Clinical Question Answering using Key-Value Memory Networks and Knowledge Graph

Sadid A. Hasan, Siyuan Zhao, Vivek Datla, Joey Liu,
Kathy Lee, Ashequl Qadir, Aaditya Prakash,* Oladimeji Farri
Artificial Intelligence Laboratory, Philips Research North America, Cambridge, MA, USA
{sadid.hasan,siyuan.zhao,vivek.datla,joey.liu}@philips.com
{kathy.lee_1,ashequl.qadir,aaditya.prakash,Dimeji.Farri}@philips.com

Abstract

In this paper, we describe our clinical question answering system implemented for the Text Retrieval Conference (TREC 2016) Clinical Decision Support (CDS) track. We submitted five runs using a combination of knowledge-driven (based on a *curated knowledge graph*) and deep learning-based (using *key-value memory networks*) approaches to retrieve relevant biomedical articles for answering generic clinical questions (diagnoses, treatment, and test) for each clinical scenario provided in three forms: *notes*, *descriptions*, and *summaries*. The submitted runs were varied based on the use of notes, descriptions, or summaries in association with different diagnostic inferencing methodologies applied prior to biomedical article retrieval. Evaluation results demonstrate that our systems achieved best or close to best scores for 20% of the topics and better than median scores for 40% of the topics across all participants considering all evaluation measures. Further analysis shows that on average our clinical question answering system performed best with *summaries* using diagnostic inferencing from the knowledge graph whereas our key-value memory network model with *notes* consistently outperformed the knowledge graph-based system for *notes* and *descriptions*.

1 Introduction

Similar to the last two years, the main objective of the 2016 CDS track[1] was to retrieve a ranked list of the top 1000 biomedical articles that can answer generic clinical questions related to three categories: diagnosis, test, and treatment. Like our previous two participations in this track, we consider the importance of inferring the most probable clinical diagnosis from the given free text clinical scenario prior to biomedical article retrieval (Hasan et al., 2014; Hasan et al., 2015). Hence, we submitted five runs using a variety of diagnostic inferencing techniques (knowledge graph-based and key-value memory network-based) to address the given clinical questions.

Knowledge graphs can embed structured data sources into a collection of facts about entities and have been shown to provide a better knowledge management capability in recent years (Rospocher et al., 2016). We propose a novel knowledge graph-based clinical diagnostic inferencing technique that can provide the most relevant diagnoses by analyzing the underlying context of the clinical narratives.

We use the Wikipedia clinical medicine category pages to build a directed knowledge graph (digraph), which possesses symptoms as leaf nodes and are connected to the diseases and medical conditions. The digraph is grounded as the activations flow directly from the leaf nodes to the entire graph. This grounded digraph-based approach uses the activation-decay cycles to identify the most probable diagnosis given the description of the patient scenario in natural language.

Memory Networks (MemNN) is a class of models, which contains scalable memory with a learning component to read from and write to it (Weston et al., 2014). A variant of memory networks (Sukhbaatar et al., 2015; Miller et al., 2016; Chandar et al., 2016) are proposed in recent years

*The author is also affiliated with Brandeis University (aprakash@brandeis.edu).

[1] http://www.trec-cds.org/

to solve complex reasoning and inferencing tasks (e.g. bAbI tasks, MovieQA, WikiQA). Inspired by the success of Key-Value Memory Networks (KV-MemNN) (Miller et al., 2016), we adapt the KV-MemNN model to perform diagnostic inferencing from the given free text clinical narratives.

Compared to Long Short-Term Memory Networks (LSTM) (Hochreiter and Schmidhuber, 1997), memory networks can pertain to both long-term and short-term memory, and are flexible enough to store a richer representation of input in memory. To use these advantages, we extract knowledge (from Wikipedia pages under the clinical medicine category) for each diagnosis and store it to memory to help model infer the most probable diagnosis. Note that, the knowledge for each diagnosis is the free text extracted from the corresponding Wikipedia page. To the best of our knowledge, this is the first work that applies memory networks to such a complex task like clinical diagnostic inferencing.

Evaluation results showed the effectiveness of our diagnostic inferencing techniques for accurate retrieval of relevant biomedical articles for the automated clinical question answering task. In the next sections, we describe the overall architecture of our system, and present the evaluation results with analyses.

2 System Description

Our overarching approach centers on three steps: (i) Topical Keyword Analysis: identifying the most clinically relevant keywords from the given topic descriptions, summaries, and clinical notes; (ii) Diagnostic Inferencing: reasoning based on the topical keywords to generate the diagnoses, tests, and treatments using the underlying clinical contexts represented within either a Key-Value Memory Network or a Knowledge Graph, both powered by an external clinical knowledge source; and, (iii) Relevant Article Retrieval: retrieving and ranking pertinent biomedical articles based on the topical keywords and clinical inferences from steps (i) and (ii).

The submitted runs are varied based on the use of topical descriptions, summaries or notes in association with different diagnostic inferencing methodologies. We describe these steps in the next subsections.

2.1 Topical Keyword Analysis

As the first step of our four submitted runs (*run1* to *run4*), similar to Hasan et al. (2015), we extract term frequency-inverse document frequency (TF-IDF) weighted topical keywords from the given descriptions, summaries or notes and map them to categories represented in the following controlled clinical ontologies: SNOMED CT[2] (Cornet and de Keizer, 2008) for diagnoses, LOINC[3] for tests, and RxNorm[4] for treatments. Prior studies have shown the effectiveness of using clinical domain ontologies to semantically categorize clinical concepts (Bodenreider, 2008; Stenzhorn et al., 2008; Garde et al., 2007). Furthermore, we identify relevant demographic information, interpret vital signs based on standard normal range values, and filter out negated clinical concepts in order to give more weight to positive clinical manifestations in a given clinical scenario.

2.2 Diagnostic Inferencing

In this key step, for three runs (*run1*, *run2*, and *run4*), we use the extracted topical concepts from the previous step to infer relevant diagnoses, test, and treatment concepts from a clinical knowledge base derived from Wikipedia articles in the clinical medicine[5] category, and embedded into a novel knowledge graph-based architecture (see details in Section 2.2.1).

For *run3*, we use the similar diagnostic inferencing approach as Hasan et al. (2015), where we directly refer to the Wikipedia clinical knowledge base articles (indexed using Elasticsearch[6]) to extract a list of candidate articles with relevant diagnoses corresponding to each topical keyword extracted in step 1. Candidate Wikipedia articles were filtered using various criteria e.g., location, gender, match with topical keywords etc., and then, the resulting list of Wikipedia articles with relevant clinical concepts were mined to retrieve specific diagnoses (from the title of the Wikipedia article).

[2] http://www.ihtsdo.org/snomed-ct/
[3] http://loinc.org/
[4] http://www.nlm.nih.gov/research/umls/rxnorm/
[5] https://en.wikipedia.org/wiki/Category:Clinical_medicine
[6] http://www.elasticsearch.org/

For *run5*, we build a novel end-to-end diagnostic inferencing model using Key-Value Memory Networks (Miller et al., 2016) trained on a large collection of MIMIC-II discharge notes (Saeed et al., 2011) along with the Wikipedia clinical knowledge base in order to capture the overall context of a given clinical note towards inferring the most probable diagnoses (see details in Section 2.2.2).

The list of possible diagnoses identified for all runs is used to extract a list of candidate Wikipedia articles to mine related tests, and treatments (from sections and subsections of the Wikipedia article) accordingly.

2.2.1 Using Knowledge Graph

We use the Wiki pages under the clinical medicine category to build a knowledge graph. The hierarchy of each Wiki page is preserved to encode its distinguishing characteristics with respect to other pages. Each page consists of several sections and is related to other medical conditions. We build a directed graph (digraph) by using these relations, where each node is a medical condition, diagnosis, test, procedure, medication or any other clinical concept, and each edge is a relation between two nodes. Note that, if a page has a hyperlink to another page, then the direction of the edge is from the current page to the other page. The constructed knowledge graph using this approach contains ~$100K$ nodes and ~$1M$ edges, where leaf nodes represent medical symptoms and are connected to relevant diseases and medical conditions.

The directed knowledge graph is grounded as the activations flow directly from the leaf nodes to the entire graph. This grounded digraph based approach exploits the activation-decay cycles to identify the most probable diagnosis given a clinical narrative (summary, description, or note). When the TF-IDF weighted clinical concepts extracted from the clinical narrative (see Section 2.1) are used to query the knowledge graph, we perform all one-hop expansion of the symptom nodes towards building a digraph with the activation weights initialized to the associated TF-IDF weights. The nodes of the initial scattered forests having the least number of children are then expanded such that a connected graph is formed. This expansion is based on a minimal context addition principle, where the objective is to build a connected digraph by minimizing the number of nodes. The expansion is discontinued when we have a spanning tree structure. The activation module spreads the activation across the digraph and is controlled using a sigmoid function. Only partial activation flows to its children as inheritance of activation is proportional to number of siblings of the current node. Activation is a continuous process and it spreads from parent to children across the nodes in the same fashion. As the activation spreads concurrently, we decay the activation. Each time during the inheritance of activation the nodes lose a variable amount of activation based on the distance of a node from the initial node. Therefore, the nodes that are farther away from the base receive the most decayed activation.

The control module monitors the activation and decay cycle, and ensures that there is no runaway activation among the nodes. This module also controls the accumulation of activations at each node and stops the activation and decay cycle when the network is stabilized. Once the network is stable, the top ranked diseases and medical conditions are extracted from the knowledge graph. Then, the demographic information obtained from the clinical narrative (see Section 2.1) is leveraged to fine-tune the ranking. For example, if a disease is not common for a demographic, its rank is lowered.

2.2.2 Using Key-Value Memory Networks

Key-Value Memory Networks (KV-MemNN) (Miller et al., 2016) contains key-value paired memories, which uses a generalized approach of how the information is stored in memory. To solve Question Answering (QA) tasks, KV-MemNN first stores facts in key-value paired memory, uses the key to address relevant memories with respect to the question, and then extracts corresponding values. The addressing step takes place on the key memory and the reading step occurs on the value memory. The key is designed with features to help match it to the question (interest), while the value is designed with features to help match it to the final answer.

We adapt the KV-MemNN model to perform diagnostic inferencing from the given free text clinical narratives. We extract knowledge (from Wikipedia pages under the clinical medicine category) for each diagnosis and store it to memory to help model infer

the most probable diagnosis. Note that, the knowledge for each diagnosis is the free text extracted from the corresponding Wikipedia page. Below, we present a general framework on how we collect data and knowledge base, represent them in the memory, and train the model.

Dataset: We use the MIMIC-II (Multiparameter Intelligent Monitoring in Intensive Care) dataset (Saeed et al., 2011), which contains physiologic signals and vital signs in a time series format captured from patient monitors, and comprehensive clinical data obtained from hospital medical information systems, for tens of thousands of Intensive Care Unit patients.

We use the MIMIC-II discharge notes for our experiments, which generally contain comprehensive clinical scenarios represented as unstructured free texts. We separate diagnosis from each medical record to create a collection of *<medical note, diagnosis>* pairs from this dataset. Then, we collect knowledge for each diagnosis from the Wikipedia pages under the clinical medicine category (described in the following paragraph). Some diagnoses only have few instances in the data set. Without enough training instances, the model may not be able to learn to recognize these diagnoses. Hence, we only select the most common diagnoses with *frequency value* > 50 yielding to 71 diagnoses for $8K$ medical note instances and thus, formulate the clinical diagnostic inferencing task as a multiclass-multilabel classification problem.

Knowledge Base: Wikipedia is a reasonable source for medical domain knowledge as WikiProject Medicine[7] is dedicated to improving the quality of medical articles in Wikipedia (Trevena, 2011). Since certain diagnosis terms from MIMIC-II do not exactly match the Wikipedia page titles, we use the Wikipedia API[8] to search for the most appropriate Wiki page by using each diagnosis term as the search keyword. Note that, the title of each Wikipedia page is the name of the diagnosis described by the page. The first section of such a Wiki page normally contains an introduction to the diagnosis. Among several other sections inside the Wiki page, the *"Signs and symptoms"* section describes the classic and common signs and symptoms for the diagnosis. Each collected Wikipedia page is turned into a key-value pair by using the following principle: the free text from the first section and the sections for sign and symptoms is the key and the title of the page is the value.

Model Description: Similar to the original KV-MemNNs model (Miller et al., 2016), in our proposed formulation of the clinical diagnostic inferencing task, the memory slots are defined as pairs of vectors $(k_1, v_1), (k_2, v_2), (k_m, v_m)$, where m is the size of memory, and clinical notes from MIMIC-II are denoted as x. The addressing and reading of the memory involve three steps:

- **Key Addressing:** Each memory slot is associated with a probability by measuring the similarity between the medical note and each key:

$$p_{h_i} = Softmax(A\Phi_X(x) \cdot A\Phi_K(k_{h_i})) \quad (1)$$

  where Φ are the feature maps of dimension D, and A denotes a $d \times D$ matrix. The softmax function is computed as: $Softmax = \exp(z_i)/\sum_j \exp(z_j)$. Note that, the medical note n is represented by $A\Phi_X(x)$.

- **Value Reading:** The reading output vectors o are computed by taking a weighted sum of the memory values based on the probabilities calculated at the previous step:

$$o = \sum_i p_{h_i} A\Phi(v_{h_i}) \quad (2)$$

- **Note Updating:** After calculating o, the medical note is updated with the following equation:

$$n_{i+1} = R_i(n_i + o) \quad (3)$$

  where R denotes a $d \times d$ matrix.

These three steps are repeated with a different matrix R_i in each hop. After a fixed number of H hops, the final probability for each diagnosis is computed using the final result o over all possible diagnoses:

[7] https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Medicine
[8] https://www.mediawiki.org/wiki/API:Main_page

$$\hat{p} = sigmoid(n_{H+1}^T B \Phi_Y(y_i)) \quad (4)$$

where y_i represents a possible diagnosis and B is a $d \times D$ matrix.

The model is trained in an end-to-end fashion. Backpropagation and stochastic gradient descent algorithms are used to learn the parameters A, B and $R_1, \cdots, R_H$.

Document Representation: We use a simple bag-of-word (BoW) representation that transfers each word w_{ij} in the document $d_i = w_{i1}, w_{i2}, w_{i3}, \cdots, w_{in}$ to corresponding vector embeddings and sums these together to the resulting vectors: $\Phi(d_i) = \sum_j A w_{ij}$, where A denotes the embedding matrix.

2.3 Relevant Article Retrieval

As the final step for all runs, topical keywords and the corresponding diagnoses, tests, and treatments obtained from the diagnostic inferencing step are used to retrieve candidate biomedical articles by searching through the given TREC-CDS corpus of over 1.25M PubMed Central[9] articles (indexed using Elasticsearch). Similar to Hasan et al. (2014) and Hasan et al. (2015), the retrieved candidate articles are ranked using multiple weighting algorithms specific to the three types of clinical questions (diagnosis, test, and treatment). The biomedical articles are further filtered by location (e.g. USA/Canada), demographic information and other contextual information from the topic description, summary or note towards improving the relevance of the results. The final list of top 1000 biomedical articles is ordered by article publication date to provide chronological biomedical evidence for the answers to each topic.

3 Experimental Setup

3.1 Test Data

Similar to last two years, the test dataset comprises 30 topics divided into three question types: topic 1-10 (diagnosis), topic 11-20 (test), and topic 21-30 (treatment). The given topics are essentially medical case narratives that describe scenarios related to patient's medical history, signs/symptoms, diagnoses, tests, and treatments. The topics are provided in three versions depending on the depth of information. Besides topic "descriptions" that include comprehensive descriptions of the patient's situation and topic "summaries" that contain an abridged version of the most important information, topic "notes" are introduced this year, which are actual admission notes derived from MIMIC-III (Johnson et al., 2016) containing numerous abbreviations and domain-specific jargons.

3.2 Corpus

A snapshot of the open access portion of PubMed Central (PMC), a freely available online database of full-text biomedical articles comprising $1.25M$ biomedical publications was made available by the TREC CDS track organizers this year.

3.3 Run Description

We submitted five runs as follows: 1) *prna1sum*: considers topic summaries with knowledge graph-based diagnostic inferencing, 2) *prna2desc*: considers topic descriptions with knowledge graph-based diagnostic inferencing, 3) *prna3note*: considers topic notes with diagnostic inferencing by directly accessing the clinical knowledge base (Hasan et al., 2015), 4) *prna4note*: considers topic notes with knowledge graph-based diagnostic inferencing, and 5) *prna5note*: considers topic notes with KV-MemNN-based diagnostic inferencing.

Our KV-MemNN model (for run *prna5note*) was implemented using the TensorFlow[10] framework. We used Adam stochastic gradient descent (Kingma and Ba, 2014) for optimizing the learned parameters. The learning rate was set to 0.005 and the batch size for each iteration was set to 100. As the final prediction layer, we used a fully connected layer on top of the output layer from Eq. 4. The model learned the parameters by minimizing a standard cross-entropy loss between a predicted diagnosis and the correct diagnosis. For regularization, we used dropout (Srivastava et al., 2014) with the probability 0.5 at the end of each hop and limit the norm of the gradients to below 4. We trained the model on 80% of the data for 200 epochs using batch gradient descent while the remaining 20% data was equally divided to a validation and a testing set. All hyperparameters were

[9] http://www.ncbi.nlm.nih.gov/pmc/

[10] https://www.tensorflow.org/

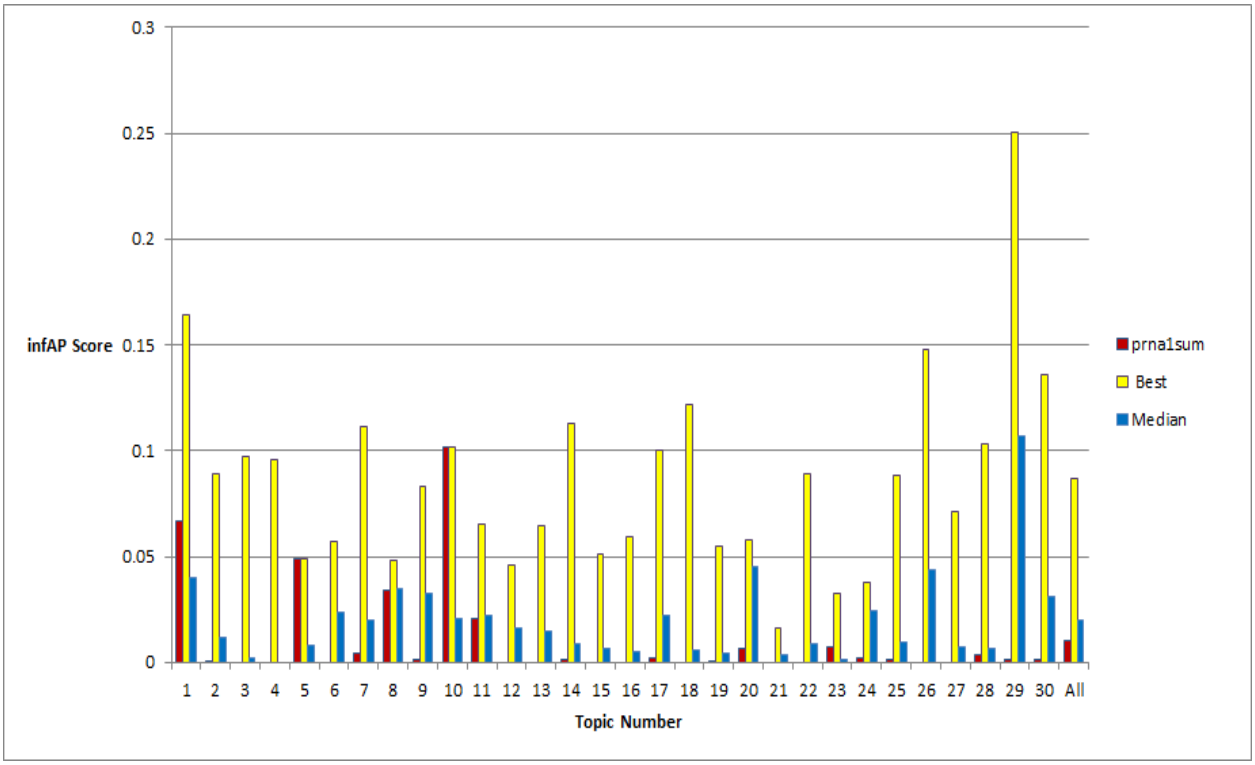

Figure 1: infAP scores for each topic (comparison with participant runs using *summaries* with knowledge-graph)

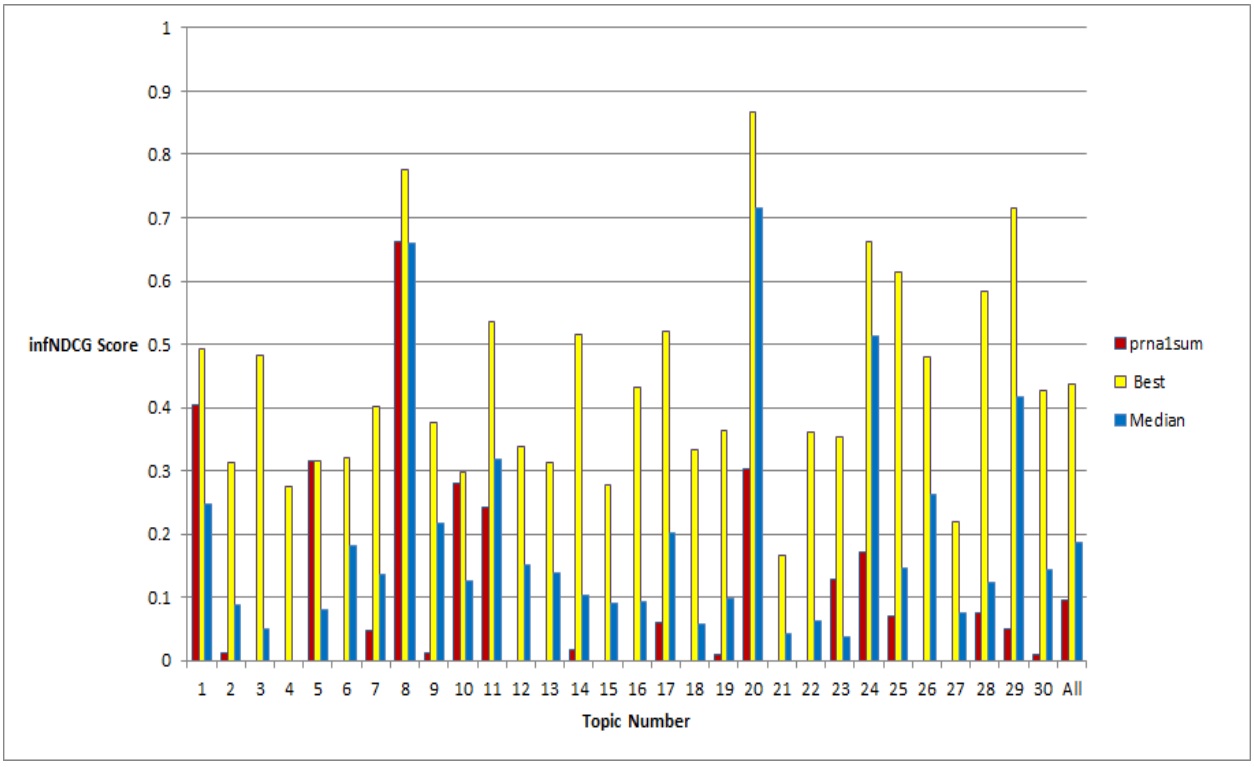

Figure 2: infNDCG scores for each topic (comparison with participant runs using *summaries* with knowledge-graph)

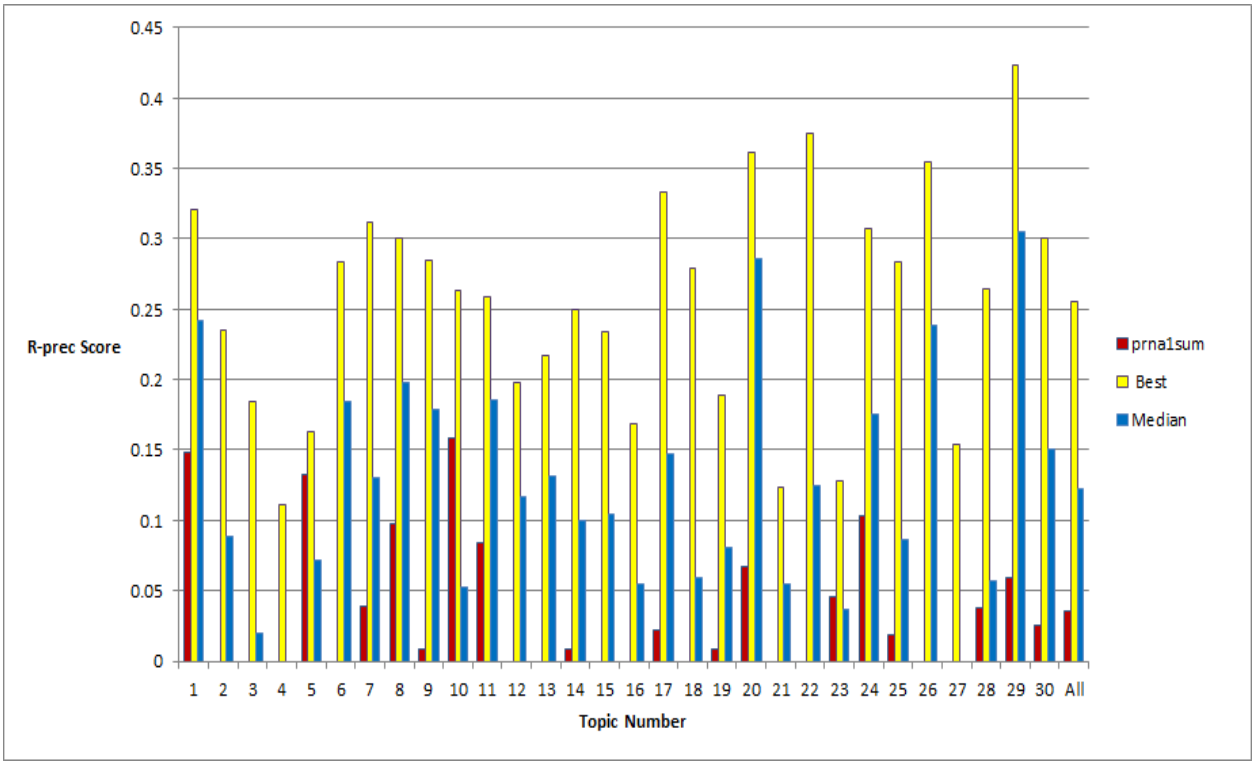

Figure 3: R-prec scores for each topic (comparison with participant runs using *summaries* with knowledge-graph)

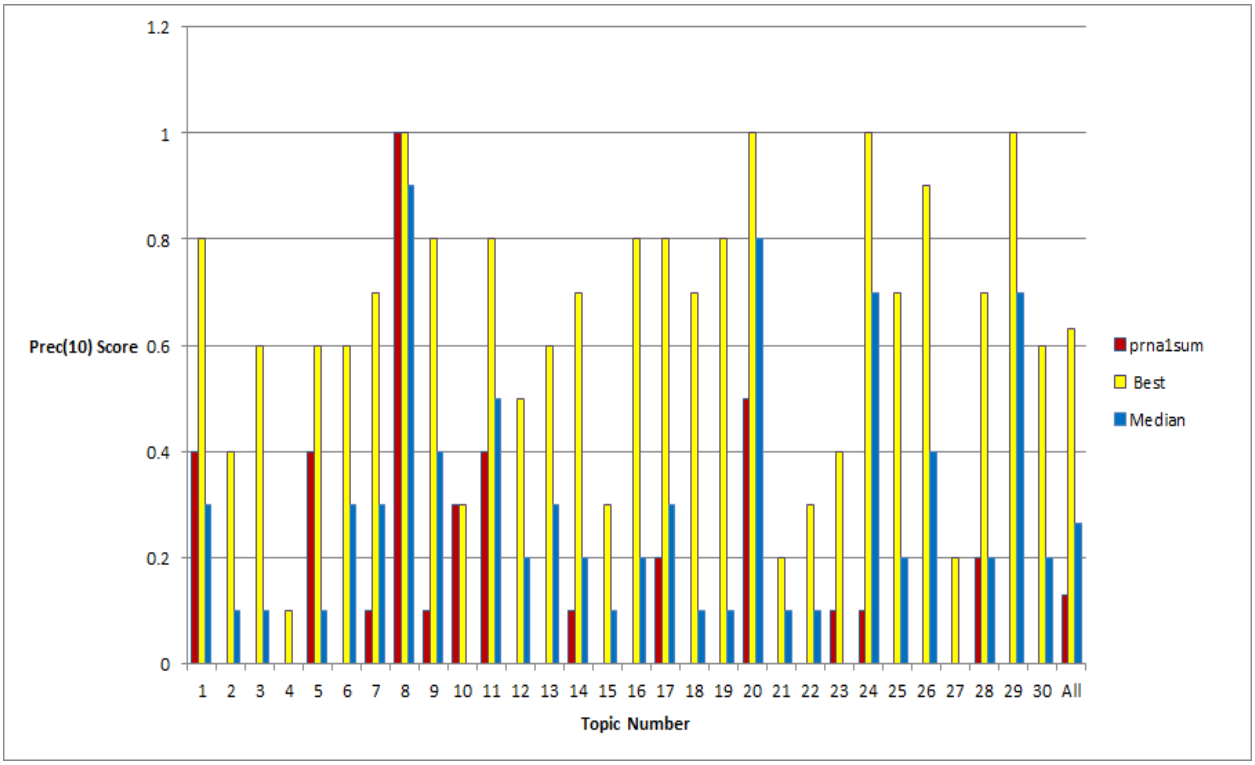

Figure 4: Prec(10) scores for each topic (comparison with participant runs using *summaries* with knowledge-graph)

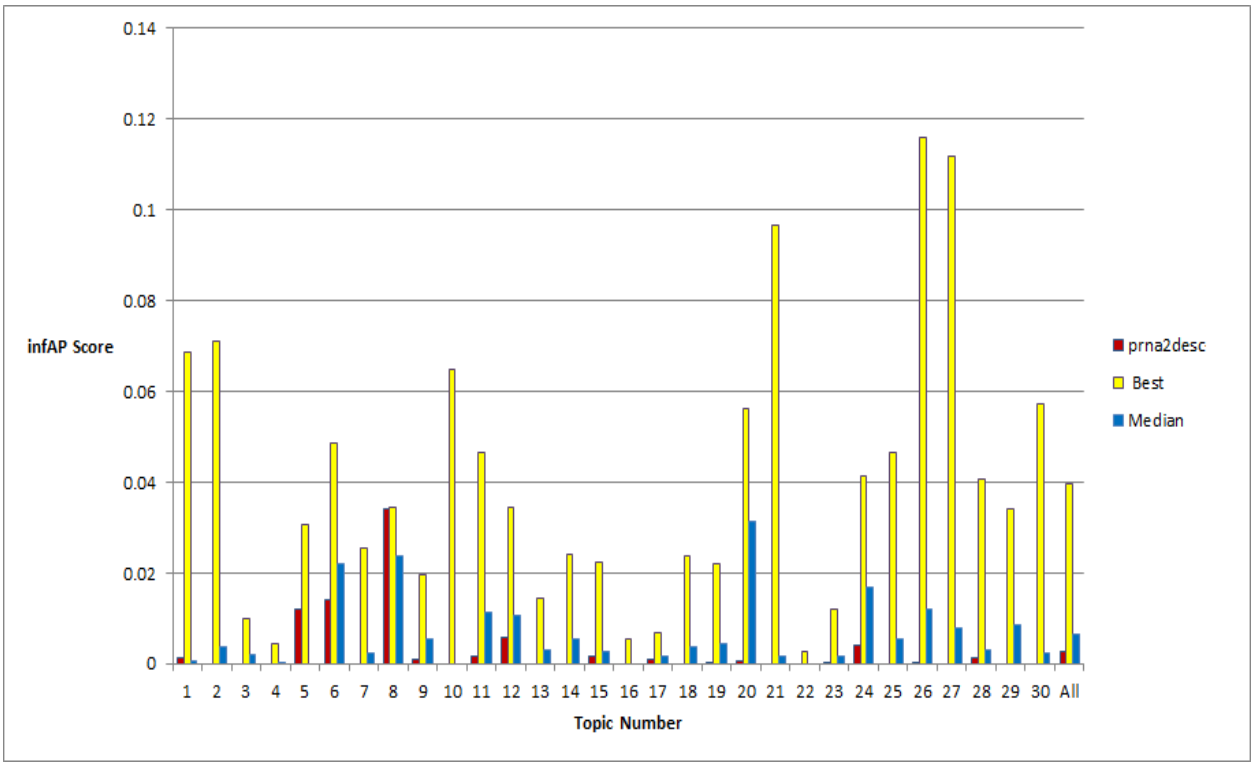

Figure 5: infAP scores for each topic (comparison with participant runs using *descriptions* with knowledge-graph)

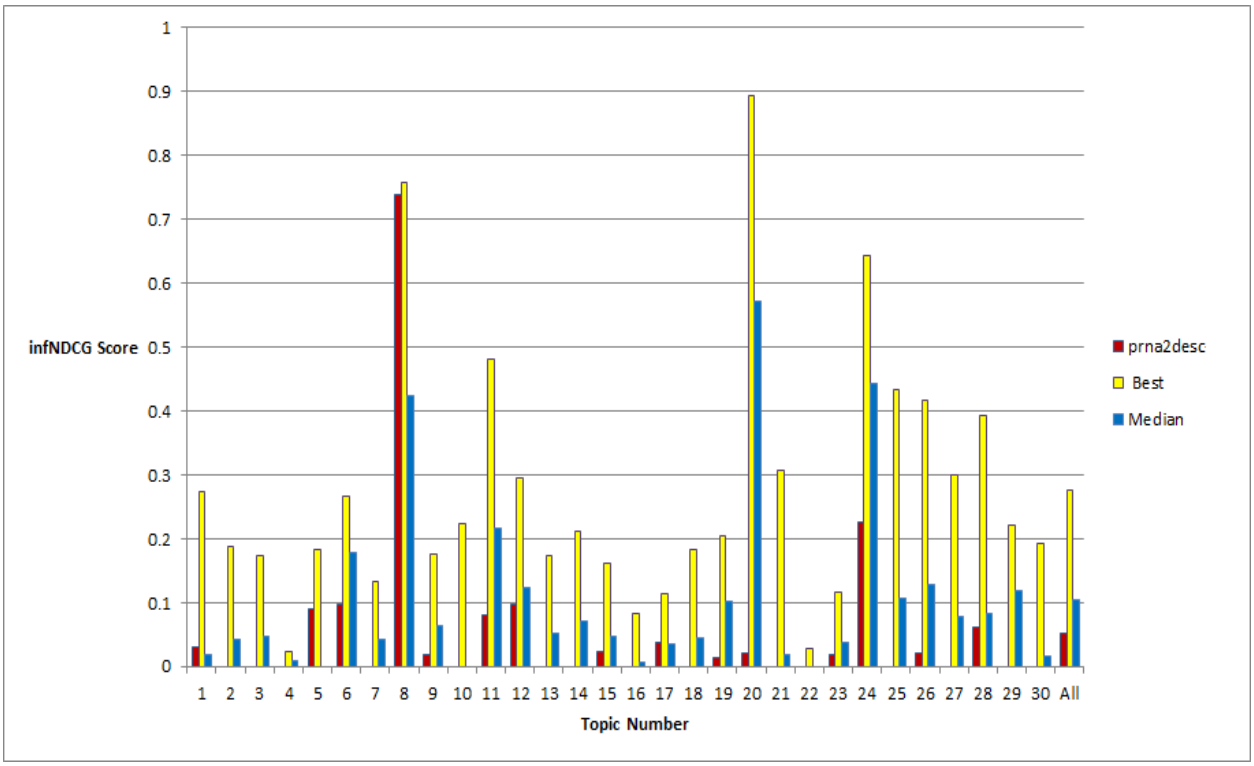

Figure 6: infNDCG scores for each topic (comparison with participant runs using *descriptions* with knowledge-graph)

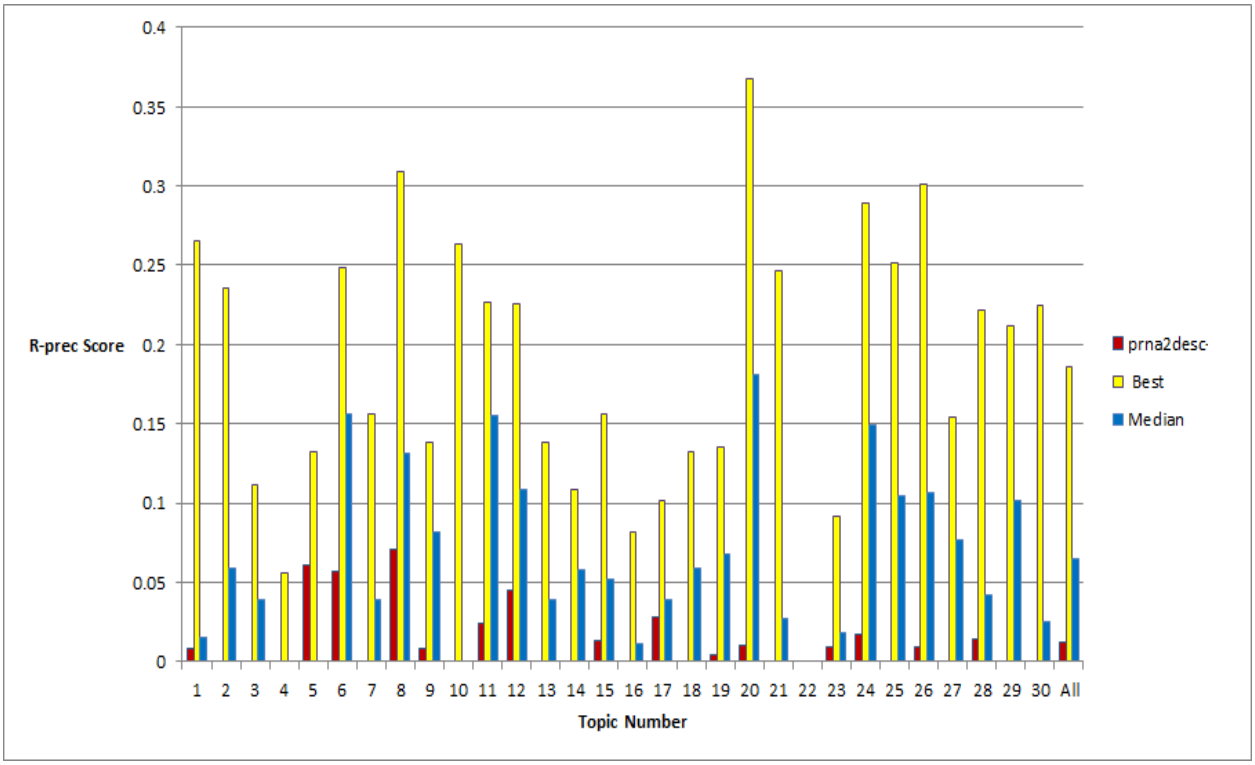

Figure 7: R-prec scores for each topic (comparison with participant runs using *descriptions* with knowledge-graph)

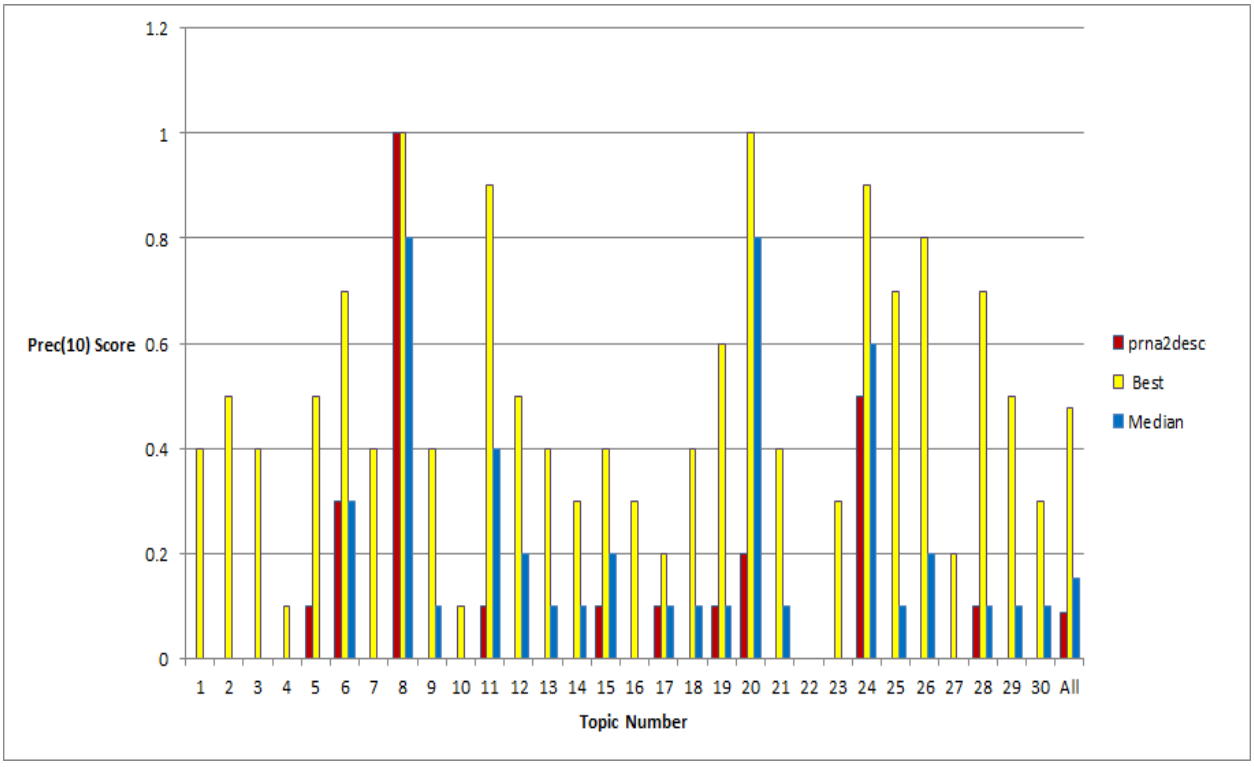

Figure 8: Prec(10) scores for each topic (comparison with participant runs using *descriptions* with knowledge-graph)

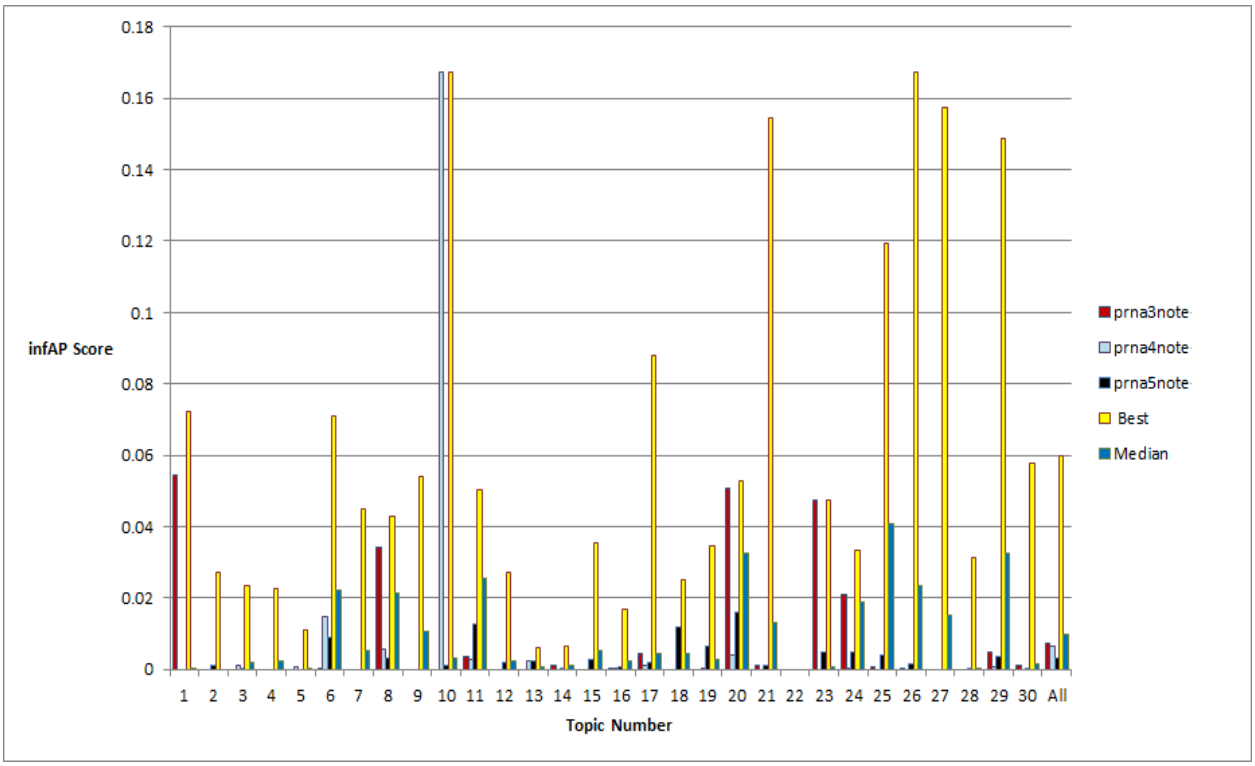

Figure 9: infAP scores for each topic (comparison with participant runs using *notes* with inferencing similar to Hasan et al. (2015), knowledge graph, and key-value memory networks, respectively)

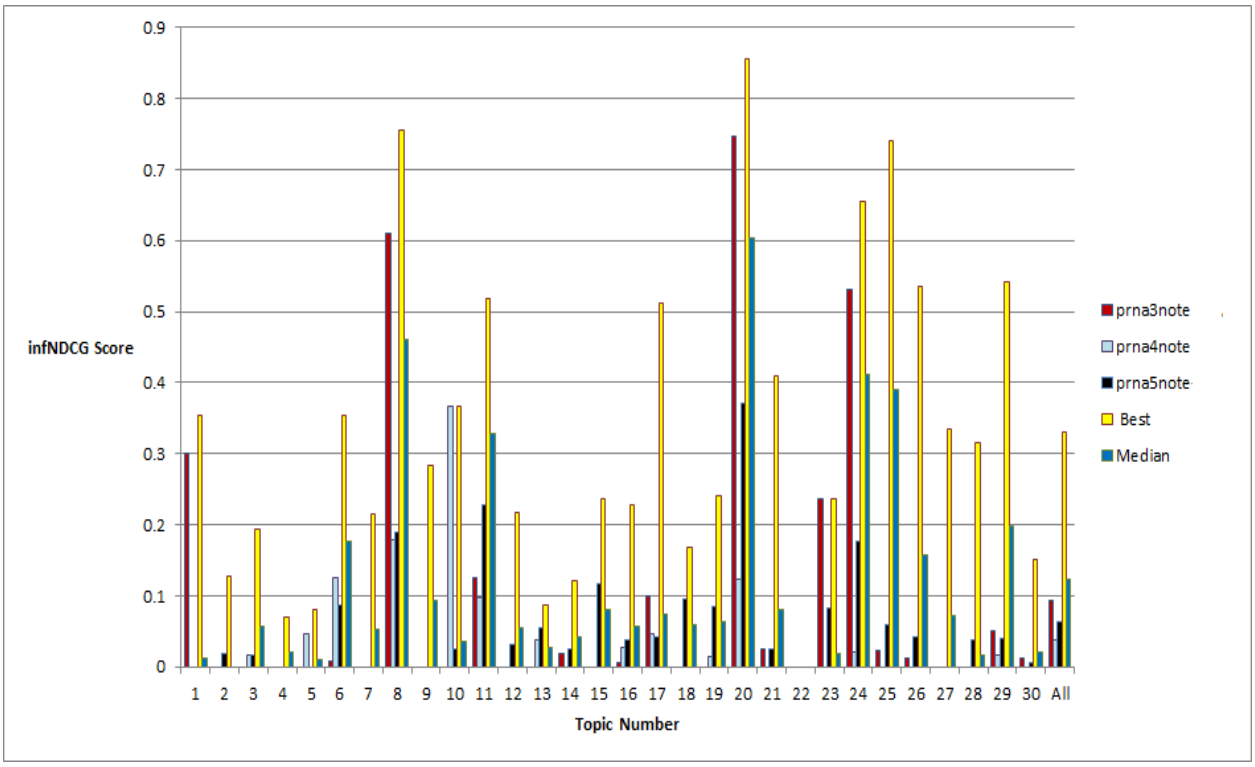

Figure 10: infNDCG scores for each topic (comparison with participant runs using *notes* with inferencing similar to Hasan et al. (2015), knowledge graph, and key-value memory networks, respectively)

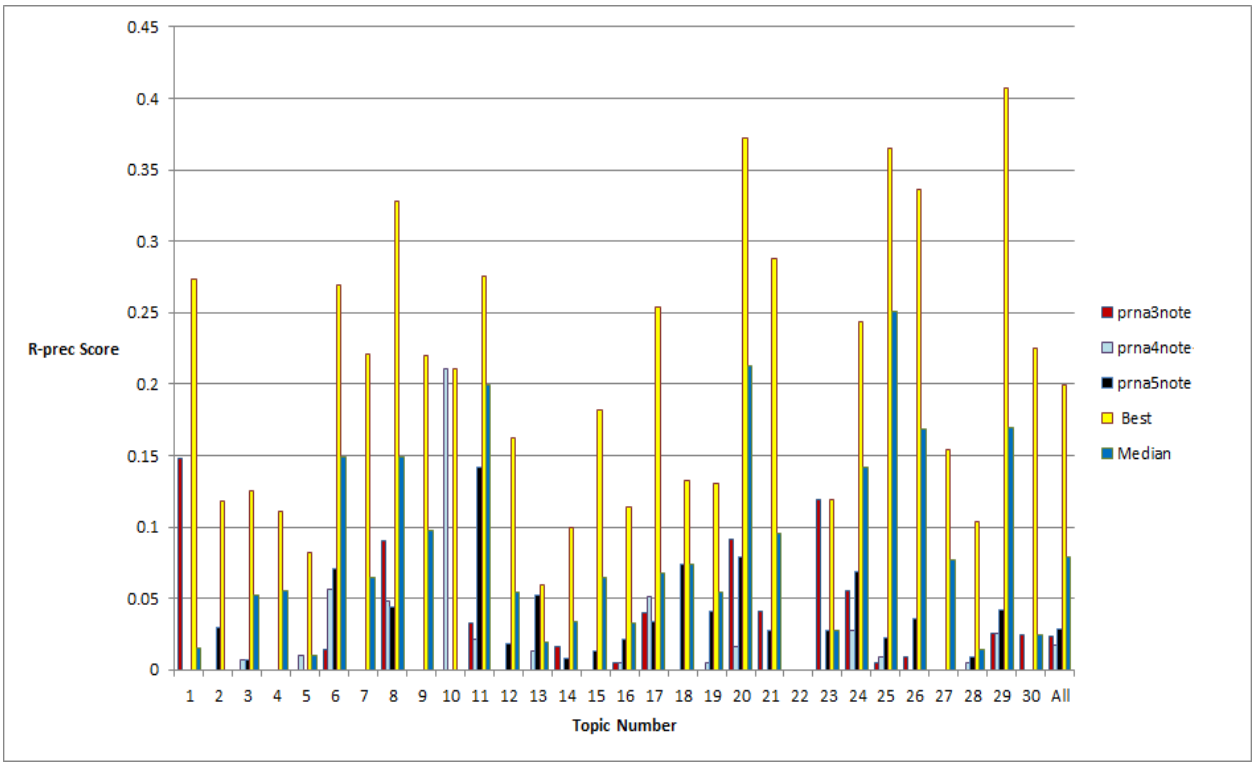

Figure 11: R-prec scores for each topic (comparison with participant runs using *notes* with inferencing similar to Hasan et al. (2015), knowledge graph, and key-value memory networks, respectively)

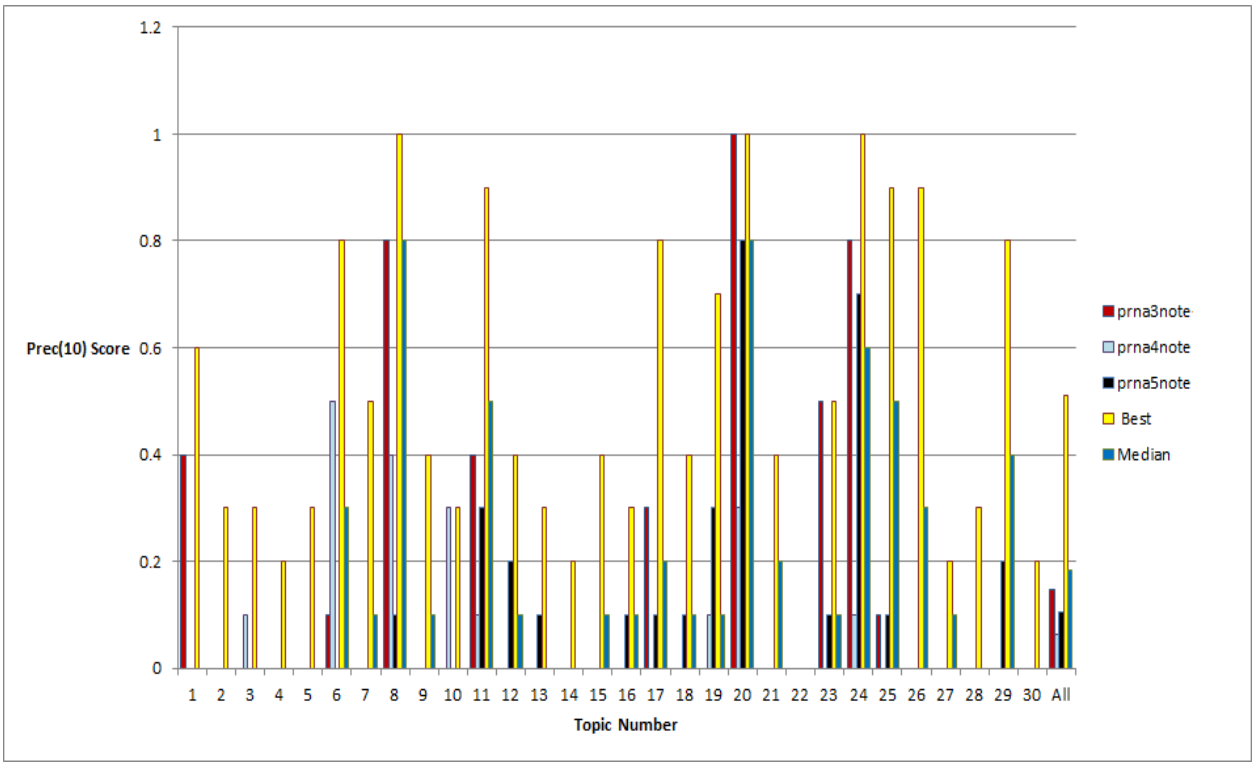

Figure 12: Prec(10) scores for each topic (comparison with participant runs using *notes* with inferencing similar to Hasan et al. (2015), knowledge graph, and key-value memory networks, respectively)

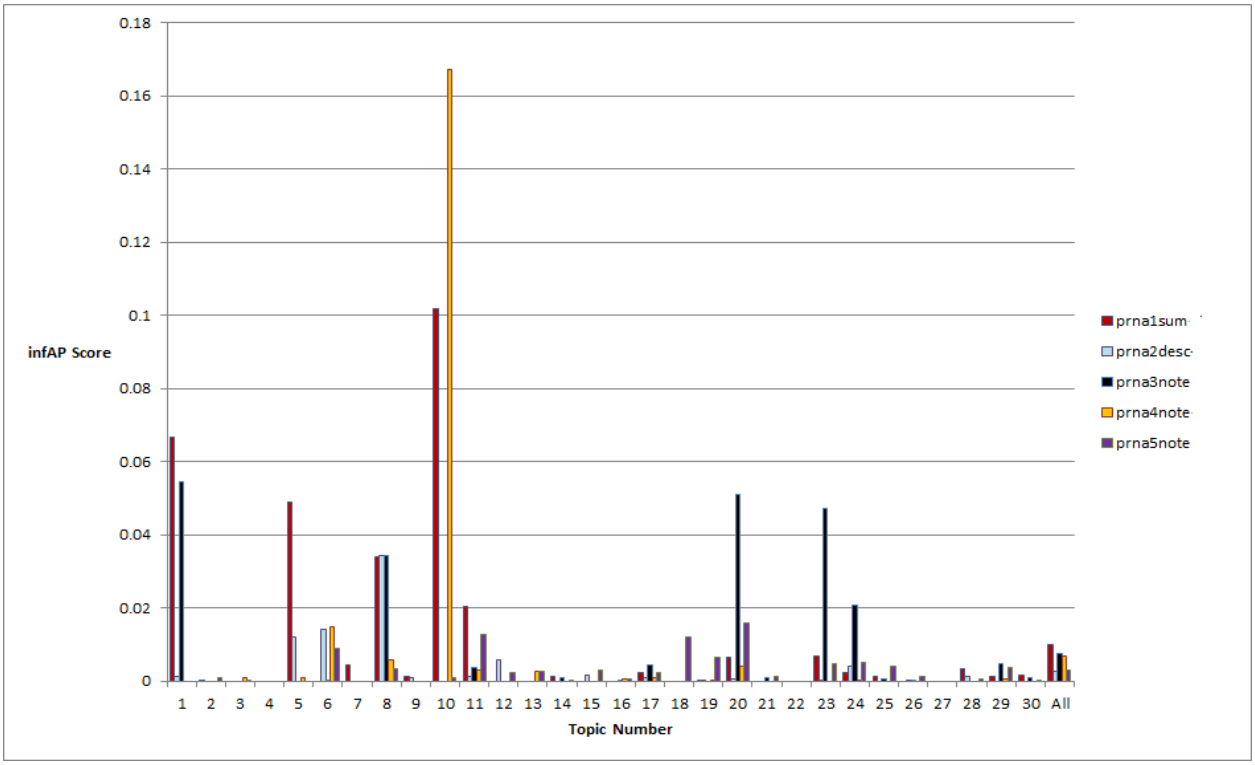

Figure 13: infAP scores for each topic (comparative analysis across five *prna* runs)

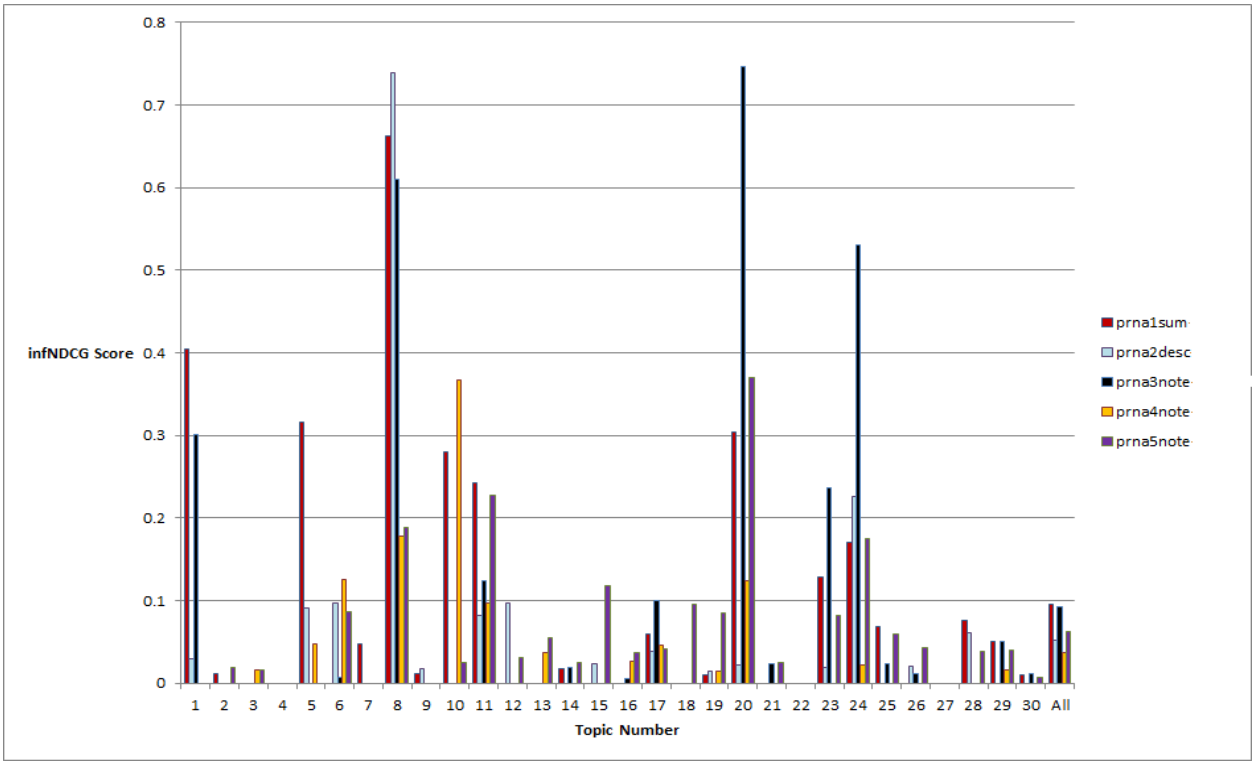

Figure 14: infNDCG scores for each topic (comparative analysis across five *prna* runs)

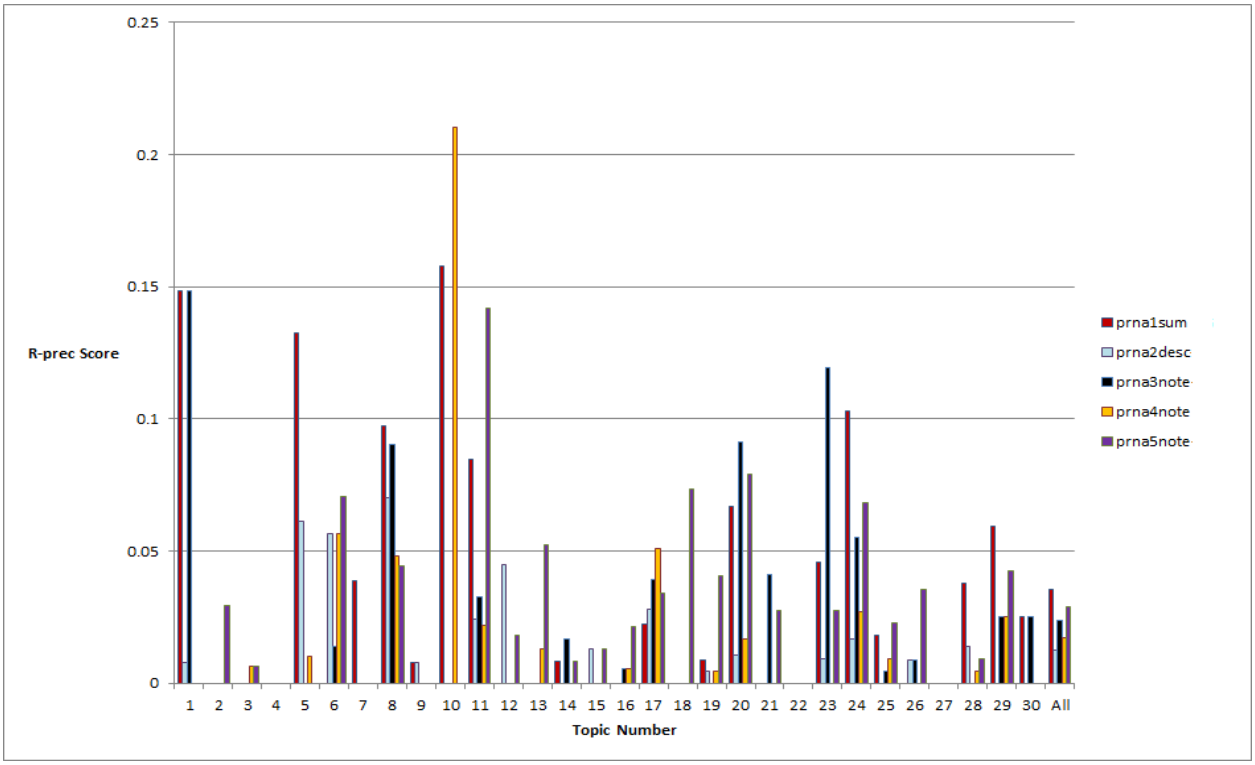

Figure 15: R-prec scores for each topic (comparative analysis across five *prna* runs)

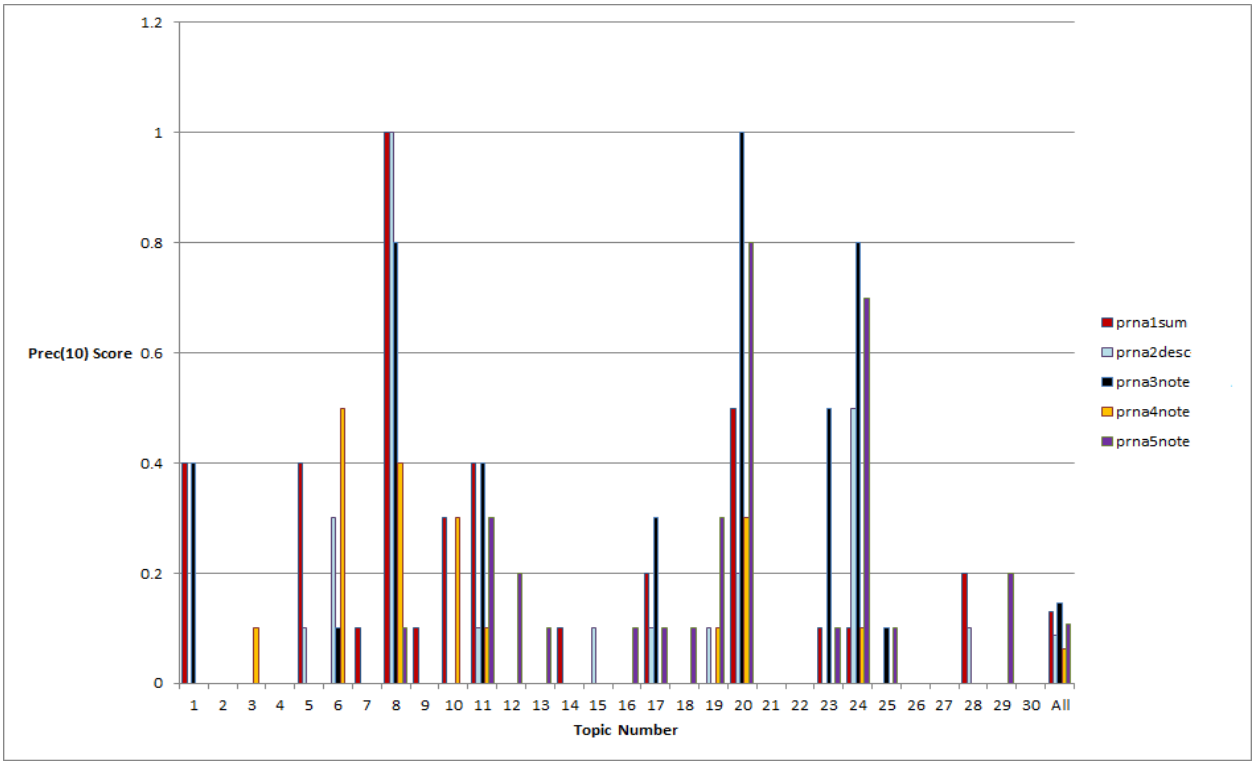

Figure 16: Prec(10) scores for each topic (comparative analysis across five *prna* runs)

chosen based on the model's performance on validation data. Finally, the learned model was used to predict the most probable diagnoses from the given medical notes for each topic.

3.4 Evaluation and Analysis

The evaluation of the CDS track was conducted using the standard TREC evaluation procedures for adhoc information retrieval tasks (Yilmaz et al., 2008; Voorhees, 2014). The highest ranked biomedical articles were sampled and judged by medical domain experts on a three-point scale of 0: not relevant, 1: possibly relevant, and 2: definitely relevant depending on the relevance of the answer to the associated question type about a given case report.

Figure 1 to Figure 4, Figure 5 to Figure 8, and Figure 9 to Figure 12 show the overall scores of our runs for topic *summaries*, topic *descriptions*, and topic *notes* respectively across all the topics as compared to the *median* and *best* scores for the submitted automatic runs for the following evaluation measures: inferred average precision[11] (infAP), inferred normalized discounted cumulative gain[12] (infNDCG), precision at R where R is the number of known relevant documents (R-prec), and precision at 10 documents (Prec (10)). The two inferred measures are used to provide more accurate estimates of a system's performance when relevance judgments are incomplete due to dynamic and/or larger document collections (Yilmaz and Aslam, 2006; Yilmaz et al., 2008). All the evaluation measures used for the CDS track contribute towards providing a comprehensive assessment of the quality of a system. Figure 13 to Figure 16 show the comparative results across our five submitted runs.

The reported results show that our clinical question answering system achieves *best* or close to *best* scores for 10% of the topics, better than *median* scores for ~17% of the topics, and close or equal to *median* scores for 10% of the topics across all evaluation measures when topic *summaries* are used with knowledge-graph based diagnostic inferencing. We can also see that our system obtains *best* or close to *best* scores for 3% of the topics, better than *median* scores for 10% of the topics, and close or equal to *median* scores for ~27% of the topics across all evaluation measures when topic *descriptions* are used with knowledge-graph based diagnostic inferencing. When using topic *notes*, our clinical question answering system with diagnostic inferencing similar to Hasan et al. (2015) performs better than the other two runs that use knowledge graph and KV-MemNN for the inferencing step.

Overall, our systems (all five runs) achieve best or close to best scores for 20% of the topics and better than median scores for 40% of the topics across all participants considering all evaluation measures. Furthermore, a detailed comparative analysis across our submitted runs demonstrates that on average our clinical question answering system performs best with *summaries* using diagnostic inferencing from the knowledge graph whereas our key-value memory network model with *notes* consistently outperforms the knowledge graph-based system for *notes* and *descriptions*.

4 Conclusion and Future Work

In this paper, we described our participation in the TREC 2016 CDS Track. Evaluation results showed additional gains with the use of a knowledge graph and a key-value memory network-based diagnostic inferencing approach for our clinical question answering system. These results further confirm the importance of accurate inferencing of diagnosis in retrieving relevant biomedical articles corresponding to underlying clinical narratives. In future, we plan to improve our clinical inferencing algorithms towards extracting the most accurate differential diagnoses by improving the performance of the memory network model besides leveraging larger collections of clinical knowledge sources.

[11] *Average Precision (AP)* is a measure that combines precision and recall for evaluating systems that retrieve a ranked list of articles. In particular, AP is the mean of the precision scores after each relevant article is retrieved.

[12] *Discounted Cumulative Gain (DCG)* measures the quality of ranking for a system when it retrieves a ranked list of results and the results are graded with relevance judgment. In particular, DCG computes the usefulness of an article based on its rank in the retrieved list. *Normalized DCG (NDCG)* is computed by using the maximum possible DCG (calculated by sorting the result list by relevance) as the normalization factor.

References

O. Bodenreider. 2008. Biomedical Ontologies in Action: Role in Knowledge Management, Data Integration and

Decision Support. *IMIA Yearbook of Medical Informatics*, 47(1):67–79.

S. Chandar, S. Ahn, H. Larochelle, P. Vincent, G. Tesauro, and Y. Bengio. 2016. Hierarchical Memory Networks. *arXiv preprint arXiv:1605.07427*.

R. Cornet and N. de Keizer. 2008. Forty Years of SNOMED: A Literature Review. *BMC Medical Informatics and Decision Making*, 8:1–7.

S. Garde, P. Knaup, E. Hovenga, and S. Heard. 2007. Towards Semantic Interoperability for Electronic Health Records. *Methods of Information in Medicine*, 46(3):332–343.

S. A. Hasan, X. Zhu, Y. Dong, J. Liu, and O. Farri. 2014. A Hybrid Approach to Clinical Question Answering. In *Proceedings of the Twenty-Third Text REtrieval Conference, TREC 2014, Gaithersburg, Maryland, USA, November 19-21, 2014*.

S. A. Hasan, Y. Ling, J. Liu, and O. Farri. 2015. Using Neural Embeddings for Diagnostic Inferencing in Clinical Question Answering. In *Proceedings of the Twenty-Fourth Text REtrieval Conference, TREC 2015, Gaithersburg, Maryland, USA, November 17-20, 2015*.

S. Hochreiter and J. Schmidhuber. 1997. Long Short-Term Memory. *In Neural Computation*, 9(8):1735–1780.

A. Johnson, T. J. Pollard, L. Shen, L. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, and R. G. Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3.

D. P. Kingma and J. Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980.

A. Miller, A. Fisch, J. Dodge, A. Karimi, A. Bordes, and J. Weston. 2016. Key-Value Memory Networks for Directly Reading Documents. *CoRR*, abs/1606.03126.

M. Rospocher, M. Erp, P. Vossen, A. Fokkens, I. Aldabe, G. Rigau, A. Soroa, T. Ploeger, and T. Bogaard. 2016. Building Event-centric Knowledge Graphs from News. *Journal of Web Semantics*, 37:132–151.

M. Saeed, M. Villarroel, A. T. Reisner, G. Clifford, L. Lehman, G. Moody, T. Heldt, T. H. Kyaw, B. Moody, and R. G. Mark. 2011. Multiparameter intelligent monitoring in intensive care ii MIMIC-II: A public-access intensive care unit database. *Critical Care Medicine*, 39:952–960.

N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

H. Stenzhorn, S. Schulz, M. Boeker, and B. Smith. 2008. Adapting Clinical Ontologies in Real-World Environments. *Journal of Universal Computer Science*, 14(22):3767–3780.

S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus. 2015. End-To-End Memory Networks. In *Advances in Neural Information Processing Systems*, pages 2440–2448.

Lyndal Trevena. 2011. Wikiproject medicine. *BMJ*, 342:d3387.

E. M. Voorhees. 2014. The Effect of Sampling Strategy on Inferred Measures. In *Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '14, pages 1119–1122.

J. Weston, S. Chopra, and A. Bordes. 2014. Memory Networks. *CoRR*, abs/1410.3916.

E. Yilmaz and J. A. Aslam. 2006. Estimating Average Precision with Incomplete and Imperfect Judgments. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, CIKM '06, pages 102–111.

E. Yilmaz, E. Kanoulas, and J. A. Aslam. 2008. A Simple and Efficient Sampling Method for Estimating AP and NDCG. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, pages 603–610.