Practical Guidelines for Intent Recognition: BERT with Minimal Training Data Evaluated in Real-World HRI Application

Matthew Huggins
MIT Media Lab
Cambridge, MA, USA
hugginsm@mit.edu

Sharifa Alghowinem
MIT Media Lab
Cambridge, MA, USA
sharifah@media.mit.edu

Sooyeon Jeong
MIT Media Lab
Cambridge, MA, USA
sooyeon6@media.mit.edu

Pedro Colon-Hernandez
MIT Media Lab
Cambridge, MA, USA
pe25171@media.mit.edu

Cynthia Breazeal
MIT Media Lab
Cambridge, MA, USA
cynthiab@media.mit.edu

Hae Won Park
MIT Media Lab
Cambridge, MA, USA
haewon@media.mit.edu

ABSTRACT

Intent recognition models, which match a written or spoken input's class in order to guide an interaction, are an essential part of modern voice user interfaces, chatbots, and social robots. However, getting enough data to train these models can be very expensive and challenging, especially when designing novel applications such as real-world human-robot interactions. In this work, we first investigate how much training data is needed for high performance in an intent classification task. We train and evaluate BiLSTM and BERT models on various subsets of the ATIS and Snips datasets. We find that only 25 training examples per intent are required for our BERT model to achieve 94% intent accuracy compared to 98% with the entire datasets, challenging the belief that large amounts of labeled data are required for high performance in intent recognition. We apply this knowledge to train models for a real-world HRI application, character strength recognition during a positive psychology interaction with a social robot, and evaluate against the Character Strength dataset collected in our previous HRI study. Our real-world HRI application results also confirm that our model can produce 76% intent accuracy with 25 examples per intent compared to 80% with 100 examples. In a real-world scenario, the difference is only one additional error per 25 classifications. Finally, we investigate the limitations of our minimal data models and offer suggestions on developing high quality datasets. We conclude with practical guidelines for training BERT intent recognition models with minimal training data and make our code and evaluation framework available for others to replicate our results and easily develop models for their own applications.

CCS CONCEPTS

• **Computing methodologies** → **Natural language processing**; • **Human-centered computing** → **Systems and tools for interaction design**.

HRI '21, March 8–11, 2021, Boulder, CO, USA.
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8289-2/21/03.
https://doi.org/10.1145/3434073.3444671

KEYWORDS

intent recognition; natural language processing; BERT; real-world HRI application

ACM Reference Format:
Matthew Huggins, Sharifa Alghowinem, Sooyeon Jeong, Pedro Colon-Hernandez, Cynthia Breazeal, and Hae Won Park. 2021. Practical Guidelines for Intent Recognition: BERT with Minimal Training Data Evaluated in Real-World HRI Application. In *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction (HRI '21), March 8–11, 2021, Boulder, CO, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3434073.3444671

1 INTRODUCTION

Voice user interfaces (VUI) (e.g. Amazon Alexa, chatbots, and social robots) are becoming an essential part of everyday life [5, 15]. For these systems to carry out effective dialogue, they must be able to determine the intent behind a user's spoken utterance. For the purpose of this paper, intent recognition is defined as commonly understood in the NLP community, i.e. the task of taking a written or spoken input, and determining which of several classes it matches in order to best respond or guide the interaction, not to be confused with the broader meaning in the HRI context, i.e. inferring goals of the user based on their observed actions from sensors or visual cues. This type of intent recognition is essential to building complex conversational experiences in HRI, which is a key challenge. While rule-based parsing is a common approach for some interactions, it is not effective for more advanced and novel dialogue contexts [27]. To improve user experience while interacting with such systems, state-of-the-art models are trained using large labeled datasets for intent recognition customized to specific applications [8, 9, 26].

Even though the importance of intent recognition models is clear, it can be very challenging to build effective intent recognition models. The amount of data typically used to train these systems can be prohibitively expensive or difficult to acquire for a new application. Models in research are trained on thousands of example utterances [24], which is far more than a small organization can easily manage to prepare, let alone a student or sole developer working on a new project. Such challenge is prevalent in HRI applications, as gathering large amount of data that matches the exact social context the robot is deployed to is often impossible. Social robot applications often involve interactions in private settings such as homes or hospital rooms [13, 20–22, 31, 35], and the dialogue contexts can be highly uncommon topics for large public dataset to exist. Even though there is a great need for understanding how high-performing intent recognition models can be trained with small amounts of data, this area has largely been unexplored.

In this paper, we propose practical guidelines for building intent recognition models that can be deployed in real-world applications such as HRI dialogue scenarios. We try to answer three key questions: 1) What is the "minimal effective dose" of training examples for intent recognition? How many training examples are actually needed for high performance in typical applications? 2) Can we achieve high performance using model architectures that are easy for anyone to setup and customize? 3) How well can our methods generalize to a more complex, real-world HRI application?

We first demonstrate that much fewer training examples are needed than apparent, and then investigate how these models' performances on small datasets can be further improved by understanding their dependence on specific syntactic and semantic features. We evaluate these findings by collecting examples to train models for character strength identification, evaluated on data previously collected through a real-world HRI interaction in which college students engaged in positive psychology intervention sessions with a social robot coach. The key contributions of this paper are as follows:

- A framework for evaluating how much training data is needed to effectively train intent recognition models for real-world applications.
- Practical guidelines to collecting, building, and testing these models on a real-world HRI application.
- An analysis of the strengths and limitations of our models, and recommendations for training intent recognition models in limited-data scenarios.

2 BACKGROUND

Intent Classification: Traditionally, intent classification systems were based on keywords [30] or Context Free Grammars [11]. Recently, many deep learning approaches have been explored, such as convolutional neural networks (CNN) [46], long short-term memory (LSTM) [34], and attention-based CNN [47]. Typically these approaches require a large labeled datasets to achieve a state-of-the-art performance. While intent classification and slot filling are often done as separate tasks, some recent work has also focused on creating models that do both tasks together. Liu and Lane [25] proposed an attention-based Bi-directional Recurrent Neural Network (BiRNN) model for joint intent classification and slot filling. Liu et al. [26] used a Collaborative Memory Network to capture slot-specific and intent-specific features in order to enhance local context representations. Finally, a BERT-based model that combines intent classification and slot filling into a single token classification task was presented in [7].

Several commercial intent recognition tools offer on-the-go and easy access solutions, but are limited in the flexibility they provide. Google's Dialogflow [3], Microsoft's LUIS [44] and Amazon Lex [2] allow users to create custom intents and upload example utterances but their models or architectures cannot be customized or modified, limiting the range of tasks these tools can be used for [6]. The Amazon Skills Kit [1] also allows users to create custom intents and provide training examples, but its use is limited to the Alexa ecosystem. Moreover, these cloud APIs require an internet connection, which may not be feasible depending on the application. With the tool we present in this paper, HRI designers and developers will be able to take advantage of better (see Results) intent recognition capabilities than these APIs offer, but in a way that is completely customizable, free of charge, and offline if needed.

Few shot learning: Few shot learning refers to the practice of training a model with a very small amount of data[43]. Previous work has explored techniques for creating models that better handle low-data scenarios, including memory modules to help neural networks learn from rare events [49], and matching networks that leverage several labeled support examples to do one-shot learning for a new input [41]. While exploration of training intent recognition models with minimal training data has been limited, Luo et al. [28] proposed combining regular expressions with a bidirectional LSTM in order to improve performance when training with small amounts of data. While this approach is effective at improving performance in a few-shot learning scenario, it requires the additional work of an expert writing the appropriate regular expressions.

Novelty of Our Approach: Unlike previous work, our main focus is empowering more people to be able to train their own intent recognition models. Previous approaches are often trained on large amounts of data, or mall amounts of training data is augmented with additional expert information [28]. We aim to demonstrate that high-performing models can be trained with nothing but small amounts of typical training examples, so that anyone can generate enough data to train a high-performing model on their own. In order to achieve high performance with limited data, we take advantage of BERT's pre-training of Deep Bidirectional Transformers [12]. By leveraging its powerful pre-trained internal language representation, the BERT model used in our framework can be effectively trained for intent recognition with much less data than a typical neural model.

3 EVALUATING INTENT RECOGNITION WITH MINIMAL DATA

We evaluate our approaches on two different benchmark datasets: ATIS [9], a corpus of airline travel information phone requests, and Snips [8], which contains utterances of individuals requests/queries to their VUI (e.g. playing music). In addition to ATIS and Snips datasets, we applied our methods to a real-world HRI application, which is described in Section 4. In order to evaluate how the amount of training data affects the performance of our models, we created a series of smaller training sets from both the ATIS and Snips datasets. To create the smaller training sets, *n* training examples of each intent were randomly taken from the respective training set (either ATIS or Snips) in order to create a new partial training set. If an intent had less than *n* training examples, all of the available examples were used, resulting in fewer than *n* examples for that intent. Because of the class imbalance present in ATIS, partial ATIS training sets with larger *n* have many intents with less than *n* examples present.

This setup allows for a practical evaluation of our models. By using the full training sets, we can understand the best possible performance of each model, but by using several smaller training sets, we can observe how the amount of training data impacts each model's performance. This allows us to not only determine what model is best with an abundance of data, but also which models can be effectively used in scenarios with limited training data. The code for our models and the testing framework are publicly available[1].

3.1 Corpora

The ATIS Dataset. ATIS [9] is a dataset of airline-related phone requests that is a commonly used benchmark for intent recognition models. It consists of utterances about air travel information, such as '*What is the flight number?*' or '*What is the destination of flight 87?*'. There are 4958 training utterances, from which we remove 20% for validation, and 893 test utterances. In the dataset, some utterances are assigned multiple intents (e.g. "flight+flight no"). Since only one intent label can be given, we use the most common intent of the ones provided as the true intent label. In the dataset, there is a strong class imbalance, with the majority of examples having the "Flight" intent, and many intents having less than 50 training examples in total. On average, each intent has 276 examples in the training set, with a median of 77 examples (min 1, max 3688).

The Snips Dataset In addition to ATIS, we also use the Snips dataset [8], which was collected through crowdsourcing for the Snips personal voice assistant. There are 7 unique intent classes for the training set, on a variety of topics including playing music, restaurant reservations, and getting the weather (e.g. '*Book an Italian place with a parking for my grand father and I*' and '*Which movie theater is playing The Good Will Hunting nearby?*'). The training set contains 13,084 utterances, and separate validation and test sets that contain 700 utterances each. Unlike ATIS, the intent classes are very balanced in the dataset, each with about 700 training examples.

3.2 Models

In this section, we describe our approach for fine-tuning both BiLSTM and BERT models for intent recognition. As a baseline, we also use a logistic regression model that takes the concatenated GloVe embeddings [32] of the utterance as input. We chose BiLSTM and logistic regression models as baselines because 1) they are smaller models that we may expect to perform better with very small amounts of data when compared to models with more parameters, and 2) they are easy approaches that someone with limited experience in machine learning research could easily attempt, serving our goal to help more researchers create their own intent recognition systems. While our BERT approach is much more complex, it can be easily replicated by using an open-source pre-trained BERT base [19].

BiLSTM. A diagram of our BiLSTM model can be found in Figure 1a. First, the input utterance is tokenized and padded to d_{pad} tokens, which is the length of the longest utterance in the training dataset. Each token is then replaced by its GloVe embedding, resulting in a d_{pad} by d_{embed} (300) input matrix. Padding and out of vocabulary (OOV) tokens are represented by a vector of zeros. The input sequence is fed through a bidirectional LSTM [18]. The hidden states of each LSTM are concatenated to form the encoding, with shape d_{pad} by twice the hidden dimension. To create the intent prediction, the encoding is multiplied at each time step by a trainable parameter matrix with bias, resulting in an output sequence of shape d_{pad} by the number of possible intents (18 for ATIS, 7 for Snips). The result of the linear transform is then flattened, followed by a Dense layer with ReLU activation. Finally, Softmax is applied to create a vector of the intent probabilities. In total, our BiLSTM model has 1.5M trainable parameters. In our experiments, the BiLSTM and BERT models are trained using categorical cross entropy loss and the ADAM optimizer [23].

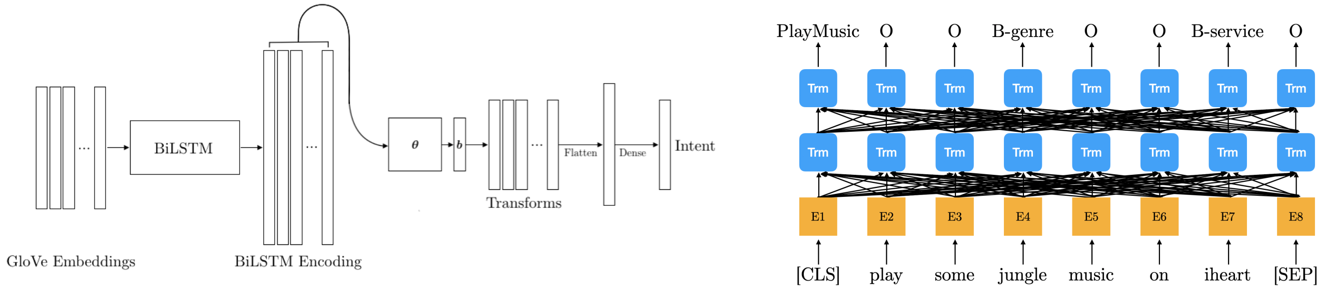
BERT. We also use a BERT-based model for intent classification, similar to the one proposed by Chen et al. [7]. BERT is state-of-the-art language modelling architecture developed by Google [12], which applies a Masked Language Modeling objective to perform bidirectional training on a stack of self attention-based layers called "transformers"[40]. The end system generates vector embeddings which give an effective contextual representation of an input text. BERT has been highly successful due to its performance, modularity, and ease of transfer learning (i.e. it is easily customized for a particular task such as intent recognition). Figure 1b shows an overview of our BERT architecture. In the model, an input utterance is tokenized using a WordPiece tokenizer [45]. This tokenized representation is then passed through a stack of encoder transformer layers (12 in our model). The special token "[CLS]" is placed before the start of the utterance by the tokenizer, and the output of this token, after the final transformer layer, is passed through a fully connected layer (called the pooling layer), and then finally passed through a fully-connected layer, followed by softmax, to create the intent predictions. In our experiments, we use the English uncased BERT base model [39], with 12 layers, with 768-dimensional hidden layers, and 12 attention heads. This model contains 110M parameters and is pre-trained on the BookCorpus (800M words) [48] and on English Wikipedia (2,500M words). We fine-tune the model end-to-end by minimizing the cross-entropy loss on our datasets. Starting with the pre-trained BERT weights, the entire model is trained, including all transformer layers, the pooling layer, and the final dense output layer, which is consistent with typical BERT fine-tuning for specific tasks.

3.3 Experiments

In our first experiments, we evaluate all of our models (logistic regression, BiLSTM, and BERT) on the complete ATIS and Snips datasets, in order to determine their best possible performances. In our second set of experiments, we evaluate our models on subsets of the ATIS and Snips datasets, varying the total number of training examples used, as described above in Section 3. For example, we trained our BERT model on a subset of the ATIS dataset with at most 10 training examples per intent, chosen from the complete training set at random. We then evaluate how the amount of data used affects each model's performance, in order to understand how much data is needed to create a strong model that is ready to be deployed to a real-world application. Moreover, to show the power of BERT end-to-end fine-tuning, we compare the results when the pre-trained BERT weights are held constant and only train the output layer. Finally, we evaluate the consistency of our BERT model's performance when using different randomly chosen partial training sets of the same size. For each value of n for each corpus, we create 10 new partial datasets, in addition to the original ones from our previous experiments. We then train BERT models on each new dataset, and evaluate the consistency of results across different random selections.

Entire Dataset BiLSTM, logistic regression, and BERT models were trained on the entire ATIS training set (with 20% of examples set aside for validation), as well as on the Snips training set. We conducted hyper-parameter sweeps and selected the best models

[1] https://github.com/mitmedialab/bert-slu



(a) Overview of our BiLSTM Model. GloVe embeddings representing the input sequence are fed through a bidirectional LSTM. A linear transform is applied to the concatenated hidden states, and is then flattened and used as input to a dense layer to create the intent prediction.

(b) Overview of our BERT Model. The input utterance is tokenized, and the "[CLS]" and "[SEP]" tokens are added. The label of the first token ("[CLS]") is used for the intent classification.

Figure 1: Overview of the Models used in this Work

based on their intent classification performance on the validation set. For ATIS, the best parameters were used to train models on the whole training set (validation included). Test performances of these models are reported in Table 1, alongside the results of Wang et al. [42], which is the state of the art at the time of writing. For the logistic regression models, the batch size was varied between 16 and 32. For the BiLSTM, the batch size was varied between 16 and 32, the LSTM hidden dimension between 200, 300, and 400, and the LSTM dropout between 0, 0.25, and 0.5. The best logistic regression model was trained for 15 epochs with a batch size of 16, and the best BiLSTM model was trained for 15 epochs with batch size 16, LSTM dimension 300, and dropout 0.25. For the BERT models, a batch size of 32 was used, and the number of epochs was chosen from between 1 and 65, in increments of 5 (1, 5, 10, etc.).

Partial Datasets As described above in Section 3, partial datasets were created from both the ATIS and Snips datasets separately with at most n training examples per intent, with $n \in [1,5,10,25,50,100]$. Logistic regression, BiLSTM, and BERT models were trained on each partial dataset. For each partial training set, hyper-parameter sweeps were done for both the logistic regression and BiLSTM models. The LSTM hidden dimension was varied between 200, 300, and 400, and the LSTM dropout between 0, 0.25, and 0.5. Batch sizes for the logistic regression and BiLSTM models were varied depending on the size of the training set, with the smallest batch size of 1, and the largest of 64. For the BERT models, a batch size of 32 was used, and the number of epochs was chosen from between 1 and 65, in increments of 5 (1, 5, 10, etc.). Best models were chosen by intent classification performance on the appropriate validation set. For the majority of the partial datasets, the best BiLSTM hidden dimension was 300, the best batch size was 16, and the best dropout was 0.25.

3.4 Results

Entire Dataset. Results for our models trained on the entire ATIS and Snips datasets can be found in Table 1. When training on the entire ATIS dataset, our logistic regression model achieved 89.6% accuracy. Our BiLSTM model's highest accuracy was 96.6%. Finally, the BERT model achieved 98.4%. The current state of the art intent accuracy by Liu et al. (2019) is 99.1%. For the Snips dataset, the BiLSTM slightly outperformed BERT model with 98.3% (compared to 98.0%). The logistic regression model achieved 95.9% intent accuracy.

Partial Datasets The results for best partial dataset models are shown in Table 2. On both corpora, the BERT models were quickly able to achieve very high performance with very little data. With at most 25 examples per intent, the BERT model achieved 94.6% intent classification accuracy on the ATIS dataset, and 94.0% for the Snips dataset. With at most 50 examples per intent, the BERT intent performance increased to 96.7% on ATIS and 95.9% on Snips.

Batch Sizes In our baselines, we explored various batch sizes similarly to the other hyperparameters. We found no consistent performance effects for different batch sizes of 8, 16, and 32. However for smaller batch sizes (1, 4) we typically get worse results. This is not surprising as usually batch size has little effect on performance, as long as it is not too small or large, however larger batch sizes require less training iterations [16, 38]. We recommend a batch size of 16 for the BiLSTM, as the model trains quickly with good performance so there is no need to increase the batch size further. For the BERT model, we chose a batch size of 32 as it results in faster training (on a CPU), and no significant performance difference between 16 and 32 was found in our BiLSTM experiments.

BERT Consistency The minimum, mean, and maximum test accuracies for BERT models trained on partial datasets of each size are shown in Table 3. While performance across the 11 random training sets was highly varied for very low numbers of examples, the variation reduced significantly at 25 examples per intent.

BERT Fine-Tuning For our tests, we evaluated the performance of fine-tuning all of the components of a pre-trained BERT model. Without fine-tuning the entire model, performance is much worse, especially when using less training data. To showcase this, we compared the effects of freezing the transformer layers, the pooling layer, and only training the final output layer against fine-tuning the complete model. This comparison can be seen in Table 4. The best test performance on the ATIS dataset fine-tuning only the final output layer, using the entire training set, was only 72.4% vs. 98.4% with complete fine-tuning. When trained on 25 samples/intent, the partially fine-tuned model achieves only 3.9% vs. 94.6% with the complete fine-tuning. The discrepancy persists with 100 samples/intent at 24.2% vs. 96.1% respectively. Tests on the Snips dataset had similar results, the best test performance using the entire training set, was only 73.5% vs. 98.0%. When trained on 25 samples/intent, the model achieves only 21.4% vs. 94.0% and the discrepancy persists with 100

Table 1: Entire Dataset Results

Model	LR	BiLSTM	BERT	Wang et al.	Chen et al.	Liu et al.
ATIS	89.6	96.6	**98.4**	98.9	97.9	99.1
Snips	95.9	**98.3**	98.0	-	98.6	99.3

Test set accuracies on the entire ATIS and Snips datasets. LR refers to the logistic regression baseline.

Table 2: Partial Dataset Intent Accuracy Results

Examples	**ATIS**			**Snips**		
Per Intent	LR	BiLSTM	BERT	LR	BiLSTM	BERT
1	14.1	**34.7**	14.5	33.9	35.6	**55.7**
5	35.3	46.6	**51.5**	59.6	78.4	**80.6**
10	41.7	66.2	**82.5**	70.4	85.7	**89.3**
25	62.5	81.7	**94.6**	78.1	90.7	**94.0**
50	72.5	89.1	**96.7**	85.9	93.3	**95.9**
100	73.1	92.7	**96.1**	87.7	93.0	**96.6**
Entire	89.6	96.6	**98.4**	95.9	**98.3**	98.0

Results for the entire datasets are included for reference. Best accuracies for each partial training dataset are in bold.

Table 3: Variation in Performance for BERT models

Examples	**ATIS**			**Snips**		
Per Intent	Min	Mean	Max	Min	Mean	Max
1	5.8	13.8±7.2	27.0	33.6	49.7±6.7	57.7
5	43.6	59.5±12.1	73.8	75.3	80.4±2.5	84.3
10	68.5	82.7±6.7	93.4	81.8	90.2±3.2	93.0
25	93.2	94.7±1.1	96.9	92.5	94.3±1.0	95.2
50	94.9	96.2±0.8	97.4	95.2	95.7±0.4	96.3
100	95.9	96.8±0.6	97.8	95.7	96.5±0.5	97.2

For each unique *n* examples per intent on both ATIS and Snips, 10 partial training sets were randomly chosen in addition to the training sets used for results in Table 2. BERT models were trained on each partial dataset, and the minimum, mean, and maximum test accuracy for each *n* are reported here.

Table 4: BERT Fine-Tuning Effects

Examples	**ATIS**		**Snips**	
Per Intent	Partial	Complete	Partial	Complete
1	4.8	2.1	12.1	13.5
5	3.5	11.7	13.8	30.9
10	3.9	55.7	20.1	89.3
25	3.9	94.6	21.4	94.0
50	9.5	96.7	26.0	95.9
100	24.2	96.1	31.0	96.6
All	72.4	98.4	73.5	98.0

We evaluated the effects of fine-tuning only the output layer (Partial) of our BERT system vs. the complete system consisting of the dense output layer, the pooling layer, and the pre-trained BERT transformer layers (Complete)

samples/intent with a test accuracy of 31.0% vs. 96.6% for the partially fine-tuned model and the completely fine-tuned model respectively.

4 EVALUATION: REAL WORLD APPLICATION

The use of conversational agents for therapeutic interventions has been growing rapidly [10]. Social robots have been developed for various health-related tasks as well, such as supporting older adults with dementia [37], serving as a home fitness coach [17], acting as a pediatric companion [21] and a life coach [4]. In order to provide engaging and fluent interactions, it is important for these robots to understand users' intents during interactions. However, it is commonly believed that creating a high-performing model for any natural language understanding task requires large labeled datasets in order to generalize well in real-time deployments [28]. The lack of such large datasets in novel interventions can hinder researchers' ability to achieve the desired outcomes, and limits user experience.

To test our guidelines for low resource tasks, and perform evaluation of our models, we used a dataset collected through a real-world deployment study. In our previous study [20], we deployed a portable robot station, which comprised of a tablet and a social robot Jibo (see Figure 2), to the dorms of 42 undergraduate students. In the study, the robot delivered daily positive psychology sessions over one week. The social robot was designed to improve participants' psychological well-being and mood through various positive psychology interventions [36], including character strengths. Peterson et al. lists 24 positive parts of personality that impact how people think, feel and behave [33]. Character-strength-based coaching has been growing extensively in the last few decades, because of its benefit of empowering people to increase their awareness of their signature strengths, so that they can better use them in everyday life [29]. The intention of the character-strength session in our intervention is to provide an automated way of delivering such coaching by the robot.

4.1 Character Strength Testing Dataset

While our results on the ATIS and Snips datasets are very promising, we want to demonstrate that the same methods can be effectively applied to a real-world HRI application. As mentioned above, the social robot delivered seven positive psychology sessions, two of which covered contents about character strengths. In the first character strengths session, participants learned about character strengths and were given an opportunity to identify their own signature strengths through a dialogue with the robot. In the second session, the participants were asked by the robot about the new ways they could use their signature strengths to improve their well-being and were encouraged to practice using their strengths. The verbal responses from the participants were recorded and transcribed post study. At the time of the study, the robot was unable to parse which particular character strength the participant was referring to. Instead, it listened to the user and provided generic responses and encouragements. After the study, three annotators labeled the dataset for which character strength each utterance was referring to. We refer to this labeled dataset as the Character Strengths dataset. In total, 118 utterances were collected across 35 participants. We used Fleiss kappa agreement for multi-raters [14] to measure inter-rater reliability (IRR). The confidence interval among the three annotators was 0.70. The cutoffs for qualitative IRR based on Fleiss kappa values is considered a "*substantial agreement*" if the value is between 0.61 and 0.80 [14]. Whenever there was a disagreement between the annotators, the majority voting was selected as the final labels. For the eight utterances that the annotators did not agree on, five were cross-checked with other utterances from the same participant, two utterances acquired a 4th annotation, and one utterance was deleted due to severe obscurity.

The final Character Strengths dataset contained 117 utterances, with the number of samples per character strength varied due to the nature of the interaction. There were no samples for Leadership, Humility, Prudence, and Judgment character strengths. An example utterance for Kindness was "*But yeah it would be nice to .. for people who need help, I mean someone wants me to do something for them, even like a volunteering thing like this*".

4.2 Training Datasets

In order to train models to perform character strength recognition on the Character Strengths dataset, we collected two small datasets. The first dataset, called "Definition" in our results, is based on web scraping of definitions, examples, quotes, synonyms and exercises for each of the 24 character strengths. The Definition dataset was collected from websites that focus on character strengths (e.g., *viacharacter.org* and *positivepsychology.com*). Examples of the collected statements are; "*Stand up for someone who is being criticised, or treated unfairly*" for Bravery, and "*Expand your knowledge in an area of interest through books, journals, magazines, TV, radio or internet, for half an hour, three times a week.*" for Curiosity. The statements from the Definition dataset are differently nuanced than our Character Strengths dataset since many were acquired from coaching instructions on the Web. Nonetheless, the goal of this dataset is to evaluate the feasibility of acquiring a dataset with minimum effort that is somewhat representative to the task. We collected an average of 65 statements per character strength (σ=5.75,min=60,max=80), for a total of 1,556 statements.

The second dataset, which we refer to in our results as "Survey", was crowdsourced through email responses and Amazon Mechanical Turk. For this dataset collection, participants were asked to select their character strength and answer the following two questions, similar to how the robot asked the user in our real-world application scenario: (1) "*How have you used your chosen strength before?*", and (2) "*In what new way would you use your chosen strength tomorrow?* ". An example utterance for Zest from this dataset is: "*When I cook new recipes I have the most fun and excitement combining ingredients and usually dance while cooking.*". In total, 1,476 statements were collected, with an average of 61.5 per character strengths (σ=0.8,min=60,max=63). We also combine the data collected from both the Survey and Definition datasets to create the third "Mixed" dataset. In each dataset, we remove 20% for validation and use the rest for training.

4.3 Experiments

In order to see if our findings in Section 3 hold for a real-world HRI scenario, we train BiLSTM and BERT models on various subsets of the Survey, Definition, and Mixed datasets, and evaluate them on the Character Strengths dataset. In addition, we also provide results from Dialogflow using the same samples to demonstrate what one can achieve using a commercially available API. As described in Section 3, partial datasets were created from the Survey, Definition, and Mixed datasets separately with at most n training examples per intent, with $n \in [1,5,10,25,50]$ for the Survey and Definition datasets. For the Mixed dataset, we randomly select half of the samples from the Survey dataset and the other half from the Definition dataset, with the total $n \in [5,10,25,50,100]$. Since the Definitions dataset has at most 80 examples per intent, and the Survey dataset at most 63, neither is large enough alone to provide 100 examples per intent.

Figure 2: Character Strength dataset was collected during positive psychology sessions delivered by a social robot. College students interacted with the robot for seven positive psychology sessions in their dormitory rooms [20].

Therefore, only the Mixed dataset has results for 100 examples per intent. Dialogflow, BiLSTM, and BERT models were trained on each partial dataset. The best BiLSTM parameters as found for ATIS and Snips were used, with a hidden dimension of 300, LSTM dropout of 0.25, and batch size 16. For BERT, best models were chosen by performance on the corresponding validation set. We then evaluated the best models on the Character Strengths dataset.

4.4 Results

The results of BERT, BiLSTM, and Dialogflow models trained on our collected datasets and evaluated on the Character Strengths dataset are shown in Table 5. When compared to the BiLSTM and Dialogflow, BERT achieves higher performance across the board. While the absolute performance falls short from the previous evaluation in Section 3, Table 2, similar patterns can be noticed with the variations of the dataset size. A significant boost in test performance is observed with every increase in the number of training examples up to 25 examples per intent, after which the performance converges. Ultimately, the 25 example per intent results for the Character Strengths dataset replicates our findings in Section 3, where 25 examples per intent resulted in performance not far from the best performance.

To further inspect the reason for the lower performance, we compared the results from the BERT model with agreement results from the three human annotators. The agreement between annotators, i.e., Fleiss Kappa of 0.7, indicated that even to the human annotators, some of the character strength statements were ambiguous, with the lowest agreement in Bravery, Spirituality, Gratitude, and Perspective. For example, a Gratitude statement such as "*I was, like, really happy that day for no particular reason. Just felt, like, it was a nice walk, the day was nice, there was, like, a cool breeze.*" had an agreement score of 0.7 and was often confused with Appreciation of Beauty. The BERT model also shares confusion with the human annotators on Gratitude and Perspective (see Figure 3).

Figure 3 illustrates the differences in models performance between the 5, 10, 25 and 100 examples with Mixed dataset. In general, the performance between the highest performing model (79.5%, 100 sample per intent) and the model trained with 25 samples per intent (76.1%) in the Mixed dataset does not differ much compared to the lowest preforming models (44.4% and 65.0% for 5 and 10 samples respectively). The only noticeable difference between the two highest performing models was the Self-Regulation results, where it is likely that certain key words appearing in the 100 samples per intent training dataset were missing or less frequent in the 25 per example training dataset (See *Syntactic and Semantic Limitations* in Section 5).

Table 5: Testing Accuracy Results using Different Datasets for Character Strengths Intent Classification

Examples	Survey			Definition			Mixed		
per Intent	BiLSTM	Dialogflow	BERT	BiLSTM	Dialogflow	BERT	BiLSTM	Dialogflow	BERT
1	4.2	2.5	**11.1**	4.2	2.5	**7.7**	-	-	-
5	7.6	20.3	**42.7**	9.3	16.9	**35.0**	6.8	25.4	**44.4**
10	6.8	42.4	**76.1**	8.5	18.6	**53.0**	5.9	39.8	**65.0**
25	9.3	53.4	**76.1**	5.1	43.2	**75.2**	10.2	56.8	**76.1**
50	16.9	55.9	**76.9**	11.9	52.5	**76.1**	18.6	56.8	**76.1**
100	-	-	-	-	-	-	16.9	64.4	**79.5**

Intent accuracies of models trained with subsets of the Definition, Survey, and Mixed datasets, and evaluated on the Character Strengths dataset. The best intent scores for each partial training dataset are in bold.

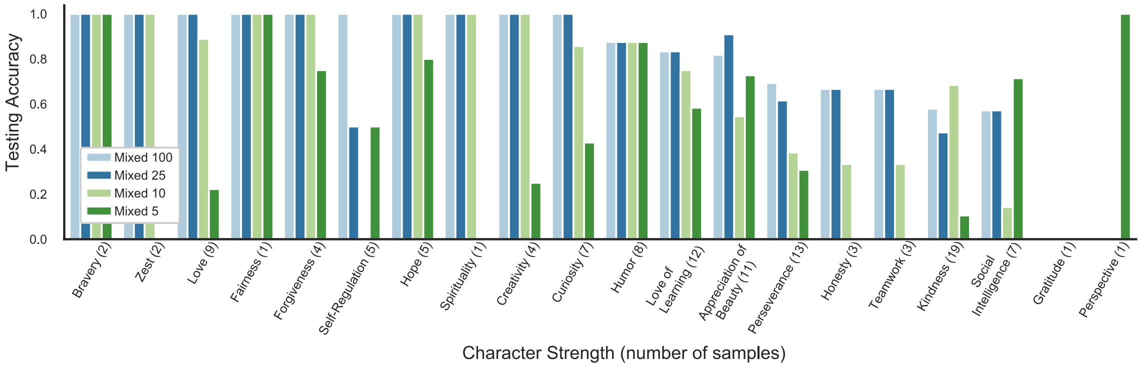


Figure 3: Performance of BERT trained on 5, 10, 25 and 100 examples per intent with the Mixed dataset. The performance difference between the 25 and 100 models is only one additional error per 25 classifications, even though the model trained with 25 examples per intent requires far less training data.

5 DISCUSSION

Performance with Minimal Training Data: Our results show that it is possible to train a very-high performing intent classification model with only about 25 examples per intent. For comparison, the entire Snips training set has around 700 examples per intent, and the ATIS training set contains 276 examples per intent on average, with a median of 77 examples per intent. Our BERT model's high performance with minimal data is true not only for the Snips dataset, which has a small number of very different intents, but also for ATIS, which has many similar intents with nuanced distinctions.

While 94% accuracy with 25 examples per intent may seem like a large drop from the entire training set performance of 98% at first, in a real-world scenario, the difference is only one additional error per 25 classifications. For someone trying to quickly develop a well-working model in a low-resource scenario, this is a very small price to pay. If a model trained on only 25 examples per intent can achieve such high performance, then anybody, whether a small organization or a sole student, can create their own deployment-ready intent classification model using a dataset they can create in a matter of hours.

A significant concern of training with such a small amount of data is the dependence of performance on the particular training examples used. It would be reasonable to believe that two different small datasets of the same size may lead to very different test performance, however our results show that this risk is minimized once there is sufficient training data, which for ATIS and Snips is around 25 examples per intent. Once this threshold is reached, the expected difference in performance of models trained on two different datasets of the same size is low. This means that our methods are still robust, even with such limited data. Of course, when gathering new training data, it is still essential that the data is of high quality, and representative of the distribution of data in testing and in the real-world application.

Evaluating on real-world HRI: Models trained for the character strengths classification task showed a similar pattern to the benchmark datasets, with 25 examples per intent resulting in a near-maximum performance. Similar to the ATIS/Snips evalution, the performance difference between the models trained with 25 and 100 examples per intent is only one additional error per 25 classifications. We also confirm that using a training dataset with a more representative distribution of the expected testing data results in better generalization (i.e., the Survey dataset that mimics the human-robot Q&A dialogues in the actual deployed study). Nonetheless, having a variety of examples from different sources of data collection, such as from Web crawling, also showed not only the ability to generalize with a small amount of data, but also the ease and performance gains of augmenting the training dataset with more samples. Comparing human annotation agreement with the performance of the BERT model reveals that the human agreement was also low in some closely nuanced character strength labels. Even with the complexity of identifying the character strengths, given their subtle definition differences and their overlapping activities in people's utterances, the BERT model with small training samples was able to generalize to our HRI application. The BiLSTM struggled to perform on the character strengths task; while it was able to effectively fit the training examples, it was unable to generalize to the character strength interactions. Across the board, our BERT models achieved much higher performance than Dialogflow, consistently by a margin of around 20%.

Syntactic and Semantic Limitations: In order to evaluate the limitations of our minimal data models, we inspected training and validation examples that were falsely classified by our BERT models trained on at most 25 examples per intent for both ATIS and Snips, when compared to the models trained on the entire datasets. Looking at the Snips misclassified examples, it is clear that the majority of false classifications are confusions between the "SearchScreeningEvent" and "SearchCreativeWork" intents. While the "SearchScreeningEvent" intent covers requests for nearby movie screenings, "SearchCreativeWork" covers searches for books, music, and other forms of media. Many utterances of both intents share similar syntax, however they differ by the use of specific keywords, such as "schedule", which occurs commonly in "SearchScreeningEvent" examples and not "SearchCreativeWork". In many of the "SearchScreeningEvent" examples misclassified as "SearchCreativeWork", these keywords are present, which would suggest that while the models may have effectively learned the syntactic patterns of the various intents, the semantics may have been less effectively modeled, especially for key words and phrases. These findings are consistent with the ATIS misclassified examples. For ATIS, confusion occurs between intents with similar syntax, but different keywords. For example, the "quantity" intent is used for questions about how many airlines use specific aircraft, while questions about how many flights leave from certain airports are contained in the "flight" intent. While these utterances have very similar syntax, specific words, especially verbs, are consistently different, however the model is unable to make this distinction. When trained on the entire dataset, this confusion is significantly reduced.

Ease of Training: One of our main goals is to demonstrate that it is easy for anyone to train a high-performing intent recognition model, without deep technical expertise and access to typical compute resources such as GPUs. Since the pre-trained BERT model that forms the core of our BERT architecture is publicly available [19], it is incredibly easy for anyone to reproduce our model and adapt it to their needs. Our models can also be trained very quickly without specialized hardware. When trained on a 2019 MacBook Pro with a 2.8 GHz Intel Core i7 CPU, our partial dataset BERT models took only 37 minutes to train on average. While the final training time depends on the number of training examples used, no model exceeded two hours of training time. This means that it is quite reasonable for these models to be trained on personal laptops.

Practical Recommendations for Intent Recognition Models in Low Data Scenarios: Based on our findings, it is feasible for any individual to train a high-performing intent recognition system using only example utterances they create themselves, or a small corpus of training examples collected in another fashion. Once enough training and validation data has been gathered, it is simple to train our models on that new data. We have a few recommendations that may be helpful for an individual attempting to train their own intent recognition model with a small amount of data:

- Even though our BERT model is certainly not small, it can be trained on a personal laptop in a matter of minutes when using several hundred (or a few thousand) training examples.
- While our models can achieve high performance with only around 25 examples per intent in our experiments, it is still essential to have enough validation data to effectively evaluate a model trained on a new corpus. A typical 80%/20% split for training/evaluation is reasonable with larger amounts of data, we recommend a 50%/50% split when using very small amounts of data.
- When comparing the performances of models trained on different partial datasets of the same size and from the same corpus, we found a positive correlation between performance and training set type-token ratio (TTR), defined as the ratio of unique tokens in the dataset to the total number of tokens. We also found positive correlation between the average TTR of examples in each intent with performance, and negative correlation between the percentage of tokens that only appear once in any intent with performance. These findings suggest that, when building a new training set, it is advantageous to gather examples with a *broad range of vocabulary*, while still having *enough similar examples* in each intent.

6 CONCLUSION

In this paper, we explored the impact of the amount of training data on intent recognition models' performance, in order to understand the minimum amount of data required for training a high-performing intent classifier. We trained logistic regression, BiLSTM, and BERT models on both the ATIS and Snips datasets, achieving results comparable to state-of-the-art. We then trained these models on smaller subsets of each dataset, and found that our BERT model can achieve 94% intent accuracy on both datasets using only 25 training examples per intent. In order to evaluate the consistency of performance for models trained on different training sets of the same size, we randomly generated 11 unique training sets from ATIS and Snips datasets for each number of examples per intent. We found that for around 25 examples per intent, our BERT models performed consistently across the random training sets, with a standard deviation in test accuracy of only 1%. Finally, we found that while these minimal data models effectively model the syntactic differences between intents, additional training data is needed to model more nuanced semantic differences.

We validated these findings on the character strength data collected from real-world HRI interactions to assess the generalizability of our approach. Our BERT model acheived 80% accuracy in this recognition task. Similarly to the results from the ATIS and Snips dataset, we found that the BERT model trained on 25 samples per intent achieved a near-maximum accuracy of 76%. Comparing our model's performance and human annotators' agreement revealed the complex nature of the character strength classification task.

This work demonstrates that high-performance intent classification models can be trained using an exceptionally small amount of data. We also made the code for our BERT model and the testing framework publicly available, so that anyone can easily replicate our results and develop models for their own HRI applications.

ACKNOWLEDGEMENT

This work was supported by the Information and Communication Technology (ICT) R&D program of the Ministry of Science and Institute for Information and Communication Technology Promotion of Republic of Korea under grant 2017-0-00162, "Development of Human-Care Robot Technology for Aging Society".

REFERENCES

[1] [n.d.]. Amazon Skills Kit. https://developer.amazon.com/en-US/alexa/alexa-skills-kit,year=2020.

[2] 2020. Amazon Lex. https://aws.amazon.com/lex/

[3] 2020. Dialogflow. https://dialogflow.cloud.google.com/

[4] Tahir Abbas, Vassilis-Javed Khan, Ujwal Gadiraju, Emilia Barakova, and Panos Markopoulos. 2020. Crowd of Oz: a crowd-powered social robotics system for stress management. *Sensors* 20, 2 (2020), 569.

[5] Thomas M Brill, Laura Munoz, and Richard J Miller. 2019. Siri, Alexa, and other digital assistants: a study of customer satisfaction with artificial intelligence applications. *Journal of Marketing Management* 35, 15-16 (2019), 1401–1436.

[6] Massimo Canonico and Luigi De Russis. 2018. A comparison and critique of natural language understanding tools. *Cloud Computing* 2018 (2018), 120.

[7] Qian Chen, Zhu Zhuo, and Wen Wang. 2019. BERT for Joint Intent Classification and Slot Filling. arXiv:1902.10909 [cs.CL]

[8] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Maël Primet, and Joseph Dureau. 2018. Snips Voice Platform: an embedded Spoken Language Understanding system for private-by-design voice interfaces. arXiv:1805.10190 [cs.CL]

[9] Deborah A. Dahl, Madeleine Bates, Michael Brown, William Fisher, Kate Hunicke-Smith, David Pallett, Christine Pao, Alexander Rudnicky, and Elizabeth Shriberg. 1994. Expanding the Scope of the ATIS Task: The ATIS-3 Corpus. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*. https://www.aclweb.org/anthology/H94-1010

[10] Caroline de Cock, Madison Milne-Ives, Michelle Helena van Velthoven, Abrar Alturkistani, Ching Lam, and Edward Meinert. 2020. Effectiveness of Conversational Agents (Virtual Assistants) in Health Care: Protocol for a Systematic Review. *JMIR Research Protocols* 9, 3 (2020), e16934.

[11] Renato De Mori, Frederic Béchet, Dilek Hakkani-Tur, Michael Mctear, Giuseppe Riccardi, and Gokhan Tur. 2008. Spoken language understanding. *Signal Processing Magazine, IEEE* 25 (06 2008), 50 – 58. https://doi.org/10.1109/MSP.2008.918413

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[13] Neta Ezer, Arthur D Fisk, and Wendy A Rogers. 2009. More than a servant: Self-reported willingness of younger and older adults to having a robot perform interactive and critical tasks in the home. In *Proceedings of the human factors and ergonomics society annual meeting*, Vol. 53. SAGE Publications Sage CA: Los Angeles, CA, 136–140.

[14] Joseph L Fleiss, Bruce Levin, Myunghee Cho Paik, et al. 1981. The measurement of interrater agreement. *Statistical methods for rates and proportions* 2, 212-236 (1981), 22–23.

[15] Leopoldina Fortunati, Anna Esposito, and Giuseppe Lugano. 2015. Introduction to the special issue "Beyond industrial robotics: Social robots entering public and domestic spheres".

[16] Noah Golmant, Nikita Vemuri, Zhewei Yao, Vladimir Feinberg, Amir Gholami, Kai Rothauge, Michael W Mahoney, and Joseph Gonzalez. 2018. On the computational inefficiency of large batch sizes for stochastic gradient descent. *arXiv preprint arXiv:1811.12941* (2018).

[17] Binnur Görer, Albert Ali Salah, and H Levent Akın. 2013. A robotic fitness coach for the elderly. In *International Joint Conference on Ambient Intelligence*. Springer, 124–139.

[18] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (Nov. 1997), 1735–1780.

[19] Hugging Face [Online]. 2020. *bert-base-uncased.* Retrieved October 5, 2020 from https://huggingface.co/bert-base-uncased

[20] S. Jeong, S. Alghowinem, L. Aymerich-Franch, K. Arias, A. Lapedriza, R. Picard, H. W. Park, and C. Breazeal. 2020. A Robotic Positive Psychology Coach to Improve College Students' Wellbeing. In *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. 187–194. https://doi.org/10.1109/RO-MAN47096.2020.9223588

[21] Sooyeon Jeong, Cynthia Breazeal, Deirdre Logan, and Peter Weinstock. 2018. Huggable: the impact of embodiment on promoting socio-emotional interactions for young pediatric inpatients. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.

[22] C. D. Kidd and C. Breazeal. 2008. Robots at home: Understanding long-term human-robot interaction. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 3230–3235.

[23] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. arXiv:1412.6980 [cs.LG]

[24] Stefan Larson, Anish Mahendran, Joseph J Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K Kummerfeld, Kevin Leach, Michael A Laurenzano, Lingjia Tang, et al. 2019. An evaluation dataset for intent classification and out-of-scope prediction. *arXiv preprint arXiv:1909.02027* (2019).

[25] Bing Liu and Ian Lane. 2016. Joint Online Spoken Language Understanding and Language Modeling With Recurrent Neural Networks. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Association for Computational Linguistics, Los Angeles, 22–30. https://doi.org/10.18653/v1/W16-3603

[26] Yijin Liu, Fandong Meng, Jinchao Zhang, Jie Zhou, Yufeng Chen, and Jinan Xu. 2019. CM-Net: A Novel Collaborative Memory Network for Spoken Language Understanding. arXiv:1909.06937 [cs.CL]

[27] Jie Lu, Vahid Behbood, Peng Hao, Hua Zuo, Shan Xue, and Guangquan Zhang. 2015. Transfer learning using computational intelligence: A survey. *Knowledge-Based Systems* 80 (2015), 14–23.

[28] Bingfeng Luo, Yansong Feng, Zheng Wang, Songfang Huang, Rui Yan, and Dongyan Zhao. 2018. Marrying Up Regular Expressions with Neural Networks: A Case Study for Spoken Language Understanding. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, 2083–2093. https://doi.org/10.18653/v1/P18-1194

[29] Michelle McQuaid, Ryan Niemiec, and Fatima Doman. 2018. A character strengths-based approach to positive psychology coaching. *Positive psychology coaching in practice* (2018), 71–79.

[30] Eda Okur, Shachi H Kumar, Saurav Sahay, Asli Arslan Esme, and Lama Nachman. 2018. Conversational intent understanding for passengers in autonomous vehicles. *arXiv preprint arXiv:1901.04899* (2018).

[31] A. K. Ostrowski, D. DiPaola, E. Partridge, H. W. Park, and C. Breazeal. 2019. Older Adults Living With Social Robots: Promoting Social Connectedness in Long-Term Communities. *IEEE Robotics Automation Magazine* 26, 2 (2019), 59–70.

[32] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, 1532–1543. https://www.aclweb.org/anthology/D14-1162

[33] Christopher Peterson, Martin EP Seligman, et al. 2004. *Character strengths and virtues: A handbook and classification*. Vol. 1. Oxford University Press.

[34] Suman Ravuri and Andreas Stolcke. 2015. Recurrent Neural Network and LSTM Models for Lexical Utterance Classification. In *Proc. Interspeech*. ISCA - International Speech Communication Association, 135–139. https://www.microsoft.com/en-us/research/publication/recurrent-neural-network-and-lstm-models-for-lexical-utterance-classification/

[35] Brian Scassellati, Laura Boccanfuso, Chien-Ming Huang, Marilena Mademtzi, Meiying Qin, Nicole Salomons, Pamela Ventola, and Frederick Shic. 2018. Improving social skills in children with ASD using a long-term, in-home social robot. *Science Robotics* 3, 21 (2018).

[36] Martin EP Seligman and Mihaly Csikszentmihalyi. 2014. Positive psychology: An introduction. In *Flow and the foundations of positive psychology*. Springer, 279–298.

[37] Takanori Shibata, Yukitaka Kawaguchi, and Kazuyoshi Wada. 2010. Investigation on people living with Paro at home. In *19th International Symposium in Robot and Human Interactive Communication*. IEEE, 470–475.

[38] Leslie N Smith. 2018. A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay. *arXiv preprint arXiv:1803.09820* (2018).

[39] Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-Read Students Learn Better: On the Importance of Pre-training Compact Models. *arXiv preprint arXiv:1908.08962v2* (2019).

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[41] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2016. Matching Networks for One Shot Learning. arXiv:1606.04080 [cs.LG]

[42] Yu Wang, Yilin Shen, and Hongxia Jin. 2018. A Bi-Model Based RNN Semantic Frame Parsing Model for Intent Detection and Slot Filling. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. Association for Computational Linguistics, New Orleans, Louisiana, 309–314. https://doi.org/10.18653/v1/N18-2050

[43] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys (CSUR)* 53, 3 (2020), 1–34.

[44] Jason D. Williams, Eslam Kamal, Mokhtar Ashour, Hani Amr, Jessica Miller, and Geoff Zweig. 2015. Fast and easy language understanding for dialog systems with Microsoft Language Understanding Intelligent Service (LUIS). In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Association for Computational Linguistics, Prague, Czech Republic, 159–161. https://doi.org/10.18653/v1/W15-4622

[45] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human

and Machine Translation. arXiv:1609.08144 [cs.CL]

[46] Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level Convolutional Networks for Text Classification. arXiv:1509.01626 [cs.LG]

[47] Zhiwei Zhao and Youzheng Wu. 2016. Attention-Based Convolutional Neural Networks for Sentence Classification. In *INTERSPEECH*.

[48] Jie Zhou and Wei Xu. 2015. End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Beijing, China, 1127–1137. https://doi.org/10.3115/v1/P15-1109

[49] Łukasz Kaiser, Ofir Nachum, Aurko Roy, and Samy Bengio. 2017. Learning to Remember Rare Events. arXiv:1703.03129 [cs.LG]