Can Taxonomy Help? Improving Semantic Question Matching using Question Taxonomy

Deepak Gupta[*], **Rajkumar Pujari**[†], **Asif Ekbal**[*], **Pushpak Bhattacharyya**[*],
Anutosh Maitra[‡] **Tom Jain**[‡] and **Shubhashis Sengupta**[‡]
[*]Indian Institute of Technology Patna, India
[†]Purdue University, USA
[‡]Accenture Labs, Bengaluru, India
[*]{deepak.pcs16,asif,pb}@iitp.ac.in, [†]rpujari@purdue.edu
[‡]{anutosh.maitra,tom.geo.jain,shubhashis.sengupta}@accenture.com

Abstract

In this paper, we propose a hybrid technique for semantic question matching. It uses a proposed two-layered taxonomy for English questions by augmenting state-of-the-art deep learning models with question classes obtained from a deep learning based question classifier. Experiments performed on three open-domain datasets demonstrate the effectiveness of our proposed approach. We achieve state-of-the-art results on partial ordering question ranking (POQR) benchmark dataset. Our empirical analysis shows that coupling standard distributional features (provided by the question encoder) with knowledge from taxonomy is more effective than either deep learning (DL) or taxonomy-based knowledge alone.

1 Introduction

Question Answering (QA) is a well investigated research area in Natural Language Processing (NLP). There are several existing QA systems that answer factual questions with short answers (Iyyer et al., 2014; Bian et al., 2008; Ng and Kan, 2015). However, systems which attempt to answer questions that have long answers with several well-formed sentences, are rare in practice. This is mainly due to some of the following challenges: (i) selecting appropriate text fragments from document(s), (ii) generating answer texts with coherent and cohesive sentences, (iii) ensuring the syntactic as well as semantic well-formedness of the answer text. However, when we already have a set of answered questions, reconstructing the answers for semantically similar questions can be bypassed. For each unseen question, the most semantically similar question is identified by comparing the unseen question with the existing set of questions. The question, which is closest to the unseen question can be retrieved as a possible semantically similar question. Thus, accurate semantic question matching can significantly improve a QA system. In the recent past, several deep learning based models such as recurrent neural networks (RNNs), convolution neural network (CNN), gated recurrent units (GRUs) etc. have been explored to obtain representation at the word (Mikolov et al., 2013; Pennington et al., 2014), sentence (Kim, 2014) and paragraph (Zhang et al., 2017) level.

In the proposed semantic question matching framework, we use attention based neural network models to generate question vectors. We create a hierarchical taxonomy by considering different types and subtypes in such a way that questions having similar answers belong to the same taxonomy class. We propose and train a deep learning based question classifier network to classify the taxonomy classes. The taxonomy information is helpful in taking a decision on semantic similarity between them. For example, the questions '*How do scientists work?*' and '*Where do scientists work?*', have very high lexical similarity but they have different answer types. This can be easily identified using a question taxonomy. Taxonomy can provide very useful information when we do not have enough data for generating useful deep learning based representations, which are generally the case with restricted domains. In such scenarios linguistic information obtained from the prior knowledge helps significantly in improving the performance of the system.

Proceedings of the 27th International Conference on Computational Linguistics, pages 499–513
Santa Fe, New Mexico, USA, August 20-26, 2018.

We propose a neural network based algorithm to classify the questions into appropriate taxonomy class(es). The information, thus obtained from taxonomy, is used along with the DL techniques to perform semantic question matching. Empirical evidence establishes that our taxonomy, when used in conjunction with Deep Learning (DL) representations, improves the performance of the system on semantic question (SQ) matching task.

We summarize the contributions of our work as follows: **(i)** We create a two-layered taxonomy for English questions; **(ii)** We propose a deep learning based method to identify taxonomy classes of questions; **(iii)** We propose a dependency parser based technique to identify the *focus* of the question; **(iv)** We propose a framework to integrate semantically rich taxonomy classes with DL based encoder to improve the performance and achieve new state-of-the-art results in semantic question ranking on benchmark dataset and Quora dataset; and finally **(v)** We release two annotated datasets, one for semantically similar questions and the other for question classification.

2 Related Works

Rapid growth of community question and answer (cQA) forums have intensified the necessity for semantic question matching in QA setup. Answer retrieval of semantically similar questions has drawn the attention of researchers in very recent times (Màrquez et al., 2015; Nakov et al., 2016). It solves the problem of *question starvation* in cQA forums by providing a semantically similar question which has already been answered. In literature, there have been attempts to address the problem of finding the most similar match to a given question, for e.g. Burke et al. (1997) and Mlynarczyk and Lytinen (2005). Wang et al. (2009) have presented syntactic tree based matching for finding semantically similar questions. 'Similar question retrieval' has been modeled using various techniques such as topic modeling (Li and Manandhar, 2011), knowledge graph representation (Zhou et al., 2013) and machine translation (Jeon et al., 2005). Semantic kernel based similarity methods for QA have also been proposed in (Filice et al., 2016; Croce et al., 2017; Croce et al., 2011).

Answer selection in QA forums is similar to the question similarity task. In recent times, researchers have been investigating DL-based models for answer selection (Wang and Nyberg, 2015; Severyn and Moschitti, 2015; Feng et al., 2015). Most of the existing works either focus on better representations for questions or linguistic information associated with the questions. On the other hand, the model proposed in this paper is a hybrid model. We also present a thorough empirical study of how sophisticated DL models can be used along with a question taxonomy concepts for semantic question matching.

3 Question Matching Framework

When framed as a computational problem, semantic question (SQ) matching for QA becomes equivalent to ranking questions in the existing question-base according to their semantic similarity to the given input question. Existing state-of-the-art systems use either deep learning models (Lei et al., 2016) or traditional text similarity methods (Jeon et al., 2005; Wang et al., 2009) to obtain the similarity scores. In contrast, our framework of SQ matching efficiently combines deep learning based question encoding and a linguistically motivated taxonomy. Algorithm 1 describes the precise method we follow. $Similarity(.)$ is the standard cosine similarity function. $fsim$ is focus embedding similarity which is described later in Section 4.4.

3.1 Question Encoder Model

Our question encoder model is inspired from the state-of-the-art question encoder architecture proposed by Lei et al. (2016). We extend the question encoder model of Lei et al. (2016) by introducing attention mechanism similar to Bahdanau et al. (2014) and Chopra et al. (2016). We propose the attention based version of two question encoder models, namely Recurrent Convolutional Neural Network (RCNN) (Lei et al., 2016) and Gated Recurrent Unit (GRU) (Chung et al., 2014; Cho et al., 2014).

A question encoder with attention does not need to capture the whole semantics of the question in its final representation. Instead, it is sufficient to capture a part of hidden state vectors of another question it needs to attend while generating the final representation. Let $\mathbf{H} \in \mathbb{R}^{d \times n}$ be a matrix consisting of

Algorithm 1 Semantic Question Matching

procedure SQ MATCHING(QSet)
  RESULTS ← {}
  for (p, q) in QSet **do**
    $\vec{p}, \vec{q} \leftarrow$ *Question-Encoder*(p, q)
    $sim \leftarrow$ `Similarity`($\vec{p}$, $\vec{q}$)
    $T_p^c, T_q^c \leftarrow$ `Taxonomy-Classes`(p, q)
    $F_p, F_q \leftarrow$ `Focus`(p, q)
    $\vec{F_p}, \vec{F_q} \leftarrow$ `Focus-Encoder`(F_p, F_q)
    $fsim \leftarrow$ `Similarity`($\vec{F_p}$, $\vec{F_q}$)
    Feature-Vector=[sim, ${T_p}^c$, ${T_q}^c$, $fsim$]
    result ← Classifier(Feature-Vector)
    RESULTS.append(result)
  return RESULTS

hidden state vectors $[h_1, h_2 \dots h_n]$ that the question encoder (RCNN, GRU) produced when reading the n words of the question, where d is a hyper parameter denoting the size of embeddings and hidden layers. The attention mechanism will produce an attention weight vector $\alpha_t \in \mathbb{R}^n$ and a weighted hidden representation $r_t \in \mathbb{R}^d$.

$$\begin{aligned} C_t &= tanh(W^H H + W^v(v_t \otimes I_n)) \\ \alpha_t &= softmax(w^T C_t) \\ r_t &= H\alpha^T \end{aligned} \tag{1}$$

where W^H, $W^v \in \mathbb{R}^{d \times d}$, are trained projection matrices. w^T is the transpose of the trained vector $w \in \mathbb{R}^d$. $v_t \in \mathbb{R}^d$ shows the embedding of token x_t and $I_n \in \mathbb{R}^n$ is the vector of 1. The product $W^v(v_t \otimes I_n)$ is repeating the linearly transformed v_t as many times (n) as there are words in the candidate question. Similarly we can obtain the attentive hidden state vectors $[r_1, r_2 \dots r_n]$. We apply the averaging pooling strategy to determine the final representation of the question.

Annotated data, $\mathscr{D} = \{(q_i, p_i^+, p_i^-)\}$ is used to optimize $f(p, q, \phi)$, where $f(.)$ is a measure of similarity between the questions p and q, and ϕ is a parameter to be optimized. Here p_i^+ and p_i^- correspond to the similar and non-similar question sets, respectively for question q_i. Maximum margin approach is used to optimize the parameter ϕ. For a particular training example, where q_i is similar to p_i^+, we minimize the max-margin loss $\mathcal{L}(\phi)$ defined as:

$$\mathcal{L}(\phi) = \max_{p \in Q'(q_i)} \left\{ f(q_i, p; \phi) - f(q_i, p_i^+; \phi) + \lambda(p, p_i^+) \right\} \tag{2}$$

where $Q'(q_i) = p_i^+ \cup p_i^-$, $\lambda(p, p_i^+)$ is a positive constant set to 1 when $p \neq p_i^+$, 0 otherwise.

3.2 Question Taxonomy

Questions are ubiquitous in natural language. Questions essentially differ on two fronts: semantic and syntactic. Questions that differ syntactically might still be semantically equivalent. Let us consider the following two questions:

- What is the number of new hires in 2018?
- How many employees were recruited in 2018?

Although the above questions are not syntactically similar but both are semantically equivalent and have the same answer. A well-formed taxonomy and question classification scheme can provide this information which eventually helps in determining the semantic similarity between the questions.

According to Gruber (1995), ontologies are commonly defined as specifications of shared conceptualizations. Informally, conceptualization is the relevant informal knowledge one can extract from their experience, observation or introspection. Specification corresponds to the encoding of this knowledge in representation language. In order to create a taxonomy for questions, we observe and analyze questions

Coarse Classes	Fine Classes
Quantification	Temprature, Time/Duration, Mass, Number, Age Distance, Money, Speed, Size, Percent, Rank/Rating
Entity	Person, Location, Organization, Animal, Technique Flora, Entertainment, Food, Abbreviation, Language Disease, Award/Title, Event, Sport/Game, Policy, Date Publication, Body, Thing, Feature/Attribute, Website Industry Sector, Monuments, Activity/Process, Other Tangible, Other Intangible
Definition	Person, Entity
Description	Reason, Mechanism, Cause & Effect, Describe Compare & Contrast, Analysis
List	Set of fine classes listed in the coarse classes *Quantification* and *Entity*
Selection	Alternative/Choice, True/False

Table 1: Set of proposed coarse and respective fine classes

from Stanford Question Answering Dataset (SQuAD) released by Rajpurkar et al. (2016) and question classifier data from Hovy et al. (2001) and Li and Roth (2002). The SQuAD dataset consists of 100,000+ questions and their answers, along with the text extracts from which the questions were formed. The other question classifier dataset contains $5,500$ questions. In the succeeding sub-section, we describe in details the coarse classes, fine classes and focus of a question. We have included an additional hierarchical taxonomy table with one example question for each class in the appendix section.

3.2.1 Coarse Classes

To choose the correct answer of a question one needs to understand the question and categorize the answer into the appropriate category which could vary from a basic implicit answer (question itself contains the answer) to a more elaborate answer (description). The coarse class of question provides a broader view of the expected answer type. We define the following six coarse class categories: *Quantification, Entity, Definition, Description, List* and *Selection*. *Quantification* class deals with the questions which look for a specific quantity as answer. Similarly *Entity, Definition, Description* class give the evidence that answer type will be entity, definition and a detail description, respectively. *Selection* class defines the question that looks for an answer which needs to be selected from the given set of answers. Few examples of questions along with their coarse class are listed here:

- **Quantity:** *Give the average speed of 1987 solar powered car winner?*
- **Entity:** *Which animal serves as a symbol throughout the book?*

3.2.2 Fine Classes

The coarse class defines the answer type at the broad level such as entity, quantity, description etc. But extracting the actual answer of question needs further classification into more specific answer types. Let us consider the following examples of two questions:

1. **Entity (Flora):** *What is one aquatic plant that remains submerged?*
2. **Entity (Animal):** *Which animal serves as a symbol throughout the book?*

Although both the questions belong to the same coarse class *entity* but they belong to the different fine classes, (*Flora* and *Animal*). Fine class of a question is based on the nature of the expected answer. It is useful in restricting the potential candidate matches. Although, questions belonging to the same fine class need not to be semantically same, questions belonging to the different fine classes rarely match. We show the set of the proposed coarse class and their respective fine classes in Table 1.

3.2.3 Focus of a Question

According to Moldovan et al. (2000), *focus* of a question is a word or a sequence of words, which defines the question and disambiguates it to find the correct answer the question is expecting to retrieve. In

the following example, *Describe the customer service model for Talent and HR BPO*, the term 'model' serves as the *focus*. As per Bunescu and Huang (2010b), *focus* of a question is contained within the noun phrases of a question. In the case of imperatives, the direct object (*dobj*) of the *question word* contains the *focus*. Similarly, in case of interrogatives, there are certain dependencies that capture the relation between the question word and its focus. The *dobj* relation of the root verb or *det* relation of *question word* for interrogatives contain the *focus*. Question word *how* has *advmod* relations that contain *focus* of the question. Priority order of the relations used to extract *focus* is obtained by observation on the SQuAD data. We depict the pseudo-code of the *focus* extraction method in the appendix section.

3.3 Question Classification

Question classification guides a QA system to extract appropriate candidate answer from the document/corpus. For example, the question *'How much does international cricket player get paid?'* should be accurately classified as the coarse class *quantification* and fine class *money* to further extract the appropriate answer. In our problem, we attempt to exploit the taxonomy information to identify the semantically similar questions. Therefore, the question classifier should be capable enough to accurately classify the coarse and fine classes of a reformulated question:

1. *What is the salary of an international level cricketer?*
2. *What is the estimated wage of an international cricketer?*

3.3.1 Question Classification Network

In order to identify the coarse and fine classes of a given question, we employ a deep learning based question classifier. In our question classification network CNN and bidirectional GRU has been applied sequentially. The obtained question vector is passed through a feed forward NN layer, and then through a softmax layer to obtain the final class of the question. We use two separate classifiers for coarse and fine class classification.

Firstly, an embedding layer maps a question $Q = [w_1, w_2 \dots w_n]$, which is a sequence of words w_i, into a sequence of dense, real-valued vectors, $E = [v_1, v_2 \dots v_n]$, $v_i \in \mathbb{R}^d$. Thereafter, a convolution operation is performed over the zero-padded sequence E^p. $F \in \mathbb{R}^{k \times m \times d}$, a set of k filters is applied to the sequence. We obtain convoluted features c_t at given time t for $t = 1, 2, \dots, n$.

$$c_t = tanh(F[v_{t-\frac{m-1}{2}} \dots v_t \dots v_{t+\frac{m-1}{2}}]) \tag{3}$$

Then, we generate the feature vectors $C' = [c'_1, c'_2 \dots c'_n]$, by applying max pooling on C. This sequence of convolution feature vector C' is passed through a bidirectional GRU network. We obtain the forward hidden states $\overrightarrow{h_t}$ and backward hidden states $\overleftarrow{h_t}$ at every step time t. The final output of recurrent layer h is obtained as the concatenation of the last hidden states of forward and backward hidden states.

Finally, the fixed-dimension vector h is fed into the softmax classification layer to compute the predictive probability $p(y = l|Q) = \frac{exp(w_l^T h + b_l)}{\sum_{i=1}^{L} exp(w_i^T h + b_i)}$ for all the question classes (coarse or fine). We assume there are L classes where w_x and b_x denote the weight and bias vectors, respectively and $x \in \{l, i\}$.

3.4 Comparison with Existing Taxonomy

In the Text REtrieval Conference (TREC) task, Li and Roth (2002) proposed a taxonomy to represent a natural semantic classification for a specific set of answers. This was built by analyzing the TREC questions. In contrast to Li and Roth (2002), along with TREC questions we also make a thorough analysis of the most recent question answering dataset (SQuAD) which has a collection of more diversified questions. Unlike Li and Roth (2002), we introduce the list and selection type question classes in our taxonomy. Each of these question types has its own strategy to retrieve an answer, and therefore, we put these separately in our proposed taxonomy. The usefulness of list as a different coarse class in semantic question matching can be understood considering the following questions:

1. *What are some techniques used to improve crop production?*
2. *What is the best technique used to improve crop production ?*

These two questions are not semantically similar as **(1)** and **(2)** belong to *list* and *entity* coarse classes, respectively. Moreover, Li and Roth (2002)'s taxonomy has overlapping classes (*Entity, Human and Location*). In our taxonomy we put all these classes in a single coarse class named *Entity*, which helps in identifying semantically similar questions better. We propose a set of coarse and respective fine classes with more coverage compared to Li and Roth (2002). Li and Roth (2002) taxonomy does not cover many important fine classes such as, *entertainment, award/title, activity, body* etc., under *entity* coarse class. We include these fine classes in our proposed taxonomy. We further redefine *description* type questions by introducing *cause & effect, compare and contrast* and *analysis* fine classes in addition to *reason, mechanism* and *description* classes. This finer categorization helps in choosing a more appropriate answer strategy for descriptive questions.

4 Experiments

4.1 Datasets

We perform experiments on three benchmark datasets, namely Partial Ordered Question Ranking (POQR)-Simple, POQR-Complex (Bunescu and Huang, 2010a) and Quora datasets. In addition to this, we also perform experiments on a new semantic question matching dataset (Semantic SQuAD[1]) created by us. In order to evaluate the system performance, we perform experiments in two different settings. The first setting deals with semantic question ranking (SQR) and the second deals with semantic question classification (SQC) with two classes (match and no-match). We perform SQR experiments on Semantic SQuAD and POQR datasets. For SQC experiments, we use Semantic SQuAD and Quora datasets.

4.1.1 Semantic SQuAD

We built a semantically similar question-pair dataset based on a portion of SQuAD data. SQuAD, a crowd-sourced dataset, consists of 100,000+ answered questions along with the text from which those question-answer pairs were constructed. We randomly selected $6,000$ question-answer pairs from SQuAD dataset and for a given question we asked 12 annotators[2] to formulate semantically similar questions referring to the same answers. Each annotator was asked to formulate 500 questions. We divided this dataset into training, validation and test sets of $2,000$ pairs each. We further constructed $4,000$ *semantically dissimilar* questions automatically. We use these $8,000$ question pairs ($4,000$ semantic similar questions pair from test and validation + $4,000$ semantically dissimilar pairs) to train the semantic question classifier for the SQC setting of the experiments. *Semantically dissimilar* questions are created by maintaining the constraint that questions should be from the different taxonomy classes. We perform 3-fold cross-validation on these $8,000$ question pairs.

4.1.2 POQR Dataset

POQR dataset consists of 60 groups of questions, each having a reference question that is associated with a partially ordered set of questions. Each group has three different sets of questions named as *paraphrase* ($\mathcal{P}$), *useful* ($\mathcal{U}$) and *neutral* ($\mathcal{N}$). For each given reference question q_r we have $q_p \in \mathcal{P}$, $q_u \in \mathcal{U}$, and $q_n \in \mathcal{N}$. As per Bunescu and Huang (2010a) the following two relations hold:

1. $(q_p \succ q_u | q_r)$: A *paraphrase* question is *'more useful than'* useful question.
2. $(q_u \succ q_n | q_r)$: A *useful* question is *'more useful than'* neutral question.

By transitivity, it was assumed by Bunescu and Huang (2010a) that the following ternary relation holds $(q_p \succ q_n | q_r)$: "A *paraphrase* question is *'more useful than'* a neutral question". We show the statistics of these datasets for *Simple* and *Complex* question types for two annotators (1, 2) in Table 2.

4.1.3 Quora Dataset

We perform experiments on semantic question matching dataset consisting of 404,290 pairs released by Quora[3]. The dataset consists of 149,263 matching pairs and 255,027 non-matching pairs.

[1]All the datasets used in the paper are publicly available at `https://figshare.com/articles/Semantic_Question_Classification_Datasets/6470726`

[2]The annotators are the post-graduate students having proficiency in English language.

[3]https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs

Datasets	Simple		Complex	
	Simple-1	Simple-2	Complex-1	Complex-2
$\mathcal{P}$	164	134	103	89
$\mathcal{U}$	775	778	766	730
$\mathcal{N}$	594	621	664	714
Pairs	11015	10436	10654	9979

Table 2: Brief statistics of POQR datasets

4.2 Evaluation Scheme

We employ different evaluation schemes for our SQR and SQC evaluation settings. For the **Semantic SQuAD** dataset, we use the following metrics for ranking evaluation: Recall in top-k results (Recall@k) for $k = 1$, 3 and 5, Mean Reciprocal Rank (MRR) and Mean Average Precision (MAP). The set of all candidate questions in $2,000$ pairs of the test set is ranked against each input question. As we have only 1 correct match out of $2,000$ questions for each question in the test set, recall@1 is equivalent to precision@1. Given that we only have one relevant result for each input question, MAP is equivalent to MRR. We evaluate the semantic question classification performance in terms of accuracy. To ensure fair evaluation, we keep the ratio of semantically similar and dissimilar questions to be 1:1. In order to compare the performance on **POQR dataset** with the state-of-the art results, we followed the same evaluation scheme as described in Bunescu and Huang (2010a). It is measured in terms of 10-fold cross validation accuracy on the set of ordered pairs, and the performance is averaged between the two annotators (1,2) for the Simple and Complex datasets. For **Quora dataset**, we perform 3-fold cross validation on the entire dataset evaluating based on the classification accuracy only. We did not perform the semantic question ranking (SQR) experiment on Quora dataset as 149,263 $\times$ 149,263 ranking experiment for matching pairs takes a very long time.

4.3 Baselines

We compare our proposed approach to the following information retrieval (IR) based baselines:
1) TF-IDF: The candidate questions are ranked using cosine similarity value obtained from the TF-IDF based vector representation.
2) Jaccard Similarity: The questions are ranked using Jaccard similarity calculated for each candidate question with the input question.
3) BM-25: The candidate questions are ranked using BM-25 score, provided by Apache Lucene [4]

4.4 Experimental Setup

Question Encoder: We train two different question encoders (hidden size=300) on *Semantic SQuAD* and *Quora* datasets. For Semantic SQuAD dataset, we used $2,000$ training pairs to train the question encoder, as mentioned in Section 4.1.1. For Quora dataset we randomly selected $74,232$ semantically similar question pairs to train the encoder, and $10,000$ question pairs for validation. The best hyper-parameters for the deep learning based attention encoder are identified on validation data. Adam (Kingma and Ba, 2014) is used as the optimization method. Other hyper-parameters used are: learning rate (0.01), dropout probability (Hinton et al., 2012): (0.5), CNN feature width (2), batch size (50), epochs (30) and size of the hidden state vectors (300). This optimal hyper-parameter values are same for the attention based RCNN and GRU encoder. We train two different question encoders trained on *Semantic SQuAD* and *Quora* datasets. We could not train the question encoder on the **POQR dataset** because of the unavailability of sufficient amount of similar question pairs in this dataset. Instead we use the question encoder trained on the Quora dataset to encode the questions from POQR dataset.

Question Classification Network: To train the model we manually label (using 3 English proficient

[4] https://lucene.apache.org/core/

annotators with an inter-annotator agreement of 87.98%) a total of $5,162$ questions[5] with their coarse and fine classes, as proposed in Section 3.2. We release this question classification dataset to the research community. We evaluate the performance of question classification for 5-fold cross-validation in terms of F-Score. Our evaluation shows that we achieve 94.72% and 86.19% F-score on coarse class (6-labels) and fine class (72-labels), respectively. We use this trained model to obtain the coarse and fine classes of questions in all datasets.

We perform the SQC experiments with SVM classifier. We use *libsvm* implementation (Chang and Lin, 2011) with linear kernel and polynomial kernel of degree $\in \{2, 3, 4\}$. Best performance was obtained using linear kernel. Due to the nature of POQR dataset as described in Section $4.1.2$ in the paper we employ SVM^{light} [6] implementation of ranking SVMs, with a linear kernel keeping standard parameters intact. In our experiments, we use pre-trained Google embeddings provided by (Mikolov et al., 2013). The focus embedding is obtained through word vector composition (averaging).

5 Results and Analysis

5.1 Results

We present extensive results of semantic question ranking experiment on the Semantic SQuAD dataset in Table 4. In Tables 3, 4 and 5 the performnce results are reported on the respective dataset using the models **GRU**, **RCNN, GRU-Attention and RCNN-Attention** (c.f. Section 3.1). For all these models the results reported in the tables are based on the cosine similarity of the respective question encoder. The introduction of attention mechansim helps the question encoder in improving the performance. The attention based model obatins the maximum gains of 2.40% and 2.60% in terms of recall and MRR for the *GRU* model. The taxonomy augmented model outperforms the respective baselines and state-of-the-art deep learning question encoder models. We obtain the best improvements for the *Tax+RCNN-Attention* model, 3.75% and 4.15% in terms of Recall and MRR, respectively. Experiments show that taxonomy features assist in consistently improving the R@k and MRR/MAP across all the models.

Models	Simple			Complex		
	Simple-1	**Simple-2**	**Overall**	**Complex-1**	**Complex-2**	**Overall**
GRU (Lei et al., 2016)	74.20	73.68	73.94	74.67	75.22	74.94
RCNN (Lei et al., 2016)	76.19	75.81	76.00	75.33	76.44	75.88
GRU-Attention	75.39	74.83	75.11	76.22	76.18	76.20
RCNN-Attention	77.28	77.01	77.14	76.63	77.31	76.97
DNN + Taxonomy based Features						
Tax+GRU	78.29	79.01	78.65	77.63	78.97	78.30
Tax+RCNN	80.92	81.55	81.23	80.15	80.83	80.49
Tax+GRU-Attention	81.69	81.03	81.36	81.22	81.56	81.39
Tax+RCNN-Attention	83.67	83.98	83.82	83.32	84.10	83.71
State-of-the art techniques						
Unsupervised *Cos* (Bunescu and Huang, 2010a)	-	-	73.70	-	-	72.60
Supervised *SVM* (Bunescu and Huang, 2010a)	-	-	82.10	-	-	82.50

Table 3: Semantic question ranking performance of various models on **POQR datasets**. All the numbers shows is in terms of accuracy.

Performance of the proposed model on POQR dataset are shown in Table 3. The '*overall*' column in Table 3 shows the performance average on simple-1,2 and complex-1,2 datasets. We obtain improvements (maximum of 1.55% with *GRU-Attention* model on Complex-1 dataset) in each model by introducing attention mechanism on both simple and complex datasets. The augmentation of taxonomy

[5] $4,000$ questions are the training set of Semantic SQuAD. Remaining $1,162$ questions from the dataset used in Li and Roth (2002)

[6] http://svmlight.joachims.org/

features helps in improving the performance further (8.75% with *Tax+RCNN-Attention* model on Simple dataset).

The system performance on semantic question classification (SQC) experiment with Semantic SQuAD and Quora datasets are shown in Table 5. Similar to ranking results, we obtain significant improvement by introducing attention mechanism and augmenting the taxonomy features on both the datasets.

Models	R@1	R@3	R@5	MRR/MAP
IR based Baselines				
TF-IDF	54.75	66.15	70.25	61.28
Jaccard Similarity	48.95	62.80	67.40	57.26
BM-25	56.40	69.35	71.45	61.93
Deep Neural Network (DNN) based Techniques				
GRU (Lei et al., 2016)	73.25	84.12	86.39	76.77
RCNN (Lei et al., 2016)	75.10	86.35	89.01	78.24
GRU-Attention	74.89	86.02	88.47	78.30
RCNN-Attention	76.41	88.41	91.78	80.28
DNN + Taxonomy based Features				
Tax + GRU	76.19	87.02	88.47	78.98
Tax +RCNN	78.32	88.91	92.35	81.49
Tax + GRU-Attention	77.35	89.22	91.28	80.95
Tax + RCNN-Attention	78.88	90.20	93.25	83.12

Table 4: **Semantic Question Ranking (SQR)** performance of various models on **Semantic SQuAD** dataset, R@k and Tax denote the recall@k & augmentation of taxonomy features.

Models	Semantic SQuAD Dataset	Quora Dataset
IR based Baselines		
TF-IDF	59.28	70.19
Jaccard Similarity	55.76	67.11
BM-25	63.78	73.27
Deep Neural network (DNN) based Techniques		
GRU (Lei et al., 2016)	74.05	77.53
RCNN (Lei et al., 2016)	77.54	79.32
GRU-Attention	75.18	79.22
RCNN-Attention	79.94	80.79
DNN + Taxonomy based Features		
Tax + GRU	77.32	79.21
Tax + RCNN	79.89	81.15
Tax + GRU-Attention	78.11	80.91
Tax + RCNN-Attention	82.25	83.17

Table 5: **Semantic Question Classification (SQC)** performance of various models on **Semantic SQuAD** and **Quora** datasets.

Sr. No.	Datasets	All	-CC	-FC	-Focus Word
1	Semantic SQuAD (SQR)	83.12	81.66	81.84	82.20
2	Semantic SQuAD (SQC)	82.25	80.85	81.19	81.13
3	POQR-Simple	83.82	80.85	81.44	82.57
4	POQR-Complex	83.71	81.04	81.97	82.19
5	Quora	83.17	80.93	81.75	82.24

Table 6: Feature ablation results on all datasets. **SQR** results are in **MAP**. The others results are shown in terms of **Accuracy**.

5.2 Qualitative Analysis

We analyze the obtained results by studying the following effects:

(1) Effect of Attention Mechanism: We analyzed hidden state representation the model is attending to when it is deciding the semantic similarity. We depicted the visualization (**in appendix**) of attention weight between two semantically similar question from Semantic SQuAD dataset. We observed that the improvement due to the attention mechanism in Quora dataset is comparatively less than the Semantic SQuAD dataset. The question pairs from Quora dataset have matching words, and the problem is more focused on difference rather than similar or related word. For example, for the questions "*How magnets are made?*" and "*What are magnets made of?*", the key difference is question words 'how' versus 'what', while the remaining words are similar.

(2) Effect of Taxonomy Features: We performed feature ablation study on all the datasets to analyze the impact of each taxonomy features. Table 4 shows the results[7] with the full features and after removing coarse class (-CC), fine class (-FC) and focus features one by one. We observed from Quora dataset that the starting word of the questions *(what, why, how etc.)* is a deciding factor for semantic similarity. As the taxonomy features categorize these questions into different coarse and fine classes, therefore, it helps the system in distinguishing between semantically similar and dissimilar questions. It can be observed from the results that the augmentation of CC and FC features significantly improves the performance

[7]The results are statistically significant as $p < 0.002$.

especially on Quora dataset. Similar trends were also observed on the other datasets.

5.3 Comparison to State-of-the-Art

We compare the system performance on POQR dataset with state-of-the-art work of Bunescu and Huang (2010a). Bunescu and Huang (2010a) used several cosine similarities as features obtained using bag-of-words, dependency tree, focus, main verb etc. Compared to Bunescu and Huang (2010a), our model achieves better performance with an improvement of 2.1% and 1.46% on simple and complex dataset respectively. A direct comparison to SemEval-2017 Task-3[8] CQA or AskUbuntu (Lei et al., 2016) datasets could not be made due to the difference in the nature of questions. The proposed classification method is designed for well-formed English questions and could not be applied to multi-sentence / ill-formed questions. We evaluate (Lei et al., 2016)'s model (RCNN) on each of our datasets and report the results in Section 5.1. Quora has not released any official test set yet. Hence, we report the performance of 3-fold cross validation on the entire dataset to minimize the variance. We can not directly make any comparisons with others due to the non-availability of an official gold test set.

5.4 Error Analysis

We observed the following as major sources of errors in the proposed system: **(1)** Misclassification at the fine class level is often propagated to semantic question classifier when some of questions contain more than one sentence. For e.g. "*What's the history behind human names? Do non-human species use names?*". **(2)** Semantically dissimilar questions having same function words but different coarse and fine class were incorrectly predicted as similar questions. It is because of the high similarity in the question vector and focus, which forces the classifier to commit mistakes. **(3)** In semantic question ranking (SQR) task, some of the questions with higher lexical similarity to the reference question are selected in prior to the actual similar question due to the high cosine similarity score with the reference question.

6 Conclusion

In this work, we have proposed an efficient model for semantic question matching where DL models are combined with pivotal features obtained from taxonomy. We have created a two layered taxonomy (coarse and fine) for questions in interest and proposed a deep learning based question classifier to classify the questions. We have established the usefulness of our taxonomy on two different task (SQR and SQC) on four different datasets. We have empirically established that effective usage of semantic classification and focus of questions helps in improving the performance of various on semantic question matching. Future work includes the efficient question encoders and handling community forum questions, which are often ill-formed, using taxonomy based features.

7 Acknowledgements

We acknowledge the partial support of Accenture IIT AI Lab. We also thank the reviewers for their insightful comments. Asif Ekbal acknowledges Young Faculty Research Fellowship (YFRF), supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Jiang Bian, Yandong Liu, Eugene Agichtein, and Hongyuan Zha. 2008. Finding the right facts in the crowd: factoid question answering over social media. In *Proceedings of the 17th international conference on World Wide Web*, pages 467–476. ACM.

[8] http://alt.qcri.org/semeval2017/task3/

Razvan Bunescu and Yunfeng Huang. 2010a. Learning the relative usefulness of questions in community qa. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 97–107. Association for Computational Linguistics.

Razvan Bunescu and Yunfeng Huang. 2010b. Towards a general model of answer typing: Question focus identification. In *Proceedings of The 11th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2010), RCS Volume*, pages 231–242.

Robin D Burke, Kristian J Hammond, Vladimir Kulyukin, Steven L Lytinen, Noriko Tomuro, and Scott Schoenberg. 1997. Question answering from frequently asked question files: Experiences with the faq finder system. *AI magazine*, 18(2):57.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Sumit Chopra, Michael Auli, and Alexander M Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Danilo Croce, Alessandro Moschitti, and Roberto Basili. 2011. Structured lexical similarity via convolution kernels on dependency trees. In *EMNLP*.

Danilo Croce, Simone Filice, Giuseppe Castellucci, and Roberto Basili. 2017. Deep learning in semantic kernel spaces. In *ACL*.

Minwei Feng, Bing Xiang, Michael R Glass, Lidan Wang, and Bowen Zhou. 2015. Applying deep learning to answer selection: A study and an open task. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 813–820. IEEE.

Simone Filice, Danilo Croce, Alessandro Moschitti, and Roberto Basili. 2016. Kelp at semeval-2016 task 3: Learning semantic relations between questions and answers. In *SemEval@NAACL-HLT*.

Thomas R. Gruber. 1995. Toward principles for the design of ontologies used for knowledge sharing. *Technical Report KSL 93-04*.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Eduard Hovy, Laurie Gerber, Ulf Hermjakob, Chin-Yew Lin, and Deepak Ravichandran. 2001. Toward semantics-based answer pinpointing. In *Proceedings of the first international conference on Human language technology research*, pages 1–7. Association for Computational Linguistics.

Mohit Iyyer, Jordan Boyd-Graber, Leonardo Claudino, Richard Socher, and Hal Daumé III. 2014. A neural network for factoid question answering over paragraphs. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 633–644, Doha, Qatar, October. Association for Computational Linguistics.

Jiwoon Jeon, W Bruce Croft, and Joon Ho Lee. 2005. Finding similar questions in large question and answer archives. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 84–90. ACM.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751. Association for Computational Linguistics.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Tao Lei, Hrishikesh Joshi, Regina Barzilay, Tommi Jaakkola, Kateryna Tymoshenko, Alessandro Moschitti, and Lluís Màrquez. 2016. Semi-supervised question retrieval with gated convolutions. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1279–1289, San Diego, California, June. Association for Computational Linguistics.

Shuguang Li and Suresh Manandhar. 2011. Improving question recommendation by exploiting information need. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1425–1434. Association for Computational Linguistics.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *Proceedings of the 19th international conference on Computational linguistics, COLING 2002*, pages 1–7. Association for Computational Linguistics.

Lluís Màrquez, James Glass, Walid Magdy, Alessandro Moschitti, Preslav Nakov, and Bilal Randeree. 2015. Semeval-2015 task 3: Answer selection in community question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

S Mlynarczyk and S Lytinen. 2005. Faqfinder question answering improvements using question/answer matching. *Proceedings of L&T-2005-Human Language Technologies as a Challenge for Computer Science and Linguistics*.

D Moldovan, S Harabagiu, M Pasca, R Mihalcea, R Goodrum, R Girji, and V Rus. 2000. Lasso: A tool for surfing the answer net. In *Proceedings 8th Text Retrieval Conference (TREC-8)*.

Preslav Nakov, Lluıs Marquez, Alessandro Moschitti, Walid Magdy, Hamdy Mubarak, Abed Alhakim Freihat, Jim Glass, and Bilal Randeree. 2016. Semeval-2016 task 3: Community question answering. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval*, volume 16.

Jun-Ping Ng and Min-Yen Kan. 2015. Qanus: An open-source question-answering platform. *arXiv preprint arXiv:1501.00311*.

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)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. *CoRR*, abs/1606.05250.

Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 373–382. ACM.

Di Wang and Eric Nyberg. 2015. A long short-term memory model for answer sentence selection in question answering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 707–712, Beijing, China, July. Association for Computational Linguistics.

Kai Wang, Zhaoyan Ming, and Tat-Seng Chua. 2009. A syntactic tree matching approach to finding similar questions in community-based qa services. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 187–194. ACM.

Yizhe Zhang, Dinghan Shen, Guoyin Wang, Zhe Gan, Ricardo Henao, and Lawrence Carin. 2017. Deconvolutional paragraph representation learning. In *Advances in Neural Information Processing Systems*, pages 4172–4182.

Guangyou Zhou, Yang Liu, Fang Liu, Daojian Zeng, and Jun Zhao. 2013. Improving question retrieval in community question answering using world knowledge. In *IJCAI*, volume 13, pages 2239–2245.

Appendices

A Proposed Taxonomy Table

	Coarse Classes	Fine Classes	Example
Non-Decision	**Quantification**	Temperature	What are the approximate temperatures that can be delivered by phase change materials?
		Time/Duration	How long did Baena worked for the Schwarzenegger/Shriver family?
		Mass	What is the weight in pounds of each of Schwarzenegger 's Hummers?
		Number	How many students are in New York City public schools?
		Distance	How many miles away from London is Plymouth?
		Money	What is the cost to build Cornell Tech?
		Speed	Give the average speed of 1987 solar powered car winner?
		Size	How large is Notre Dame in acres?
		Percent	What is the college graduation percentage among Manhattan residents?
		Age	How old was Schwarzenegger when he won Mr. Universe?
		Rank/Rating	What rank did iPod achieve among various computer products in 2006?
	Entity	Person	Who served as Plymouth 's mayor in 1993?
		Location	In what city does Plymouth 's ferry to Spain terminate?
		Organization	Who did Apple partner with to monitor its labor policies?
		Animal	Which animal serves as a symbol throughout the book?
		Flora	What is one aquatic plant that remains submerged?
		Entertainment	What album caused a lawsuit to be filed in 2001?
		Food	What type of food is NYC 's leading food export?
		Abbreviation	What does AI stand for?
		Technique	What is an example of a passive solar technique?
		Language	What language is used in Macedonia?
		Monuments	Which art museum does Notre Dame administer?
		Activity/Process	What was the name of another activity like the Crusades occurring on the Iberian peninsula?
		Disease	What kind of pain did Phillips endure?
		Award/Title	Which prize did Frederick Buechner create?
		Date	When was the telephone invented?
		Event	What event in the novel was heavily criticized for being a plot device?
		Sport/Game	Twilight Princess uses the control setup first employed in which previous game?
		Policy	What movement in the '60s did the novel help spark?
		Publication	Which book was credited with sparking the US Civil War?
		Body	What was the Executive Council an alternate name for?
		Thing	What is the name of the aircraft circling the globe in 2015 via solar power?
		Feature/Attribute	What part of the iPod is needed to communicate with peripherals?
		Industry Sector	In which industry did the iPod have a major impact?
		Website	Which website criticized Apple 's battery life claims?
		Other Tangible	In what body of water do the rivers Tamar and Plym converge?
		Other Intangible	The French words Notre Dame du Lac translate to what in English?
	Definition	Person	Who was Abraham Lincoln?
		Entity	What is a solar cell?
	Description	Reason	Why are salts good for thermal storage?
		Mechanism	How do the BBC 's non-domestic channels generate revenue?
		Cause & Effect	What caused Notre Dame to become notable in the early 20th century?
		Compare & Contrast	What was not developing as fast as other Soviet Republics?
		Describe	What do greenhouses do with solar energy?
		Analysis	How did the critics view the movie , " The Fighting Temptations "?
	List	Set of fine classes listed in the coarse classes *Quantification* and *Entity*	What are some examples of phase change materials? Which two national basketball teams play in NYC?
Decision	**Selection**	Alternative/Choice	Are the Ewell 's considered rich or poor?
		True/False	Is the Apple SDK available to third-party game publishers?

Table 7: The exemplar description of proposed taxonomy classes

B Algorithms

Algorithm 2 Question word extraction

procedure QUESTION WORD(QuesTokens)
  WhTags ← [WDT, WP, WP$, WRB]
  VbTags ← [VB, VBD, VBP, VBZ]
  for $t \in$ QuesTokens **do**
    if t.POS $\in$ *WhTags* **then return** t
  for $t \in$ QuesTokens **do**
    if t.POS $\in$ *VbTags* **then return** t

Algorithm 3 Focus Word Extraction

```
procedure FOCUS(QuesTokens)
    qw ← QUESTION WORD (QuesTokens)
    depP ← DependencyParse(QuesTokens)
    if qw is 'how' then
        return tail of 'advmod' of qw
    if qw.POS is V* then
        obj ← OBJECT(QuesTokens, qw)
        return obj
    if qw.POS is WH* then
        if 'root' is qw then
            nsubj ← tail of 'nsubj' of qw
            return nsubj
        else
            obj ← OBJECT (QuesTokens, 'root')
            return obj
    return <unk>
```

Algorithm 4 Object Extraction

```
procedure OBJECT(QuesTokens, qw)
    depP ← DependencyParse(QuesTokens)
    obj ← tail of 'det' of qw
    if obj not NULL then return obj
    obj ← tail of 'dobj' of qw
    if obj not NULL then return obj
    qw ← tail of 'conj:*' of qw
    obj ← tail of 'dobj' of qw
    if obj is NULL then
        comp ← tail of 'ccomp'/'xcomp' of qw
        obj ← tail of 'dobj' of comp
    return obj
```

C Additional Results

C.1 K-means Clustering

The k-means clustering was performed on the question representation obtained from the best question (RCNN-Attention) encoder of $2,000$ semantic question pairs. The clustering experiment was evaluated on the test set of Semantic SQuAD dataset (4000 questions). The performance was evaluated using the following metric:

$$\texttt{Recall} = \frac{100 \times \textit{no. of SQ pairs in same cluster}}{\textit{total no. of SQ pairs}} \tag{4}$$

K-means Clustering results are as follows: R@1:50.12, R@3:62.44 and R@5:66.58. As the number of clusters decreases `Recall` is expected to increase as there is higher likelihood of matching questions falling in the same cluster. `Recall` with $2,000$ clusters for $2,000$ SQ pairs i.e. 4,000 questions is comparable to Recall@1 as we have 2 questions per cluster on average, `Recall` with $1,000$ clusters is a proxy for Recall@3 and `Recall` with 667 clusters is comparable to Recall@5.

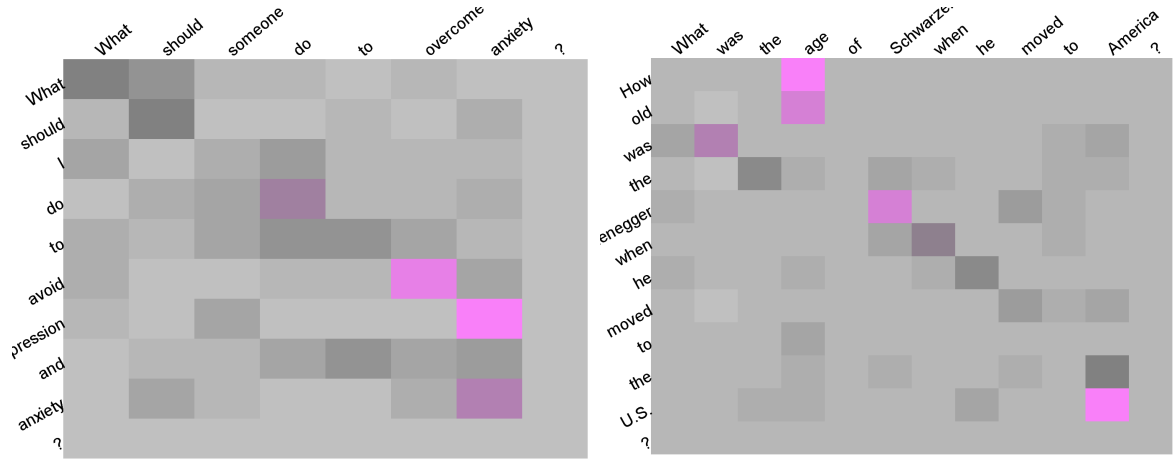
C.2 Semantic question classification (SQC) using IR-based Similarity

We have used TF-IDF, BM-25 and Jaccard similarity to classify a pair of question to similar or non-similar. We calculate the score between the question using the said algorithms thereafter a optimal thresholds are used to label a question pair as 'matching' or 'non-matching'. If the similarity score is greater than or equal to the threshold value we set the label 'matching' otherwise 'non-matching'. The optimal threshold value are calculated using the validation data. The optimal threshold value are given in the table 8.

Algorithm / Dataset	TF-IDF	BM-25	Jaccard Similarity
Semantic SQuAD Dataset	0.72	12.98	0.29
Quora Dataset	0.79	13.18	0.56

Table 8: IR based Optimal threshold value for each dataset

C.3 Attention Visualizations



(a) (b)

Figure 1: In **(a)** Attention mechanism detects semantically similar words (*avoid*, *overcome*). Attention mechanism is also able to align the multi-word expression '*how old*' to '*age*' as shown in **(b)**