Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 750–758
Marseille, 11–16 May 2020

AMUSED: A Multi-Stream Vector Representation Method for Use in Natural Dialogue

Gaurav Kumar*[1], Rishabh Joshi*[2], Jaspreet Singh*[3], Promod Yenigalla[1]
[1]Samsung Research Institute, Bangalore, [2]Carnegie Mellon University, [3]Stony Brook University
[1]{gaurav.k1, promod.y}@samsung.com, [2]rjoshi2@andrew.cmu.edu, [3]jaspreet.ahluwalia@stonybrook.edu

Abstract

The problem of building a coherent and non-monotonous conversational agent with proper discourse and coverage is still an area of open research. Current architectures only take care of semantic and contextual information for a given query and fail to completely account for syntactic and external knowledge which are crucial for generating responses in a chit-chat system. To overcome this problem, we propose an end to end multi-stream deep learning architecture which learns unified embeddings for query-response pairs by leveraging contextual information from memory networks and syntactic information by incorporating Graph Convolution Networks (GCN) over their dependency parse. A stream of this network also utilizes transfer learning by pre-training a bidirectional transformer to extract semantic representation for each input sentence and incorporates external knowledge through the neighborhood of the entities from a Knowledge Base (KB). We benchmark these embeddings on next dialogue prediction task and significantly improve upon the existing techniques. Furthermore, we use AMUSED to represent query and responses along with its context to develop a retrieval based conversational agent which has been validated by expert linguists to have comprehensive engagement with humans.

Keywords: Natural Language Processing, Dialogue Systems, Knowledge Graphs, GCN, Memory Networks

1. Introduction

With significant advancements in Automatic speech recognition systems (Hinton et al., 2012; Kumar et al., 2018) and the field of natural language processing, conversational agents have become an important part of the current research. It finds its usage in multiple domains ranging from self-driving cars (Chen et al., 2017b) to social robots and virtual assistants (Chen et al., 2017a). Conversational agents can be broadly classified into two categories: a task-oriented chatbot and a chit-chat based system, respectively. The former works towards completion of a certain goal and are specifically designed for domain-specific needs such as restaurant reservations (Wen et al., 2017), movie recommendation (Dhingra et al., 2017), flight ticket booking systems (Wei et al., 2018) among many others. The latter is more of a personal companion and engages in human-computer interaction for entertainment or emotional companionship. An ideal chit chat system should be able to perform non-monotonous interesting conversation with context and coherence.

Current chit chat systems are either generative (Vinyals and Le, 2015) or retrieval based in nature. The generative ones tend to generate natural language sentences as responses and enjoy scalability to multiple domains without much change in the network. Even though easier to train, they suffer from error-prone responses (Zhang et al., 2018b). Retrieval based methods select the best response from a given set of answers, which makes them error-free. But, since the responses come from a specific dataset, they might suffer from distribution bias during the course of conversation.

A chit-chat system should capture semantic, syntactic, contextual, and external knowledge in a conversation to model human-like performance. Recent work by Bordes et al. (2016) proposed a memory network based approach to encode contextual information for a query while performing generation and retrieval later. Such networks can capture long term context but fail to encode relevant syntactic information through their model. Things like anaphora resolution are appropriately taken care of if we incorporate syntax. Another important component lacking in current dialogue systems is proper incorporation of external knowledge. We attempt to do this by including an explicit knowledge module that can gather information from connected entities in a Knowledge Base.

Our work improves upon previous architectures by creating enhanced representations of the conversation using multiple streams which includes Graph Convolution networks (Bruna et al., 2014), transformers (Vaswani et al., 2017) and memory networks (Bordes et al., 2016) in an end to end setting, where each component captures conversation relevant information from queries, subsequently leading to better responses. Our contribution to this paper can be summarized as follows:

- We propose AMUSED, a novel multi-stream deep learning model that learns rich unified embeddings for query response pairs using triplet loss.
- We propose an approach to incorporate external knowledge from a KB in an open domain dialog settings.
- We use Graph Convolutions Networks in a chit-chat setting to incorporate the syntactical information in the dialogue using its dependency parse.
- Even with the lack of a concrete metric to judge a conversational agent, our embeddings have shown to

* Denotes equal contribution. Work was done when authors were in Samsung Research Institute, Bangalore

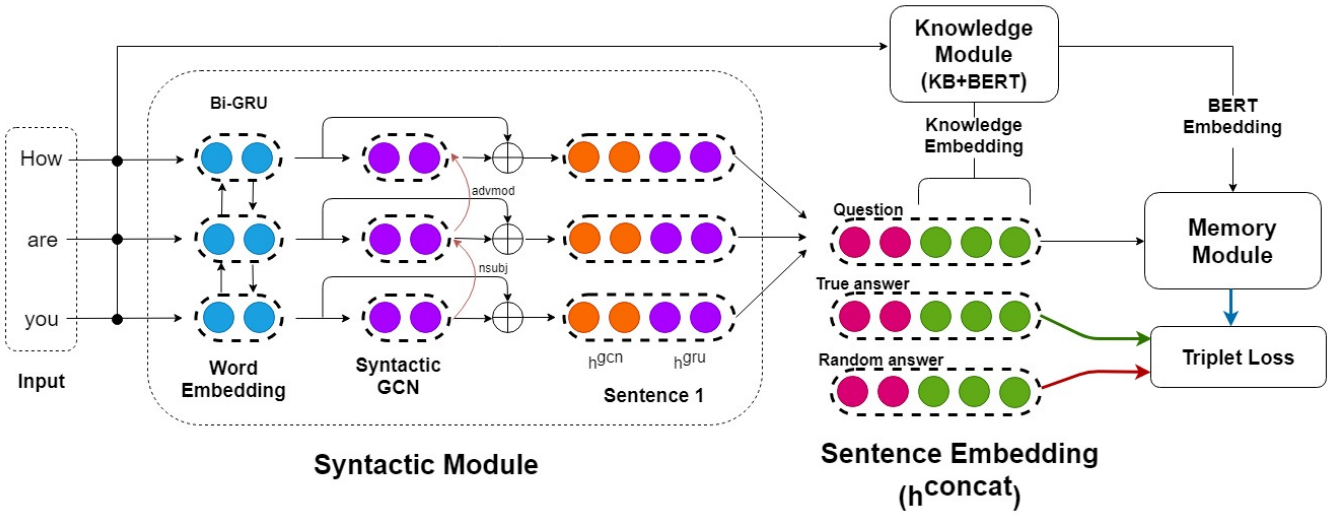

Figure 1: Overview of AMUSED. AMUSED first encodes each sentence by concatenating embeddings (denoted by $\oplus$) from Bi-LSTM and Syntactic GCN for each token, followed by word attention. The sentence embedding is then concatenated with the knowledge embedding from the Knowledge Module (Figure 2). The query embedding passes through the Memory Module (Figure 3) before being trained using triplet loss. Please see Section 4. for more details.

perform interesting response retrieval on Persona-Chat dataset.

2. Related Work

The task of building a conversational agent has gained much traction in the last decade, with various techniques being tried to generate relevant human-like responses in a chit-chat setting. Previous modular systems (Martin and Jurafsky, 2009) had a complex pipeline based structure containing various hand-crafted rules and features, making them difficult to train. This led to the need for simpler models which could be trained end to end and extended to multiple domains. Vinyals and Le (2015) proposed a simple sequence to sequence model that could generate answers based on the current question, without needing extensive feature engineering and domain specificity. However, the responses generated by this method lacked context. To alleviate this problem, Sordoni et al. (2015) introduced a dynamic-context generative network that is shown to have improved performance on the unstructured Twitter Conversation Dataset. Serban et al. (2017a) refined the sequence to sequence framework further by introducing a multi-resolution recurrent neural network that models natural language generation as two parallel processes with normal language tokens and high-level coarse tokens as their respective inputs. It helps overcome sparsity in natural language when the query, as well as context, is huge.

To model complex dependencies between sub-sequences in an utterance, Serban et al. (2017b) proposed a hierarchical latent variable encoder-decoder model. It can generate longer outputs while maintaining context at the same time. Reinforcement learning based approaches have also been deployed to generate interesting responses (Zhang et al., 2018a) and tend to possess unique conversational styles (Asghar et al., 2017). Adversarial Learning frameworks on certain specific datasets have also proved to be useful (Olabiyi et al., 2018).

With the emergence of several large datasets, retrieval methods have gained vast popularity. Even though the set of responses is limited in this scenario, it doesn't suffer from the problem of generating meaningless responses. A Sequential Matching Network proposed by Wu et al. (2017) performs word matching of responses with the context before passing their vectors to an RNN. The addition of external information along with the current input sentence and context improves the system as is evident by incorporating a large common sense knowledge base into an end to end conversational agent (Young et al., 2018). To maintain diversity in the responses, Song et al. (2018) suggests a method to combine a probabilistic model defined on itemsets with a seq2seq model. Responses like *'I am fine'* can make conversations monotonous; a specificity controlled model (Zhang et al., 2018b) in conjunction with seq2seq architecture overcomes this problem. These networks help solve one or the other problem in isolation.

To maintain proper discourse in the conversation, context vectors are passed together with input query vector into a deep learning model (Sordoni et al., 2015). A context modeling approach, which includes a concatenation of dialogue history, has also been tried (Martin and Jurafsky, 2009). However, the success of memory networks on the Question-Answering task (Sukhbaatar et al., 2015) opened the door for its further use in conversational agents. Bordes et al. (2016) used the same in a task-oriented setting for the restaurant domain and reported accuracies close to 96% in a full dialogue scenario. Zhang et al. (2018c) further used these networks in a chit chat setting on the Persona-Chat dataset and came up with personalized responses.

In our network, we make use of Graph Convolution Net-

works (Kipf and Welling, 2017; Defferrard et al., 2016), which have been found to be quite effective for encoding the syntactic information present in the dependency parse of sentences (Marcheggiani and Titov, 2017). External Knowledge Bases (KBs) have been exploited in the past to improve the performances in various tasks (Vashishth et al., 2018a; Vashishth et al., 2018b; Ling and Weld, 2012). The relation-based strategy followed by Hixon et al. (2015) creates a KB from the dialogue itself, which is later used to improve Question-Answering (Saha et al., 2018). Han et al. (2015; Ghazvininejad et al. (2018) have used KBs to generate more informative responses by using properties of entities in the graph. (Young et al., 2018) focused more on introducing knowledge from semantic-nets rather than general KBs.

3. Background: Graph Convolution Networks

GCN for undirected graph: For an undirected graph $G = (V, E)$, where V is the set of n vertices and E is the set of edges, the representation of the node v is given by $x_v \in \mathbb{R}^m, \forall v \in V$. The output hidden representation $h_v \in \mathbb{R}^d$ of the node after one layer of GCN is obtained by considering only the immediate neighbors of the node as given by Kipf and Welling (2017). To capture the multi-hop representation, GCN layers can be stacked on top of each other.

GCN for labeled directed graph: For a directed graph $G = (V, E)$, where V is the set of vertices we define the edge set E as a set of tuples $(u, v, l(u, v))$ where there is an edge having label $l(u, v)$ between nodes u and v. Marcheggiani and Titov (2017) proposed the assumption that information doesn't necessarily propagate in certain directions in the directed edge, therefore, we add tuples having inverse edges $(v, u, l(u, v)^{-1})$ as well as self loops (u, u, Ω), where Ω denotes self loops, to our edge set E to get an updated edge set E'. The representation of a node x_v, after the k^{th} layer is given as :

$$h_v^{k+1} = f\left(\sum_{u \in N(v)} (W_{l(u,v)}^k h_u^k + b_{l(u,v)}^k)\right).$$

where $W_{l(u,v)}^k \in \mathbb{R}^{d \times d}$ and $b_{l(u,v)}^k \in \mathbb{R}^d$ are trainable edge-label specific parameters for the layer k, $N(v)$ denotes the set of all vertices that are immediate neighbors of v and f is any non-linear activation function (e.g., ReLU: $f(x) = \max(0, x)$).

Since we are obtaining the dependency graph from Stanford CoreNLP (Manning et al., 2014), some edges can be erroneous. Edgewise gating (Bastings et al., 2017) helps to alleviate this problem by decreasing the effects of such edges. For this, each edge $(u, v, l(u, v))$ is assigned a score which is given by :

$$g_{uv}^k = \sigma(h_u^k \cdot \hat{w}_{l(u,v)}^k + \hat{b}_{l(u,v)}^k),$$

where $\hat{w}_{l(u,v)}^k \in \mathbb{R}^m$ and $\hat{b}_{l(u,v)}^k) \in \mathbb{R}$ are trained and σ denotes the sigmoid function. Incorporating this, the final GCN embedding for a node v after n^{th} layer is given as :

$$h_v^{n+1} = f\Big(\sum_{u \in N(v)} g_{uv}^k \times (W_{l(u,v)}^n h_u^n + b_{l(u,v)}^n)\Big). \quad (1)$$

4. AMUSED Details

This section provides details of three main components of AMUSED, which can broadly be classified into Syntactic, Knowledge, and Memory Module. We hypothesize that each module captures relevant information to learn representations for a query-response pair in a chit-chat setting.

Suppose that we have a dataset $\mathcal{D}$ consisting of a set of conversations $d_1, d_2, ..., d_C$ where d_c represents a single full length conversation consisting of multiple dialogues. A conversation d_c is given by a set of tuples $(q_1, r_1), (q_2, r_2), ..., (q_n, r_n)$ where a tuple (q_i, r_i) denotes the query and response pair for a single turn. The context for a given query $q_i \; \forall \; i \geq 2$ is defined by a list of sentences $l : [q_1, r_1, ..., q_{i-1}, r_{i-1}]$. We need to find the best response r_i from the set of all responses, $\mathcal{R}$. The training set $\mathcal{D}'$ for AMUSED is defined by set of triplets $(q_i, r_i, n_i) \; \forall \; 1 \leq i \leq N$ where N is the total number of dialogues and n_i is a negative response randomly chosen from set $\mathcal{R}$.

4.1. Syntactic Module

Syntax information from dependency trees has been successfully exploited to improve a lot of Natural Language Processing (NLP) tasks (Vashishth et al., 2018a; Mintz et al., 2009). In dialog agents, where anaphora resolution, as well as sentence structure, influences the responses, it finds special usage. A Bi-GRU (Cho et al., 2014) followed by a syntactic GCN is used in this module.

Each sentence s from the input triplet is represented with a list of k-dimensional GloVe embedding (Pennington et al., 2014) corresponding to each of the m tokens in the sentence. The sentence representation $S \in \mathbb{R}^{m \times k}$ is then passed to a Bi-GRU to obtain the representation $S^{gru} \in \mathbb{R}^{m \times d_{gru}}$, where d_{gru} is the dimension of the hidden state of Bi-GRU.

This contextual encoding (Graves et al., 2013) captures the local context well but fails to capture the long-range dependencies that can be obtained from the dependency trees. We use GCN to encode this syntactic information. Stanford CoreNLP (Manning et al., 2014) is used to obtain the dependency parse for the sentence s. Giving the input as S^{gru}, we use GCN Equation 1, to obtain the syntactic embedding S^{gcn}. Following Nguyen and Grishman (2018), we only use three edge labels, namely, forward-edge, backward-edge, and self-loop. This is done because incorporating all the edge labels from the dependency graph heavily over-parameterizes the model.

The final token representation is obtained by concatenating the contextual Bi-GRU representation h^{gru} and the syntactic GCN representation h^{gcn}. A sentence representation is then obtained by passing the tokens through a layer of word attention (Bahdanau et al., 2014) as used by (Vashishth et

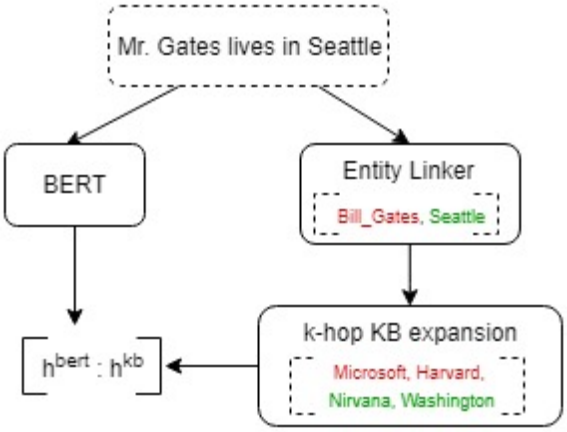

Figure 2: Description of Knowledge Module. The input sentence is passed to a pre-trained BERT model, output from which is concatenated with averaged embedding from the KB-neighbors of entities present in the input. Refer Section 4.2. for a detailed explanation.

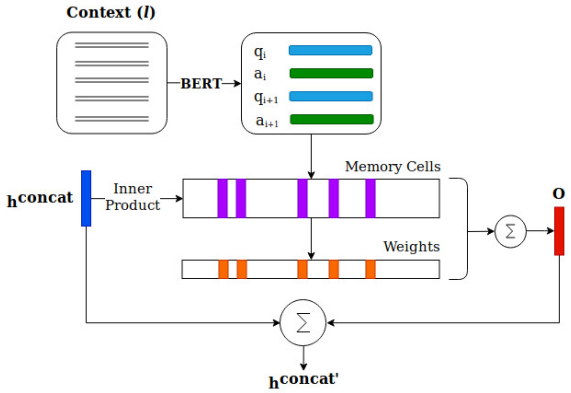

Figure 3: Memory Module description. The query representation and BERT embeddings of the context sentences is passed to the memory network to capture the dialogue context. Please see Section 4.3. for more details.

al., 2018b; Jat et al., 2018), which is concatenated with the embedding obtained from the Knowledge Module (described in Section 4.2.) to obtain the final sentence representation h^{concat}.

4.2. Knowledge Module

The final sentence representation h^{concat} of the query is then passed into the Knowledge Module. It is further subdivided into two components: a pre-trained Transformer model for the next dialogue prediction problem and a component to incorporate information from external Knowledge Bases (KBs).

4.2.1. Next Dialogue Prediction Using Transformers

The next dialogue prediction task is described as follows: For each query-response pair in the dataset, we generate a positive sample (q, r) and a negative sample (q, n) where n is randomly chosen from the set of responses $\mathcal{R}$ in dataset D. Following Devlin et al. (2018), a training example is defined by concatenating q and r which are separated by a delimiter $||$ and is given by $[q||r]$. The problem is to classify if the next dialogue is a correct response or not.

A pre-trained BERT model is used to further train a binary classifier for the next dialogue prediction task, as described above. After the model is trained, the pre-final layer is considered, and the vector from the special cls token is chosen as the dialogue representation. The representation thus obtained would tend to be more inclined towards its correct positive responses. Multi-head attention in the transformer network, along with positional embeddings during training, helps it to learn intra as well as inter sentence dependencies (Devlin et al., 2018; Vaswani et al., 2017). The input query sentence is then passed from this network to obtain the BERT embedding, h^{bert}. We are motivated by the fact that pretrained BERT would help incorporate more generic knowledge (Petroni et al., 2019).

4.2.2. k-Hop KB Expansion

In our day-to-day conversations, to ask succinct questions, or to keep the conversation flowing, we make use of some background knowledge . For example, if someone remarks that they like rock music, we can ask a question if they have listened to Nirvana. It can be done only if we know that Nirvana plays rock music. To incorporate such external information, we can make use of existing Knowledge Bases like Wikipedia, Freebase (Bollacker et al., 2008), and Wikidata (Vrandečić and Krötzsch, 2014). Entities in these KBs are linked to each other using relations. We can expand the information we have about an entity in our dialogue by looking at its linked entities in a Knowledge Graph. Multiple hops of the (KG) can be used to expand knowledge.

In AMUSED, we do this by passing the input query into Stanford CoreNLP to obtain entity linking information to Wikipedia. Suppose the Wikipedia page of an entity e contains links to the set of entities E. We ignore relation information and only consider one-hop direct neighbors of e. To obtain a KB-expanded embedding h^{kb} of the input sentence, we take the average of GloVE embeddings of each entity in E. In place of Wikipedia, bigger knowledge bases like Wikidata, as well as the relation information, can be used to improve KB embeddings. We leave that for future work.

4.3. Memory Module

For effective conversations, it is imperative that we form a sense of the dialogues that have already happened. A question about '*Who is the president of USA*' followed by '*What about France*' should be self-containing. This dialogue context is encoded using a memory network (Sukhbaatar et al., 2015). The memory network helps to capture the context of the conversation by storing dialogue history i.e., both questions and responses. The query representation, h^{concat} is passed to the memory network,

along with BERT embeddings h^{bert} of the context, from the Knowledge Module (Section 4.2.).

In AMUSED, memory network uses supporting memories to generate the final query representation ($h^{concat'}$). Supporting memories contains input (m_i) and output (c_i) memory cells (Sukhbaatar et al., 2015). The incoming query q_i as well as the history of dialogue context l : $[(q_1, r_1), .., (q_{i-1}, r_{i-1})]$ is fed as input. The memory cells are populated using the BERT representations of context sentences l as follows: $m_i = c_i = \{BERT(x), x \in \tau\}$, where $\tau = [q_1, r_1, q_2, r_2, ..., q_i, r_i] \forall (q_i, r_i) \in l$.

Following Bordes et al. (2016), the incoming query embedding along with input memories is used to compute relevance of context stories as a normalized vector of attention weights as $a_i = \|(< m_i, h^{concat} >)\|$, where $< a, b >$ represents the inner product of a and b. The response from the output memory, o, is then generated as : $o = \sum_i a_i c_i$. The final output of the memory cell, u is obtained by adding o to h^{concat}. To capture context in an iterative manner, memory cells are stacked in layers (Sukhbaatar et al., 2015) which are called as hops. The output of the memory cell after the k^{th} hop is given by $u^k = o^{k-1} + u^{k-1}$ where $u^0 = h^{concat}$. The memory network performs k such hops and the final representation $h^{concat'}$ is given by sum of o^k and u^k.

4.4. Triplet Loss

Triplet loss has been successfully used for face recognition (Schroff et al., 2015). Our insight is that traditional loss metrics might not be best suited for a retrieval-based task with a multitude of valid responses to choose from. We define a *Conversational Euclidean Space*, where the representation of a sentence is driven by its context in the dialogue along with its syntactic and semantic information. We have used this loss to bring the query and response representations closer in the conversational space. Questions with similar answers should be closer to each other and the correct response. An individual data point is a triplet which consists of a query (q_i), its correct response (r_i) and a negative response (n_i) selected randomly. We need to learn their embeddings $\phi(q_i) = h_{q_i}^{concat'}$, $\phi(r_i) = h_{r_i}^{concat}$ and $\phi(n_i) = h_{n_i}^{concat}$ such that the positive pairs are closer in the embedding space compared to the negative ones. This leads to the following equation:

$$\|\phi(q_i) - \phi(r_i)\|_2^2 + \alpha < \|\phi(q_i) - \phi(n_i)\|_2^2 ,$$

where α is the margin hyper-parameter used to separate negative and positive pairs. If I be the set of triplets, N the number of triplets and w the parameter set, then, triplet loss ($\mathcal{L}$) is defined as :

$$\mathcal{L}(I, w) = \sum_{i=0}^{N} [\|\phi(q_i) - \phi(r_i)\|_2^2 - \|\phi(q_i) - \phi(n_i)\|_2^2 + \alpha]_+$$

	Size
Training conversations	9,907
Validation conversations	1000
Test conversations	1000
Query- Response pairs	131438
Vocabulary size	19,262

Table 1: Dataset statistics for Persona-Chat. Refer Section 5.1.

5. Experimental Setup

5.1. Datasets

Persona-Chat: We use this dataset to build and evaluate the chit-chat system. Persona-Chat (Zhang et al., 2018c) is an open domain dataset on personal conversations created by randomly pairing two humans on Amazon Mechanical Turk. The paired crowd workers converse naturally for $6 - 12$ turns. This made sure that the data mimic normal conversations between humans, which is very crucial for building such a system. This data is not limited to social media comments or movie dialogues. We use it for training AMUSED as it provides consistent conversations with proper context.

DSTC: Dialogue State Tracking Challenge dataset (Henderson et al., 2014) contains conversations for restaurant booking tasks. Due to its task-oriented nature, it doesn't need an external knowledge module, so we train it only using memory and syntactic module and test on an automated metric.

MNLI and **MRPC**: We further use Multi-Genre Natural Language Inference and Microsoft Research Paraphrase Corpus (Wang et al., 2019) to fine-tune parts of the network i.e.; Knowledge Module. It is done because these datasets resemble the core nature of our problem, wherein we want to predict the correctness of one sentence in response to a particular query.

5.2. Training

Pre-training BERT: Before training AMUSED, the knowledge module is processed by pre-training a bidirectional transformer network and extracting one-hop neighborhood entities from Wikipedia KB. We use the approach for training, as explained in Section 4.2.1.. There are 104,224 positive training and 27,214 validation query-response pairs from Persona Chat. We perform three different operations: a) Equal sampling: Sample equal number of negative examples from dataset, b) Oversampling: Sample double the negatives to make training set biased towards negatives and c) Under sampling: Sample 70% of negatives to make training set biased towards positives. Batch size and maximum sequence length are 32 and 128, respectively. We fine-tune this next sentence prediction model with MRPC and MNLI datasets, which improves the performance. Evaluation is done on above mentioned three methods, and we choose the best model

Model	Concat	Diff	Min
Bi-GRU & GCN only	72.8%	73.2%	59.4%
BERT only	74.3%	71.9%	66.3%
BERT with Bi-GRU & GCN	77.7%	73.3%	71.4%
BERT with Bi-GRU and memory networks	78.6%	74.4%	73.5%
BERT and KB with Bi-GRU and memory networks	**83.6%**	78.4%	69.2%

Table 2: Accuracy as an automatic evaluation metric on Next Dialogue Prediction task over Persona Chat. We perform different operations on embeddings of sentence pairs to study ablation. Concat, Diff and Min refers to Concatenation, Difference and Element wise min respectively. See Section 5.4.2. for more details.

from them.

Training to learn Embeddings: AMUSED requires triplets to be trained using triplet loss. A total of $131,438$ triplets of the form (q, r, n) are randomly split in a 90:10 ratio to form the training and validation set. The network is trained with a batch size of 64 and a dropout of 0.5. Word embedding size is chosen to be 50. Bi-GRU and GCN hidden state dimensions are chosen to be 192 and 32, respectively. One layer of GCN is employed. Validation loss is used as a metric to stop training, which converges after 50 epochs using Adam optimizer at 0.001 learning rate.

5.3. Retrieval

As a retrieval-based model, the system selects a response from the predefined answer set. The retrieval unit extracts embedding (h^{concat}) for each answer sentence from the trained model and stores it in a representation matrix, which will be utilized later during inference.
First, a candidate subset A is created by sub-sampling a set of responses having overlapping words with a given user query. Then, the final output is retrieved on the basis of cosine similarity between query embedding $h^{concat'}$ and the extracted set of potential responses (A). The response with the highest score is then labeled as the final answer, and the response embedding is further added into the memory to take care of context.

5.4. Results And Evaluation

5.4.1. Selecting the Pre-Trained Model

The model resulting from the oversampling method beats its counterparts by **more than 3**$\%$ in accuracy. It clearly indicates that a better model is one which learns to distinguish negative examples well. The sentence embeddings obtained through this model is further used for lookup in the Knowledge Module (Section 4.2.) in AMUSED.

5.4.2. Ablation Studies on Automated Metrics

We use two different automated metrics to check the effectiveness of the model and the query-response representations that we learned.

Method	Precision@1
Seq2Seq	0.092
Profile Memory	0.092
IR Baseline	0.214
AMUSED (Persona Chat)	**0.326**
AMUSED (DSTC)	**0.78**

Table 3: Precision @1 comparison between different methods on Persona Chat. Precision@1 % tell us the number of times the correct response from the dataset comes up. Details in Section 5.4.2.

Next Dialogue Prediction Task: Various components of AMUSED are analyzed for their performance on the next dialogue prediction task. This task tells us that, given two sentences (a query and a response) and the context, whether the second sentence is a valid response to the first sentence or not. Embeddings for queries and responses are extracted from our trained network and then multiple operations, which include a) Concatenation, b) Element wise min and c) Subtraction are performed on those before passing them to a binary classifier. A training example consists of embeddings of two sentences from a (q, a) or (q, n) pair which are created in a similar fashion as in Section 4.2.1..

Accuracy on this binary classification problem has been used to select the best network. Furthermore, we perform ablation studies using different modules to understand the effect of each component in the network. A 4 layer neural network with ReLU activation in its hidden layers and softmax in the final layer is used as the classifier. External knowledge in conjunction with memory and GCN module has the best accuracy when embeddings of query and response are concatenated together. A detailed study of the performance of various components over these operations is shown in Table 2.

Precision@1: This is another metric used to judge the effectiveness of our network. It is different from the next sentence prediction task accuracy. It measures that for n trials, the number of times a relevant response is reported with the highest confidence value. Table 3 reports a comparative study of this metric on 500 trials conducted for AMUSED along with results for other methods. DSTC dataset is also evaluated on this metric without the knowledge module as explained in Section 5.1.

Looking for exact answers might not be a great metric as many diverse answers might be valid for a particular question. So, we must look for answers which are contextually relevant for that query. Overall, we use next sentence prediction task accuracy to choose the final model before retrieval.

Model	Coherence	Context Aware	Non Monotonicity	Average Rating	%gain
Bi-GRU & GCN only	6.82	7.35	6.77	6.98	Baseline
BERT only	7.61	7.24	6.33	7.06	1.14
BERT with Bi-GRU & GCN	7.54	6.91	7.38	7.27	4.15
BERT and External KB with Bi-GRU & GCN	7.16	7.34	7.72	7.40	6.01
KV Memory Networks(Zhang et al., 2018c)	7.56	8.09	7.84	7.83	12.18
BERT & External KB with Bi-GRU, GCN and memory networks	8.21	8.34	7.82	8.12	16.33

Table 4: Human based evaluation is conducted for 5 different components in the network as well as KV memory networks. AMUSED achieves the highest percent gain over specified baseline model. The scale is 1-10.

5.4.3. Ablation Study by Humans

There is no concrete metric to evaluate the performance of an entire conversation in a chit-chat system. Hence, the human evaluation was conducted using expert linguists to check the quality of conversation. They were asked to chat for 7 turns and rate the quality of responses on a scale of $1-10$ where 1 being the worst and 10 being the best. Each expert was asked to conduct the experiment 50 times with a time gap after 10 sessions. This gap was provided to make sure that the linguistic experts don't get biased in their results. A group of 20 linguists provided a total of 945 sample points for each model. Similar to Zhang et al. (2018c), there were multiple parameters to rate the chat based on coherence, context awareness, and non-monotonicity to measure various factors that are essential for natural dialogue. By virtue of our network being retrieval based, we don't need to judge the responses based on their structural correctness as this will be implicit.

To monitor the effect of each neural component, we get it rated by experts either in isolation or in conjunction with other components. Such a study helps us understand the impact of different modules on a human-based conversation. Dialogue system proposed by Zhang et al. (2018c) is also reproduced and reevaluated for comparison. From Table 4, we can see that human evaluation follows a similar trend as the automated metric, with the best rating given to the combined architecture.

6. Conclusion

In the paper, we propose AMUSED, a multi-stream architecture that effectively encodes semantic information from the query while properly utilizing external knowledge for improving performance on natural dialogue. It also employs GCN to capture long-range syntactic information and improves context-awareness in dialogue by incorporating a memory network. Through our experiments and results using different metrics, we demonstrate that learning these rich representations through smart training (using triplets) would improve the performance of chit-chat systems. The ablation studies show the importance of different components for better dialogue. Our ideas can easily be extended to various conversational tasks that would benefit from such enhanced representations.

7. Acknowledgements

We would like to thank the anonymous reviewers for providing constructive and valuable feedback for this paper. The authors would also like to express their gratitude to Samsung Research Institute, Bangalore, India for supporting this research work completely.

8. Bibliographical References

Asghar, N., Poupart, P., Jiang, X., and Li, H. (2017). Deep active learning for dialogue generation. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, pages 78–83, Vancouver, Canada, August. Association for Computational Linguistics.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv e-prints*, abs/1409.0473, September.

Bastings, J., Titov, I., Aziz, W., Marcheggiani, D., and Simaan, K. (2017). Graph convolutional encoders for syntax-aware neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1957–1967, Copenhagen, Denmark, September. Association for Computational Linguistics.

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

Bordes, A., Boureau, Y.-L., and Weston, J. (2016). Learning end-to-end goal-oriented dialog. *arXiv preprint arXiv:1605.07683*.

Bruna, J., Zaremba, W., Szlam, A., and Lecun, Y. (2014). Spectral networks and locally connected networks on graphs. In *International Conference on Learning Representations (ICLR2014), CBLS, April 2014*.

Chen, H., Liu, X., Yin, D., and Tang, J. (2017a). A survey on dialogue systems: Recent advances and new frontiers. *SIGKDD Explor. Newsl.*, 19(2):25–35, November.

Chen, S., Zhang, S., Shang, J., Chen, B., and Zheng, N. (2017b). Brain inspired cognitive model with attention for self-driving cars. *CoRR*, abs/1702.05596.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734. Association for Computational Linguistics.

Defferrard, M., Bresson, X., and Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 3844–3852, USA. Curran Associates Inc.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2018). BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Dhingra, B., Li, L., Li, X., Gao, J., Chen, Y.-N., Ahmed, F., and Deng, L. (2017). Towards end-to-end reinforcement learning of dialogue agents for information access. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–495, Vancouver, Canada, July. Association for Computational Linguistics.

Ghazvininejad, M., Brockett, C., Chang, M.-W., Dolan, B., Gao, J., Yih, S. W.-t., and Galley, M. (2018). A knowledge-grounded neural conversation model. February.

Graves, A., r. Mohamed, A., and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649, May.

Han, S., Bang, J., Ryu, S., and Lee, G. G. (2015). Exploiting knowledge base to generate responses for natural language dialog listening agents. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 129–133, Prague, Czech Republic, September. Association for Computational Linguistics.

Henderson, M., Thomson, B., and Williams, J. D. (2014). The second dialog state tracking challenge. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 263–272.

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Kingsbury, B., et al. (2012). Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal processing magazine*, 29.

Hixon, B., Clark, P., and Hajishirzi, H. (2015). Learning knowledge graphs for question answering through conversational dialog. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 851–861, Denver, Colorado, May–June. Association for Computational Linguistics.

Jat, S., Khandelwal, S., and Talukdar, P. (2018). Improving Distantly Supervised Relation Extraction using Word and Entity Based Attention. *ArXiv e-prints*, April.

Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Kumar, A., Verma, S., and Mangla, H. (2018). A survey of deep learning techniques in speech recognition. In *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pages 179–185, Oct.

Ling, X. and Weld, D. S. (2012). Fine-grained entity recognition. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, pages 94–100. AAAI Press.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Marcheggiani, D. and Titov, I. (2017). Encoding sentences with graph convolutional networks for semantic role labeling. volume abs/1703.04826.

Martin, J. H. and Jurafsky, D. (2009). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson/Prentice Hall Upper Saddle River.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Nguyen, T. and Grishman, R. (2018). Graph convolutional networks with argument-aware pooling for event detection.

Olabiyi, O., Salimov, A., Khazane, A., and Mueller, E. (2018). Multi-turn dialogue response generation in an adversarial learning framework. *arXiv preprint arXiv:1805.11752*.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., and Miller, A. (2019). Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China, November. Association for Computational Linguistics.

Saha, A., Pahuja, V., Khapra, M. M., Sankaranarayanan, K., and Chandar, S. (2018). Complex sequential question answering: Towards learning to converse over linked question answer pairs with a knowledge graph. In *AAAI*, pages 705–713. AAAI Press.

Schroff, F., Kalenichenko, D., and Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823. IEEE Computer Society.

Serban, I. V., Klinger, T., Tesauro, G., Talamadupula, K.,

Zhou, B., Bengio, Y., and Courville, A. (2017a). Multiresolution recurrent neural networks: An application to dialogue response generation. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Serban, I. V., Sordoni, A., Lowe, R., Charlin, L., Pineau, J., Courville, A., and Bengio, Y. (2017b). A hierarchical latent variable encoder-decoder model for generating dialogues. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pages 3295–3301. AAAI Press.

Song, Y., Yan, R., Feng, Y., Zhang, Y., Zhao, D., and Zhang, M. (2018). Towards a neural conversation model with diversity net using determinantal point processes. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015). A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 196–205, Denver, Colorado, May–June. Association for Computational Linguistics.

Sukhbaatar, S., Weston, J., Fergus, R., et al. (2015). End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448.

Vashishth, S., Jain, P., and Talukdar, P. (2018a). CESI: Canonicalizing open knowledge bases using embeddings and side information. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pages 1317–1327, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Vashishth, S., Joshi, R., Prayaga, S. S., Bhattacharyya, C., and Talukdar, P. (2018b). RESIDE: Improving distantly-supervised neural relation extraction using side information. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1257–1266. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *CoRR*, abs/1706.03762.

Vinyals, O. and Le, Q. (2015). A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Vrandečić, D. and Krötzsch, M. (2014). Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, September.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. (2019). GLUE: A multi-task benchmark and analysis platform for natural language understanding. In the Proceedings of ICLR.

Wei, W., Le, Q., Dai, A., and Li, J. (2018). AirDialogue: An environment for goal-oriented dialogue research. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3844–3854, Brussels, Belgium, October-November. Association for Computational Linguistics.

Wen, T.-H., Vandyke, D., Mrkšić, N., Gasic, M., Rojas Barahona, L. M., Su, P.-H., Ultes, S., and Young, S. (2017). A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain, April. Association for Computational Linguistics.

Wu, Y., Wu, W., Xing, C., Zhou, M., and Li, Z. (2017). Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 496–505, Vancouver, Canada, July. Association for Computational Linguistics.

Young, T., Cambria, E., Chaturvedi, I., Zhou, H., Biswas, S., and Huang, M. (2018). Augmenting end-to-end dialogue systems with commonsense knowledge. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Zhang, H., Lan, Y., Guo, J., Xu, J., and Cheng, X. (2018a). Reinforcing coherence for sequence to sequence model in dialogue generation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4567–4573. International Joint Conferences on Artificial Intelligence Organization, 7.

Zhang, R., Guo, J., Fan, Y., Lan, Y., Xu, J., and Cheng, X. (2018b). Learning to control the specificity in neural response generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1108–1117.

Zhang, S., Dinan, E., Urbanek, J., Szlam, A., Kiela, D., and Weston, J. (2018c). Personalizing dialogue agents: I have a dog, do you have pets too? *CoRR*, abs/1801.07243.