110	-	1580	97.16
		DistillBERT (Agarwal et al., 2021; Sanh et al., 2019)	-	66	-	781	96.98
		DistillBERT-FAN (Huang et al., 2024)	-	59.9	857	-	97.90
		MobileBERT (Agarwal et al., 2021; Sun et al., 2020)	-	24.6	-	545	96.30
		TinyBERT-FAN (Huang et al., 2024)	-	15.8	485	-	97.80
		TinyBERT (Agarwal et al., 2021; Jiao et al., 2020)	-	14.5	-	162	95.97
	RNN	SF-ID (BLSTM)net (Agarwal et al., 2021; Haihong et al., 2019)	11.61	-	-	-	97.76
		SG-BiLSTM+Attention (Agarwal et al., 2021; Goo et al., 2018)	11.57	-	-	-	94.10
		Stack-Prop (Agarwal et al., 2021; Qin et al., 2019)	3.32	-	-	-	96.90
		LIDSNet (Agarwal et al., 2021)	0.63	0.065	-	18	95.97
SNIPS	Transformer	Stack-Prop+BERT (Agarwal et al., 2021; Qin et al., 2019)	>1200	-	-	-	99.00
		BERT-FAN76	-	116.6	1456	-	98.30
		$BERT_{Base}$ (Agarwal et al., 2021; Devlin et al., 2019)	-	110	-	1580	98.26
		DistillBERT (Agarwal et al., 2021; Sanh et al., 2019)	-	66	-	781	97.94
		DistillBERT-FAN (Huang et al., 2024)	-	59.9	857	-	98.00
		MobileBERT (Agarwal et al., 2021; Sun et al., 2020)	-	24.6	-	545	97.71
		TinyBERT-FAN (Huang et al., 2024)	-	15.8	485	-	97.80
		TinyBERT (Agarwal et al., 2021; Jiao et al., 2020)	-	14.5	-	162	98.00
	RNN	SF-ID (BLSTM)net (Agarwal et al., 2021; Haihong et al., 2019)	11.61	-	-	-	97.43
		SG-BiLSTM+Attention (Agarwal et al., 2021; Goo et al., 2018)	11.57	-	-	-	97.00
		Stack-Prop (Agarwal et al., 2021; Qin et al., 2019)	3.32	-	-	-	98.00
		LIDSNet (Agarwal et al., 2021)	0.63	0.59	-	18	98.00

However, despite their impressive performance, Transformer-based models encounter several challenges. They tend to be more computationally demanding during both training and inference stages, especially evident in larger variants such as LLAMA-2 (Touvron et al., 2023) and GPT-3 (Brown et al., 2020), which possess the remarkable ability to perform ID tasks without explicit training due to their versatile nature. Nonetheless, these advanced models still grapple with issues such as hallucination.

Moreover, current ID methodologies face various challenges, including inefficiency and limited generalizability. The scarcity of non-English data sets in the Data set section highlights a promising research direction for addressing low-resource languages. Exploring diverse research avenues could help mitigate these challenges, emphasizing the importance of balancing model complexity, performance, and computational efficiency.

6 | CONCLUSION

The present study conducted an in-depth review of existing scholarly literature concerning ID for Task-oriented Conversational Agents. It provided a comprehensive overview of RNNs and Transformer Models, tracing their evolution and various adaptations for the ID task in CAs. A comparative analysis was presented, evaluating different RNN and Transformer-based models across widely used benchmark data sets from performance, training tasks, and efficiency perspectives.

Moreover, the study highlighted the current status of state-of-the-art models, along with their associated challenges and potential research avenues. In terms of future research, the aim is to explore novel methods that overcome the limitations of existing techniques and validate their efficacy across more complex and diversified data sets. This review seeks to push the boundaries of current methodologies and advance the field of Task-oriented Conversational Agents.

DATA AVAILABILITY STATEMENT

Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

ORCID

Abdellah Chehri https://orcid.org/0000-0002-4193-6062
Gwanggil Jeon https://orcid.org/0000-0002-0651-4278

REFERENCES

Abro, W. A., Qi, G., Aamir, M., & Ali, Z. (2022). Joint intent detection and slot filling using weighted finite state transducer and BERT. *Applied Intelligence*, *52*(15), 17356–17370. https://doi.org/10.1007/S10489-022-03295-9

Agarwal, V., Shivnikar, S. D., Ghosh, S., Arora, H., & Saini, Y. (2021). LIDSNet: A lightweight on-device intent detection model using deep Siamese network. In M. A. Wani, I. K. Sethi, W. Shi, G. Qu, D. S. Raicu, & R. Jin (Eds.), *20th IEEE international conference on machine Learning and applications, ICMLA 2021* (pp. 1112–1117). IEEE. https://doi.org/10.1109/ICMLA52953.2021.00182

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In I. Y. Bengio & Y. LeCun (Eds.), *3rd international conference on Learning representations, ICLR 2015*. Conference Track Proceedings. http://arxiv.org/abs/1409.0473

Bhathiya, H. S., & Thayasivam, U. (2020). Meta Learning for few-shot joint intent detection and slot-filling. In *Proceedings of the 2020 5th international conference on machine learning technologies* (pp. 86–92). Association for Computing Machinery. https://doi.org/10.1145/3409073.3409090

Braun, D., Hernandez-Mendez, A., Matthes, F., & Langen, M. (2017). Evaluating natural language understanding services for conversational question answering systems. In K. Jokinen, M. Stede, D. DeVault, & A. Louis (Eds.), *Proceedings of the 18th annual SIGdial meeting on discourse and dialogue* (pp. 174–185). Association for Computational Linguistics. https://doi.org/10.18653/V1/W17-5522

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (Vol. 2020, pp. 1877–1901). Curran Associates, Inc. https://arxiv.org/abs/2005.14165

Casanueva, I., Vulic, I., Spithourakis, G., & Budzianowski, P. (2022). NLU++: A multi-label, slot-rich, Generalisable dataset for natural language understanding in task-oriented dialogue. In M. Carpuat, M.-C. de Marneffe, & I. V. M. Ruiz (Eds.), *Findings of the Association for Computational Linguistics: NAACL 2022* (pp. 1998–2013). Association for Computational Linguistics. https://doi.org/10.18653/V1/2022.FINDINGS-NAACL.154

Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., & Chua, T. (2017). SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017*, 6298–6306. https://doi.org/10.1109/CVPR.2017.667

Chen, L., Zhou, P., & Zou, Y. (2022). Joint multiple intent detection and slot filling via self-distillation. In *IEEE international conference on acoustics, speech and signal processing, ICASSP 2022* (pp. 7612–7616). IEEE. https://doi.org/10.1109/ICASSP43922.2022.9747843

Chen, Q., Hu, Q., Huang, J. X., He, L., & An, W. (2017). Enhancing recurrent neural networks with positional attention for question answering. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval* (pp. 993–996). Association for Computing Machinery. https://doi.org/10.1145/3077136.3080699

Chen, X., Ghoshal, A., Mehdad, Y., Zettlemoyer, L., & Gupta, S. (2020). Low-resource domain adaptation for compositional task-oriented semantic parsing. In B. Webber, T. Cohn, Y. He, & Y. Liu (Eds.), *Proceedings of the 2020 conference on empirical methods in natural language processing, EMNLP 2020* (pp. 5090–5100). Association for Computational Linguistics. https://doi.org/10.18653/V1/2020.EMNLP-MAIN.413

Cheng, L., Jia, W., & Yang, W. (2021). An effective non-autoregressive model for spoken language understanding. In G. Demartini, G. Zuccon, J. S. Culpepper, Z. Huang, & H. Tong (Eds.), *CIKM '21: The 30th ACM international conference on information and knowledge management, virtual event, Queensland, Australia* (pp. 241–250). ACM. https://doi.org/10.1145/3459637.3482229

Cheng, L., Yang, W., & Jia, W. (2023). A scope sensitive and result attentive model for multi-intent spoken language understanding. In B. Williams, Y. Chen, & J. Neville (Eds.), *Thirty-seventh AAAI conference on artificial intelligence, AAAI 2023, thirty-fifth conference on innovative applications of artificial intelligence, IAAI 2023, thirteenth symposium on educational advances in artificial intelligence, EAAI 2023* (pp. 12691–12699). AAAI Press. https://doi.org/10.1609/AAAI.V37I11.26493

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence Modeling. 1–9. http://arxiv.org/abs/1412.3555

Coucke, A., Saade, A., Ball, A., Bluche, T., Caulier, A., Leroy, D., Doumouro, C., Gisselbrecht, T., Caltagirone, F., Lavril, T., Primet, M., & Dureau, J. (2018). Snips voice platform: An embedded spoken language understanding system for private-by-design voice interfaces. http://arxiv.org/abs/1805.10190

Daha, F. Z., & Hewavitharana, S. (2019). Deep neural architecture with character embedding for semantic frame detection. In *13th IEEE international conference on semantic computing, ICSC 2019* (pp. 302–307). IEEE. https://doi.org/10.1109/ICOSC.2019.8665582

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 Conference of the North American chapter of the Association for Computational Linguistics: Human language technologies* (Vol. 1, pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

Einolghozati, A., Arora, A., Lecanda, L. S.-M., Kumar, A., & Gupta, S. (2021). El Volumen louder por favor: Code-switching in task-oriented semantic parsing. In P. Merlo, J. Tiedemann, & R. Tsarfaty (Eds.), *Proceedings of the 16th conference of the European chapter of the Association for Computational Linguistics: Main volume, EACL 2021* (pp. 1009–1021). Association for Computational Linguistics. https://doi.org/10.18653/V1/2021.EACL-MAIN.87

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211. https://doi.org/10.1207/s15516709cog1402_1

Epure, E. V., Compagno, D., Salinesi, C., Deneckere, R., Bajec, M., & Žitnik, S. (2018). Process models of interrelated speech intentions from online health-related conversations. *Artificial Intelligence in Medicine*, *91*, 23–38. https://doi.org/10.1016/j.artmed.2018.06.007

Firdaus, M., Golchha, H., Ekbal, A., & Bhattacharyya, P. (2021). A deep multi-task model for dialogue act classification, intent detection and slot filling. *Cognitive Computation*, *13*(3), 626–645. https://doi.org/10.1007/S12559-020-09718-4

Goo, C.-W., Gao, G., Hsu, Y.-K., Huo, C.-L., Chen, T.-C., Hsu, K.-W., & Chen, Y.-N. (2018). Slot-gated Modeling for joint slot filling and intent prediction. In M. A. Walker, H. Ji, & A. Stent (Eds.), *Proceedings of the 2018 conference of the north American chapter of the Association for Computational Linguistics: Human language technologies, NAACL-HLT* (Vol. 2, pp. 753–757). Association for Computational Linguistics. https://doi.org/10.18653/V1/N18-2118

Graves, A. (2013). Generating sequences with recurrent Neural Networks 1–43. http://arxiv.org/abs/1308.0850

Graves, A., Mohamed, A. R., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, *3*, 6645–6649. https://doi.org/10.1109/ICASSP.2013.6638947

Graves, A., Wayne, G., & Danihelka, I. (2014). Neural Turing Machines. CoRR, abs/1410.5401. http://arxiv.org/abs/1410.5401

Gupta, S., Shah, R., Mohit, M., Kumar, A., & Lewis, M. (2018). Semantic parsing for task oriented dialog using hierarchical representations. In E. Riloff, D. Chiang, J. Hockenmaier, & J. Tsujii (Eds.), *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 2787–2792). Association for Computational Linguistics. https://doi.org/10.18653/V1/D18-1300

Haihong, E., Niu, P., Chen, Z., & Song, M. (2019). A novel bi-directional interrelated model for joint intent detection and slot filling. In A. Korhonen, D. R. Traum, & L. Màrquez (Eds.), *Proceedings of the 57th conference of the association for computational linguistics, ACL 2019* (Vol. 1, pp. 5467–5471). Association for Computational Linguistics. https://doi.org/10.18653/V1/P19-1544

Hao, X., Wang, L., Zhu, H., & Guo, X. (2023). Joint agricultural intent detection and slot filling based on enhanced heterogeneous attention mechanism. *Computers and Electronics in Agriculture*, *207*(January), 107756. https://doi.org/10.1016/j.compag.2023.107756

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Huang, L., Liang, S., Ye, F., & Gao, N. (2024). A fast attention network for joint intent detection and slot filling on edge devices. *IEEE Transactions on Artificial Intelligence*, *5*(2), 530–540. https://doi.org/10.1109/TAI.2023.3309272

Huang, Z., Liu, F., Zhou, P., & Zou, Y. (2021). Sentiment injected iteratively Co-interactive network for spoken language understanding. In *IEEE international conference on acoustics, speech and signal processing, ICASSP 2021* (pp. 7488–7492). IEEE. https://doi.org/10.1109/ICASSP39728.2021.9413885

Huang, Z., Liu, F., & Zou, Y. (2020). Federated Learning for spoken language understanding. In D. Scott, N. Bel, & C. Zong (Eds.), *Proceedings of the 28th international conference on computational linguistics* (pp. 3467–3478). International Committee on Computational Linguistics. https://doi.org/10.18653/v1/2020.coling-main.310

Iovine, A., Narducci, F., de Gemmis, M., Polignano, M., Basile, P., & Semeraro, G. (2020). A comparison of Services for Intent and Entity Recognition for conversational recommender systems. In P. Brusilovsky, M. de Gemmis, A. Felfernig, P. Lops, J. O'Donovan, G. Semeraro, & M. C. Willemsen (Eds.), *Proceedings of the 7th joint workshop on interfaces and human decision making for recommender systems co-located with 14th ACM conference on recommender systems (RecSys 2020), online event 2020* (Vol. 2682, pp. 37–47). Association for Computational Linguistics. CEUR-WS.org, https://ceur-ws.org/Vol-2682/paper4.pdf

Jbene, M., Raif, M., Tigani, S., Chehri, A., & Saadane, R. (2022). User sentiment analysis in conversational systems based on augmentation and attention-based BiLSTM. *Procedia Computer Science*, *207*, 4106–4112. https://doi.org/10.1016/j.procs.2022.09.473

Jbene, M., Tigani, S., Saadane, R., & Chehri, A. (2022). An LSTM-based intent detector for conversational recommender systems. In *2022 IEEE 95th vehicular technology conference: (VTC2022-spring)* (pp. 1–5). IEEE. https://doi.org/10.1109/VTC2022-Spring54318.2022.9860839

Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., & Liu, Q. (2020). TinyBERT: Distilling BERT for natural language understanding. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the association for computational linguistics: EMNLP 2020 Online Event, EMNLP* (pp. 4163–4174). Association for Computational Linguistics. https://doi.org/10.18653/V1/2020.FINDINGS-EMNLP.372

Laranjo, L., Dunn, A. G., Tong, H. L., Kocaballi, A. B., Chen, J., Bashir, R., Surian, D., Gallego, B., Magrabi, F., Lau, A. Y. S., & Coiera, E. (2018). Conversational agents in healthcare: A systematic review. *Journal of the American Medical Informatics Association*, *25*(9), 1248–1258. https://doi.org/10.1093/jamia/ocy072

Li, C., Zhou, Y., Chao, G., & Chu, D. (2022). Understanding users' requirements precisely: A double Bi-LSTM-CRF joint model for detecting user's intentions and slot tags. *Neural Computing and Applications*, *34*(16), 13639–13648. https://doi.org/10.1007/S00521-022-07171-Y

Li, H., Arora, A., Chen, S., Gupta, A., Gupta, S., & Mehdad, Y. (2021). MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In P. Merlo, J. Tiedemann, & R. Tsarfaty (Eds.), *Proceedings of the 16th conference of the European chapter of the Association for Computational Linguistics: Main volume, EACL 2021* (pp. 2950–2962). Association for Computational Linguistics. https://doi.org/10.18653/V1/2021.EACL-MAIN.257

Liu, B., & Lane, I. (2016). Attention-based recurrent neural network models for joint intent detection and slot filling. In *Proceedings of the annual conference of the international speech communication association* (pp. 685–689). INTERSPEECH. https://doi.org/10.21437/Interspeech.2016-135

Liu, H., Liu, Y., Wong, L.-P., Lee, L.-K., & Hao, T. (2020). A hybrid neural network BERT-cap based on pre-trained language model and capsule network for user intent classification. *CompLex*, *2020*, 8858852:1–8858852:11. https://doi.org/10.1155/2020/8858852

Liu, X., Eshghi, A., Swietojanski, P., & Rieser, V. (2019). Benchmarking natural language understanding Services for Building Conversational Agents. In E. Marchi, S. M. Siniscalchi, S. Cumani, V. M. Salerno, & H. Li (Eds.), *Increasing naturalness and flexibility in spoken dialogue interaction - 10th international workshop on spoken dialogue systems, IWSDS 2019* (Vol. 714, pp. 165–183). Springer. https://doi.org/10.1007/978-981-15-9323-9

Liu, Y., Meng, F., Zhang, J., Zhou, J., Chen, Y., & Xu, J. (2019). CM-net: A novel collaborative memory network for spoken language understanding. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 1051–1060). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1097

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. CoRR. http://arxiv.org/abs/1907.11692

Louvan, S., & Magnini, B. (2020). Simple is better! Lightweight data augmentation for low resource slot filling and intent classification. In M. le Nguyen, M. C. Luong, & S. Song (Eds.), *Proceedings of the 34th Pacific Asia conference on language, information and computation, PACLIC 2020* (pp. 167–177). Association for Computational Linguistics. https://aclanthology.org/2020.paclic-1.20/

Luong, T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In L. Màrquez, C. Callison-Burch, & J. Su (Eds.), *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 1412–1421). Association for Computational Linguistics. https://doi.org/10.18653/v1/D15-1166

Mullick, A. (2023). Novel intent detection and active Learning based classification (student abstract). In B. Williams, Y. Chen, & J. Neville (Eds.), *Thirty-seventh AAAI conference on artificial intelligence, AAAI 2023, thirty-fifth conference on innovative applications of artificial intelligence, IAAI 2023, thirteenth symposium on educational advances in artificial intelligence, EAAI 2023* (pp. 16286–16287). AAAI Press. https://doi.org/10.1609/AAAI.V37I13.27003

Ni, P., Li, Y., Li, G., & Chang, V. (2020). Natural language understanding approaches based on joint task of intent detection and slot filling for IoT voice interaction. *Neural Computing and Applications*, *32*(20), 16149–16166. https://doi.org/10.1007/S00521-020-04805-X

Price, P. J. (1990). Evaluation of spoken language systems: The ATIS domain. In *Proceedings of the workshop on speech and natural language* (pp. 91–95). Association for Computational Linguistics. https://doi.org/10.3115/116580.116612

Qin, L., Che, W., Li, Y., Wen, H., & Liu, T. (2019). A stack-propagation framework with token-level intent detection for spoken language understanding. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 2078–2087). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1214

Qin, L., Che, W., Ni, M., Li, Y., & Liu, T. (2021). Knowing where to leverage: Context-aware graph convolutional network with an adaptive fusion layer for contextual spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *29*, 1280–1289. https://doi.org/10.1109/TASLP.2021.3053400

Qin, L., Liu, T., Che, W., Kang, B., Zhao, S., & Liu, T. (2021). A Co-interactive transformer for joint slot filling and intent detection. In *IEEE international conference on acoustics, speech and signal processing, ICASSP 2021* (pp. 8193–8197). IEEE. https://doi.org/10.1109/ICASSP39728.2021.9414110

Qin, L., Wei, F., Xie, T., Xu, X., Che, W., & Liu, T. (2021). GL-GIN: fast and accurate non-autoregressive model for joint multiple intent detection and slot filling. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *Proceedings of the 59th annual meeting of the Association for Computational Linguistics and the 11th international joint conference on natural language processing, ACL/IJCNLP 2021* (Vol. 1, pp. 178–188). Association for Computational Linguistics. https://doi.org/10.18653/V1/2021.ACL-LONG.15

Qin, L., Xu, X., Che, W., & Liu, T. (2020). AGIF: An adaptive graph-interactive framework for joint multiple intent detection and slot filling. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the association for computational linguistics: EMNLP 2020* (pp. 1807–1816). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.findings-emnlp.163

Ravuri, S., & Stolcke, A. (2015). Recurrent neural network and LSTM models for lexical utterance classification. In *Proceedings of the annual conference of the international speech communication association* (Vol. 2015-January). INTERSPEECH. https://doi.org/10.21437/interspeech.2015-42

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. CoRR, abs/1910.01108. http://arxiv.org/abs/1910.01108

Sarker, I. H. (2021). Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. *SN Computer Science*, *2*(6), 1–20. https://doi.org/10.1007/s42979-021-00815-1

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2673–2681. https://doi.org/10.1109/78.650093

Schuster, S., Gupta, S., Shah, R., & Lewis, M. (2019). Cross-lingual transfer Learning for multilingual task oriented dialog. In J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 conference of the North American chapter of the Association for Computational Linguistics: Human language technologies, NAACL-HLT 2019* (Vol. 1, pp. 3795–3805). Association for Computational Linguistics. https://doi.org/10.18653/V1/N19-1380

Shen, Y., Hsu, Y.-C., Ray, A., & Jin, H. (2021). Enhancing the generalization for intent classification and out-of-domain detection in SLU. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing, ACL/IJCNLP 2021* (Vol. 1, pp. 2443–2453). Association for Computational Linguistics. https://doi.org/10.18653/V1/2021.ACL-LONG.190

Shivnikar, S. D., Arora, H., & Harichandana, B. S. S. (2021). A character representation enhanced on-device Intent Classification. CoRR, abs/2101.04456. https://arxiv.org/abs/2101.04456

Song, Y., Zhao, J., Koehler, S., Abdullah, A., & Harris, I. G. (2023). PCMID: Multi-intent detection through supervised prototypical contrastive Learning. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023* (pp. 9481–9495). Association for Computational Linguistics. https://doi.org/10.18653/V1/2023.FINDINGS-EMNLP.636

Sowanski, M., & Janicki, A. (2020). Leyzer: A dataset for multilingual virtual assistants. In P. Sojka, I. Kopecek, K. Pala, & A. Horák (Eds.), *Text, speech, and dialogue - 23rd international conference, TSD 2020* (Vol. 12284, pp. 477–486). Springer. https://doi.org/10.1007/978-3-030-58323-1

Sun, Z., Yu, H., Song, X., Liu, R., Yang, Y., & Zhou, D. (2020). MobileBERT: A compact task-agnostic BERT for resource-limited devices. In D. Jurafsky, J. Chai, N. Schluter, & J. R. Tetreault (Eds.), *Proceedings of the 58th annual meeting of the Association for Computational Linguistics, ACL 2020* (pp. 2158–2170). Association for Computational Linguistics. https://doi.org/10.18653/V1/2020.ACL-MAIN.195

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*, *4*-(January), 3104–3112. http://arxiv.org/abs/1409.3215

Tan, Z., Wang, M., Xie, J., Chen, Y., & Shi, X. (2018). *Deep semantic role labeling with self-attention. 32nd AAAI conference on artificial intelligence 2018* (Vol. 2015, pp. 4929–4936). AAAI. https://doi.org/10.1609/aaai.v32i1.11928

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Canton-Ferrer, C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., … Scialom, T. (2023). Llama 2: Open foundation and fine-tuned chat models. CoRR, abs/2307.09288. https://doi.org/10.48550/ARXIV.2307.09288

Tu, N. A., Uyen, H. T. T., Phuong, T. M., & Bach, N. X. (2023). Joint multiple intent detection and slot filling with supervised contrastive Learning and self-distillation. In K. Gal, A. Nowé, G. J. Nalepa, R. Fairstein, & R. Radulescu (Eds.), *ECAI 2023 - 26th European conference on artificial intelligence, September 30–October 4, 2023, Kraków, Poland - Including 12th conference on prestigious applications of intelligent systems (PAIS 2023)* (Vol. 372, pp. 2370–2377). IOS Press. https://doi.org/10.3233/FAIA230538

Tür, G., Hakkani-Tür, D., & Heck, L. P. (2010). What is left to be understood in ATIS? In D. Hakkani-Tür & M. Ostendorf (Eds.), *2010 IEEE spoken language technology workshop, SLT 2010* (pp. 19–24). IEEE. https://doi.org/10.1109/SLT.2010.5700816

Van der Goot, R., Sharaf, I., Imankulova, A., Üstün, A., Stepanovic, M., Ramponi, A., Khairunnisa, S. O., Komachi, M., & Plank, B. (2021). From masked language Modeling to translation: Non-English auxiliary tasks improve zero-shot spoken language understanding. In K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tür, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, & Y. Zhou (Eds.), *Proceedings of the 2021 conference of the North American chapter of the Association for Computational Linguistics: Human language technologies, NAACL-HLT 2021* (pp. 2479–2497). Association for Computational Linguistics. https://doi.org/10.18653/V1/2021.NAACL-MAIN.197

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *2017*, 5999–6009. http://arxiv.org/abs/1706.03762

Wei, P., Zeng, B., & Liao, W. (2022). Joint intent detection and slot filling with wheel-graph attention networks. *Journal of Intelligent Fuzzy Systems*, *42*(3), 2409–2420. https://doi.org/10.3233/JIFS-211674

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Learning, Y. B. B., & T.-P. of the 32nd I. C. on M. (2015). In F. Bach & D. Blei (Eds.), *Show, attend and tell: Neural image caption generation with visual attention* (Vol. 37, pp. 2048–2057). PMLR. http://proceedings.mlr.press/v37/xuc15.pdf

Xue, S., & Ren, F. (2021). Intent-enhanced attentive Bert capsule network for zero-shot intention detection. *Neurocomputing*, *458*, 1–13. https://doi.org/10.1016/J.NEUCOM.2021.05.085

Yang, P., Ji, D., Ai, C., & Li, B. (2021). AISE: Attending to intent and slots explicitly for better spoken language understanding. *Knowledge-Based Systems*, *211*, 106537. https://doi.org/10.1016/j.knosys.2020.106537

Yin, S., Huang, P., & Xu, Y. (2024). Uni-MIS: United multiple intent spoken language understanding via multi-view intent-slot interaction. In M. J. Wooldridge, J. G. Dy, & S. Natarajan (Eds.), *Thirty-eighth AAAI conference on artificial intelligence, AAAI 2024, thirty-sixth conference on innovative applications of artificial intelligence, IAAI 2024, fourteenth symposium on educational advances in artificial intelligence, EAAI 2014* (pp. 19395–19403). AAAI Press. https://doi.org/10.1609/AAAI.V38I17.29910

Yin, S., Huang, P., Xu, Y., Huang, H., & Chen, J. (2024). Do large language model understand multi-intent spoken language? CoRR, abs/2403.04481. https://doi.org/10.48550/ARXIV.2403.04481

Ying, H., Zhuang, F., Zhang, F., Liu, Y., Xu, G., Xie, X., Xiong, H., & Wu, J. (2018). *Sequential recommender system based on hierarchical attention network* (pp. 3926–3932). IJCAI International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2018/546

Yolchuyeva, S., Németh, G., & Gyires-Tóth, B. (2019). Self-attention networks for intent detection. In R. Mitkov & G. Angelova (Eds.), *Proceedings of the international conference on recent advances in natural language processing, RANLP 2019* (pp. 1373–1379). INCOMA Ltd. https://doi.org/10.26615/978-954-452-056-4

Yoon, Y., Lee, J., Kim, K., Park, C., & Kim, T. (2024). BlendX: Complex multi-intent detection with blended patterns. In N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, & N. Xue (Eds.), *Proceedings of the 2024 joint international conference on computational linguistics, language resources and evaluation, LREC/COLING 2024* (pp. 2428–2439). ELRA and ICCL. https://aclanthology.org/2024.lrec-main.218

You, Q., Jin, H., Wang, Z., Fang, C., & Luo, J. (2016). Image captioning with semantic attention. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016*, 4651–4659. https://doi.org/10.1109/CVPR.2016.503

Zhang, C., Li, Y., Du, N., Fan, W., & Yu, P. S. (2019). Joint slot filling and intent detection via capsule neural networks. In A. Korhonen, D. R. Traum, & L. Màrquez (Eds.), *Proceedings of the 57th conference of the Association for Computational Linguistics, ACL 2019* (Vol. 1, pp. 5259–5267). Association for Computational Linguistics. https://doi.org/10.18653/V1/P19-1519

Zhao, S., Gupta, R., Song, Y., & Zhou, D. (2021). Extremely small BERT models from mixed-vocabulary training. In P. Merlo, J. Tiedemann, & R. Tsarfaty (Eds.), *Proceedings of the 16th conference of the European chapter of the Association for Computational Linguistics: Main volume, EACL 2021* (pp. 2753–2759). Association for Computational Linguistics. https://doi.org/10.18653/V1/2021.EACL-MAIN.238

Zhong, V., Xiong, C., & Socher, R. (2018). *Global-locally self-attentive encoder for dialogue state tracking*. Annual Meeting of the Association for Computational Linguistics. https://doi.org/10.18653/v1/P18-1135

AUTHOR BIOGRAPHIES

Mourad Jbene received his Bachelor's degree in Computer Science and Mathematics in 2017, followed by a Master's degree in Big Data Analytics and Smart Systems (BDSaS) in 2019, both from USMBA, Fez, Morocco. He is currently pursuing a Ph.D. at the Hassania School of Public Works, within the Laboratory of Systems Engineering (Intelligent Systems and Sensor Networks team) in Casablanca, Morocco. His research interests include interactive and conversational recommendation systems, information retrieval, and natural language processing.

Dr Abdellah. Chehri is a Professor at the Department of Mathematics and Computer Science at the Royal Military College of Canada (RMC), Kingston, Ontario. Dr. Chehri completed his Ph.D. at University Laval (Quebec, Canada) and his Master's studies at University Nice-Sophia Antipolis-Eurecom (France). Dr. Chehri is a co-author of more than 200 peer-reviewed publications in established journals and conference proceedings sponsored by established publishers such as IEEE, ACM, Elsevier, and Springer. Dr. Chehri has served on roughly thirty conference and workshop program committees. In addition, he served as guest/associate editor for several well-reputed journals. Dr Chehri is a Senior Member of IEEE, a member of the IEEE Communication Society, IEEE Vehicular Technology Society (VTS), and IEEE Photonics Society.

Rachid Saadane holds a BC.S. (2001), M.S. (2003) from Mohamed V University, and a Ph.D. (2007) from the University of Mohamed V in collaboration with Eurecom Institute. Currently, he serves as a full professor in the Department of Electrical Engineering at the Hassania School of Public Works (HSPW or EHTP). From March 2003 to July 2006, he worked at Eurecom Institute, France, where he focused on developing a framework for UWB channel characterization and modeling as a research engineer. His research interests span various areas including Wireless Communications Systems (Reflective Intelligent Surface, Massive MIMO, SC-FDMA, OFDM, Dynamic Spectrum Detection, and Radio

Cognitive), Big Data, System Recommenders, Artificial Intelligence, Machine Learning, Deep Learning, Signal Processing, Image Processing, Estimation Theory, Smart City, and Smart Agriculture Applications, as well as Smart Farming. Additionally, he has contributed significantly to reliability in UWB Communication Systems. He was honored with the SCA'19 Award for the best paper at the Smart Cities Applications Conference in 2019. With 139 publications and more than 400 verified peer reviews, his profile on web of science reflects his active involvement in academia. Furthermore, he serves as the Head Laboratory for the LaGeS laboratory at HSPW (from 2018 to present). Currently, he holds the position of Deputy Director in charge of research, cooperation, and partnership at HSPW.

Dr. Smail TIGANI is a Digital Engineering expert with extensive experience in AI, Data Science, and Web Technology. He has served as a professor, researcher, and director in several higher education institutions (HEIs), including EIDIA and FEMG at the Euromed University, as well as EMSI and others. His career also spans roles in various IT companies, where he has applied his expertise to drive innovation. In addition to his academic and research roles, Dr. TIGANI is the founder, CEO, and CTO of Accsellium LLC, a digital engineering and AI company. His work focuses on Machine Learning Algorithms and Analytical Software Design, where he integrates theory with practical applications to create real-world impact.

Gwanggil Jeon received the B.S., M.S., and Ph.D. (summa cum laude) degrees from the Department of Electronics and Computer Engineering, Hanyang University, Seoul, Korea, in 2003, 2005, and 2008, respectively. He was with the School of Information Technology and Engineering, University of Ottawa, Ottawa, ON, Canada, as a Post-Doctoral Fellow, from 2009 to 2011. He was with the Graduate School of Science and Technology, Niigata University, Niigata, Japan, as an assistant professor, from 2011 to 2012. He was a prestigious visiting professor in the Dipartimento di Informatica, Università degli Studi di Milano Statale, from 2019 to 2020. He is currently a professor at Xidian University, Xi'an, China and Incheon National University, Incheon, Korea. Dr. Jeon was a recipient of the IEEE Chester Sall Award in 2007 and the ETRI Journal Paper Award in 2008.

How to cite this article: Jbene, M., Chehri, A., Saadane, R., Tigani, S., & Jeon, G. (2025). Intent detection for task-oriented conversational agents: A comparative study of recurrent neural networks and transformer models. *Expert Systems*, *42*(2), e13712. https://doi.org/10.1111/exsy.13712