Identifying Supporting Facts for Multi-hop Question Answering with Document Graph Networks

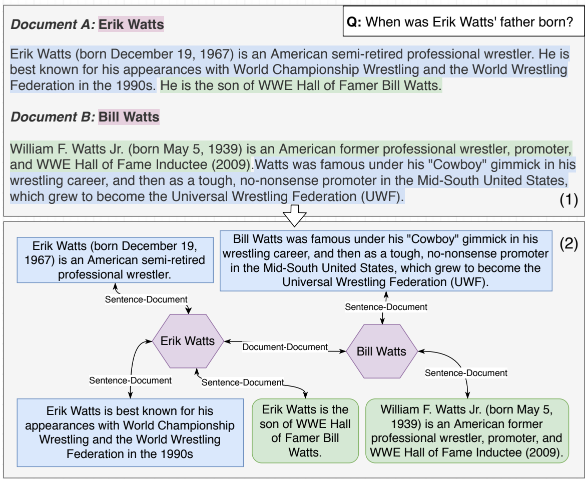
Mokanarangan Thayaparan[*], **Marco Valentino**[*], **Viktor Schlegel**[*], **André Freitas**
Department of Computer Science
University of Manchester
{thayaparan.mokanarangan,marco.valentino,viktor.schlegel,andre.freitas}
@manchester.ac.uk

Abstract

Recent advances in reading comprehension have resulted in models that surpass human performance when the answer is contained in a single, continuous passage of text. However, complex Question Answering (QA) typically requires multi-hop reasoning – i.e. the integration of supporting facts from different sources, to infer the correct answer.

This paper proposes Document Graph Network (DGN), a message passing architecture for the identification of supporting facts over a graph-structured representation of text.

The evaluation on HotpotQA shows that DGN obtains competitive results when compared to a reading comprehension baseline operating on raw text, confirming the relevance of structured representations for supporting multi-hop reasoning.



Document A: **Erik Watts**

Q: When was Erik Watts' father born?

Erik Watts (born December 19, 1967) is an American semi-retired professional wrestler. He is best known for his appearances with World Championship Wrestling and the World Wrestling Federation in the 1990s. He is the son of WWE Hall of Famer Bill Watts.

Document B: **Bill Watts**

William F. Watts Jr. (born May 5, 1939) is an American former professional wrestler, promoter, and WWE Hall of Fame Inductee (2009).Watts was famous under his "Cowboy" gimmick in his wrestling career, and then as a tough, no-nonsense promoter in the Mid-South United States, which grew to become the Universal Wrestling Federation (UWF).

(1)

Erik Watts (born December 19, 1967) is an American semi-retired professional wrestler.

Bill Watts was famous under his "Cowboy" gimmick in his wrestling career, and then as a tough, no-nonsense promoter in the Mid-South United States, which grew to become the Universal Wrestling Federation (UWF).

Erik Watts is best known for his appearances with World Championship Wrestling and the World Wrestling Federation in the 1990s

Erik Watts is the son of WWE Hall of Famer Bill Watts.

William F. Watts Jr. (born May 5, 1939) is an American former professional wrestler, promoter, and WWE Hall of Fame Inductee (2009).

Figure 1: Is structure important for complex, multi-hop Question Answering (QA) over unstructured text passages? To answer this question we explore the task of identifying supporting facts (rounded rectangles) by transforming a corpus of documents (1) into an undirected graph (2) connecting sentence nodes (rectangles) and document nodes (hexagons).

1 Introduction

Question Answering (QA) is the task of inferring the answer for a natural language question in a given knowledge source. Acknowledged as a suitable task for benchmarking natural language understanding, QA is gradually evolving from mere retrieval task to a well-established tool for testing complex forms of reasoning. Recent advances in deep learning have sparked interest in a specific type of QA emphasising Machine Comprehension (MC) aspects, where background knowledge is entirely expressed in form of unstructured text.

State-of-the-art techniques for MC typically retrieve the answer from a continuous passage of text by adopting a combination of character and word-level models with various forms of attention mechanisms (Seo et al., 2016; Yu et al., 2018). By employing unsupervised pre-training on large corpora (Devlin et al., 2018), these models are capable of outperforming humans in reading comprehension tasks where the context is represented by a single paragraph (Rajpurkar et al., 2018).

However, when it comes to answering complex questions on large document collections, it is unlikely that a single passage can provide sufficient evidence to support the answer. Complex QA typically requires multi-hop reasoning, i.e. the ability of combining multiple information fragments from different sources.

Moreover, recent studies have raised concerns on inference capabilities, generalisation and interpretability of current MC models (Wiese et al., 2017; Dhingra et al., 2017; Kaushik and Lipton, 2018), leading to the creation of novel datasets that propose multi-hop reading comprehension as a benchmark for evaluating complex reasoning and explainability (Yang et al., 2018).

[*]: equal contribution

Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-13), pages 42–51
Hong Kong, November 4, 2019. ©2019 Association for Computational Linguistics

Consider the example in Figure 1. In order to answer the question *"When was Erik Watts' father born?"* a QA system has to retrieve and combine supporting facts stored in different documents:

1. Document A: *"Erik Watts is the son of WWE Hall of Famer Bill Watts"*;
2. Document B: *"William F. Watts Jr. (born May 5, 1939) is an American former professional wrestler, promoter, and WWE Hall of Fame Inductee (2009)"*.

The explicit selection of supporting facts has a dual role in a multi-hop QA pipeline:

(a) It allows the system to consider all and only those facts that are relevant to answer a specific question;

(b) It provides an explicit trace of the reasoning process, which can be presented as justification for the answer.

This paper explores the task of identifying supporting facts for multi-hop QA over large collections of documents where several passages act as distractors for the MC model. In this setting, we hypothesise that graph-structured representations play a key role in reducing complexity and improving the ability to retrieve meaningful evidence for the answer.

As shown in Figure 1.1, identifying supporting facts in unstructured text is challenging as it requires capturing long-term dependencies to exclude irrelevant passages. On the other hand (Figure 1.2), a graph-structured representation connecting related documents simplifies the integration of relevant facts by making them mutually reachable in few hops. We put this observation in practice by transforming a text corpus in a global representation that links documents and sentences by means of mutual references.

In order to identify supporting facts on undirected graphs, we investigate the use of message passing architectures with relational inductive bias (Battaglia et al., 2018). We present the Document Graph Network (DGN), a specific type of Gated Graph Neural Network (GGNN) (Li et al., 2015) trained to identify supporting facts in the aforementioned structured representation.

We evaluate DGN on HotpotQA (Yang et al., 2018), a recently proposed dataset for assessing MC performance on supporting facts identification. The experiments show that DGN is able to obtain improvements in F1 score when compared to a MC baseline that adopts a sequential reading strategy. The obtained results confirm the value of pursuing research towards the definition of novel MC architectures, which are able to incorporate structure as an integral part of their learning and inference processes.

2 Document Graph Network

The following section presents the Document Graph Network (DGN), a message passing architecture designed to identify supporting facts for multi-hop QA on graph-structured representations of documents.

Here, we discuss in details the construction of the underlying graph, the DGN model, and a pre-filtering step implemented to alleviate the impact of large graphs on model complexity.

2.1 Graph-structured Representation

Given an arbitrary corpus of documents $\mathcal{D} = \{D_1, D_2, \ldots, D_n\}$, we aim to build an undirected document graph DG as structured representation of $\mathcal{D}$ (Figure 1).

The advantage of using graph-structured representations lies in reducing the inference steps necessary to combine two or more supporting facts. Therefore, we want to extract a representation that increases the probability of connecting relevant sentences with short paths in the graph. We observe that multi-hop questions usually require reasoning on two concepts/entities that are described in different, but interlinked documents. We put in practice this observation by connecting two documents if they contain mentions to the same entities.

The Document Graph (DG) contains nodes of two types. We represent each document D_i in $\mathcal{D}$ as a document node d_i and each of its sentences S_{jD_i} as a sentence node s_{jD_i}. We then add an edge of type $e_{sentence-document}$ that links them. This edge type represents the fact that a specific sentence belongs to a specific document. We apply coreference resolution to solve implicit entity mentions within the documents. Subsequently, we add an edge of type $e_{document-document}$ between two document nodes d_1, d_2, if the entities described in D_1 are referenced in D_2 or vice-versa.

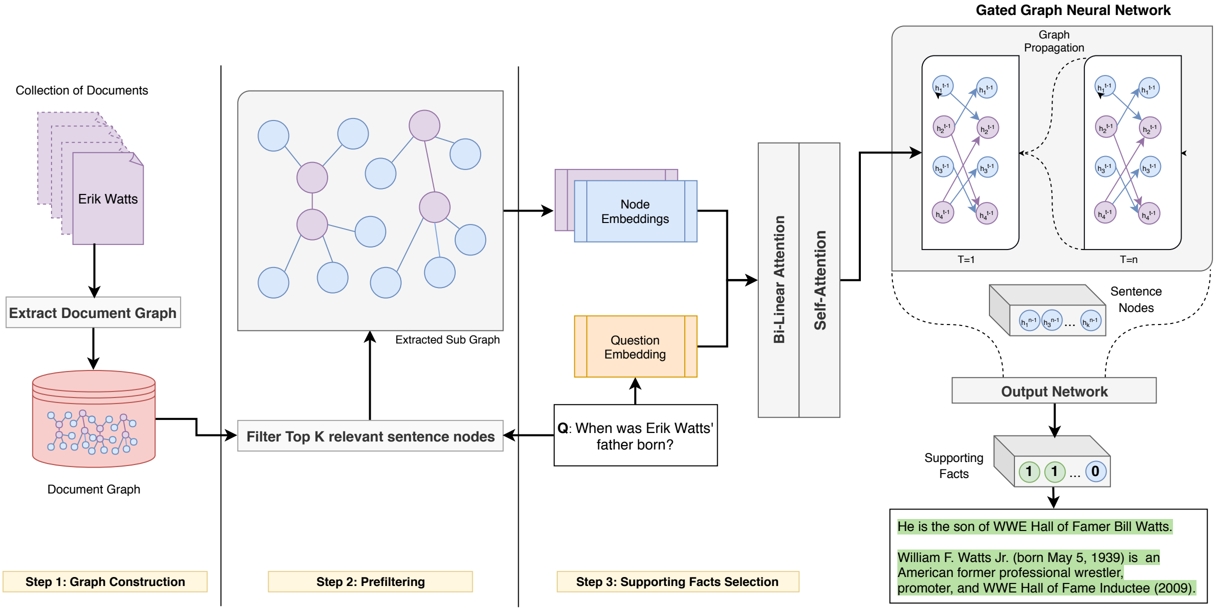

Collection of Documents
Erik Watts
Extract Document Graph
Document Graph
Step 1: Graph Construction
Extracted Sub Graph
Filter Top K relevant sentence nodes
Step 2: Prefiltering
Node Embeddings
Question Embedding
Q: When was Erik Watts' father born?
Bi-Linear Attention
Self-Attention
Step 3: Supporting Facts Selection
Gated Graph Neural Network
Graph Propagation
h_1^{t-1}
h_2^{t-1}
h_3^{t-1}
h_4^{t-1}
T=1
T=n
Sentence Nodes
h_1^{n-1} h_3^{n-1} ... h_k^{n-1}
Output Network
Supporting Facts
1 1 ... 0
He is the son of WWE Hall of Famer Bill Watts.
William F. Watts Jr. (born May 5, 1939) is an American former professional wrestler, promoter, and WWE Hall of Fame Inductee (2009).

Figure 2: Overview of the approach for the identification of supporting facts in a multi-hop QA pipeline. Step 1 is applied offline for extracting a graph-structured representation from large corpora (Sec. 2.1). In Step 2, we employ a filtering algorithm (Sec. 2.3) to retrieve a sub-graph containing the top k relevant sentences nodes. The final step (Step 3) adopts the DGN model for message passing and binary classification of the supporting facts (Sec. 2.4).

Given a question q, we use DG (instead of $\mathcal{D}$) as input for the DGN model. the representation does not include edges between sentences since we observed increasing complexity in the model without gaining substantial benefits in terms of performance.

2.2 Architectural Overview

Figure 2 highlights the main components of the DGN architecture.

From the target corpus, we automatically extract a Document Graph DG encoding the background knowledge expressed in a corpus of documents (Step 1). This data and its graphical structure is permanently stored into a database, ready to be loaded when it is required. The first step is performed offline, allowing the integration of new knowledge regardless of the runtime pipeline implemented to address the task.

In order to speed up the computation and alleviate current drawbacks of Gated Graph Neural Networks (Li et al., 2015), the question answering pipeline is augmented with a prefiltering step (Step 2). The adopted algorithm (Sec 2.3), based on a relevance score, is aimed at reducing the number of nodes involved in the computation. Current limitations of Gated Graph Neural Networks, in fact, are mainly connected with the size of the input graph used for learning and prediction. Performance in terms of computational efficiency and learning degrades proportionally to the number of nodes and edge types in the input graph. In order to reduce the negative impact of large graphs, we adopt the prefiltering step to prune DG, and retrieve a set of sentence nodes $\mathcal{S} = \{S_1, S_2, \ldots, S_k\}$ expected to contain supporting facts for a question q.

The subsequent step (Step 3) is aimed at selecting supporting facts for q. For this task we employ the Document Graph Network (DGN) on the subset of DG induced by $\mathcal{S}$ (section 2.4). Specifically, we apply the aforementioned architecture to learn a distributed representation of each node in the graph via message passing. This representation is then used by an Output Network (ON) to perform binary classification on the sentence nodes in $\mathcal{S}$ and select a set of supporting facts $SF = \{sf_1, sf_2, \ldots, sf_m\}$ with $SF \subseteq \mathcal{S}$. In the experiments we perform supervised learning on the training set provided by HotpotQA (Yang et al., 2018) to correctly predict the elements belonging to SF.

2.3 Prefiltering Step

Given a question q and a set of documents $D = \{D_1, D_2, \ldots, D_n\}$ as context, the aim of the prefiltering step is to retrieve a subset of the context containing the k most relevant sentences to q.

In order to achieve this goal, we adopt a ranking based approach similar to the one illustrated in (Narasimhan et al., 2018). Specifically, we consider all the sentences occurring in the documents and compute the similarity between each word in a sentence and each word in the question q. We adopt pre-trained GloVe vectors (Pennington et al., 2014) to obtain the distributed representation of each word. Subsequently, we produce the relevance score of each sentence by calculating the mean among the m highest similarity values. The final subset is obtained by selecting the sentences with the top k relevance scores.

An empirical analysis suggested that $m = 5$ gives the best results on the development set. We evaluated this approach by computing the recall of retrieving the top k supporting facts for $k = \{20, 25, 30\}$, obtaining values greater than 90% for $k = 25$ and $k = 30$. Since the average number of candidate sentences for each question in the corpus is 50.89, the described algorithm allows us to discard 60.7$\%$ ($k = 20$), 50.87$\%$ ($k = 25$) and 41.05$\%$ ($k = 30$) of irrelevant context.

2.4 Identifying Supporting Facts

The Document Graph Network (DGN) is employed for the identification of supporting facts. The DGN model is based on a standard Gated Graph Neural Network architecture (GGNN) (Li et al., 2015) where the inner representation of the nodes is customised to carry out this specific task. We apply DGN on the sub-graph retrieved by the filtering module.

In alignment with prior research in the field we encode Question(Q), Nodes(N) and Graph(G) as follows:

1. **Question Representation**: The question is stripped of punctuation and stop words and tokenised to obtain W words. These words are subsequently converted into a tensor $Q \in R^{|W| \times D}$ using pre-trained GloVe vectors (Pennington et al., 2014) of dimension D.

2. **Node Representation**: Similar to the question representation, each node is also converted to $V \in R^{|W| \times D}$ using entities for document nodes and sentences words for sentence nodes.

3. **Graph Representation**: Each document graph DG is represented by an adjacency matrix $A \in R^{|V| \times 2|E||V|}$ where V and E denote the vertices and edge types respectively.

Each node (v_i) is conditioned on the question (q_i) using Bi-Linear Attention (Kim et al., 2018). The attention weights α_i of each word w in the nodes are determined by a learned function f_{BAN} as shown in Equation 2. Here f_{BAN} computes the attention scores between two matrices using a bilinear attention function. This function has a matrix of weights W and a bias vector b used to calculate the similarity between the two matrices as $VWQ^T + b$:

$$e_{iw} = f_{BAN}(v_{iw}, q_i) \tag{1}$$

$$\alpha_{iw} = \frac{exp(e_{iw})}{\sum_{k=1}^{|W|} exp(e_{ik})} \tag{2}$$

Following the calculation of the attention scores, the question conditioned vectors are determined as follows:

$$\hat{v}_i = \phi(\{v_i\}, \{\alpha_i\}) \tag{3}$$

Here, ϕ is a learned function that combines the attention scores of each word by employing a non-linear transformation.

After conditioning the nodes representation on the question, we employ a Self-Attention Model function f_{SAN} (Vaswani et al., 2017) to calculate the weight of each vector δ_i. Here, the learned function f_{SAN} is responsible for computing the weights of each vector in a node. The rationale behind this operation is to condense the matrices to a vector suitable for a Gated Graph Neural Network architecture while retaining the most discriminative semantic information.

$$r_{iw} = f_{SAN}(\hat{v}_{iw}) \tag{4}$$

$$\delta_{iw} = \frac{exp(r_{iw})}{\sum_{k=1}^{|W|} exp(r_{ik})} \tag{5}$$

After computing the self-attention score, we calculate the initial annotation vectors for the GGNN as follows:

$$x_v = \sigma(\{\hat{v}_i\}, \{\delta_i\}) \tag{6}$$

where σ is a function that returns a single vector by multiplying the corresponding attention scores and summing them up. The basic recurrent unit of a GGNN can be formalised as follows:

$$h_v^{(1)} = [x_v^T, 0]^T \tag{7}$$

$$a_v^{(t)} = A_{v:}^T [h_1^{(t-1)T} ... h_{|V|}^{(t-1)T}]^T + b \tag{8}$$

$$z_v^t = \sigma(W^z a_v^{(t)} + U^z h_v^{(t-1)}) \tag{9}$$

$$r_v^t = \sigma(W^r a_v^{(t)} + U^r h_v^{(t-1)}) \tag{10}$$

$$\widetilde{h_v^t} = tanh(W a_v^{(t)} + U(r_v^t \odot h_v^{(t-1)})) \tag{11}$$

$$h_v^{(t)} = (1 - z_v^t) \odot h_v^{(t-1)} + z_v^t \odot \widetilde{h_v^{(t)}} \tag{12}$$

We perform T time steps of propagation and retrieve the distributed nodes representation by using the final hidden state. The computed representation of each node implicitly captures the semantic information of its neighbours at a distance up to T hops. In the experiments, we found it sufficient to set $T = 3$.

The graph is heterogeneous with nodes representing questions, sentences and documents. As the supporting facts identification task requires sentence classification, we retain the final hidden state of the sentence nodes while discarding the others. We use the sentence representations as input to a feed forward neural network called Output Network. We perform binary classification of each sentence to predict whether it is a supporting fact or not:

$$o_v = g(h_v^{(T)}, x_v) \tag{13}$$

3 Evaluation

The experiments are motivated by the guiding research question of the paper: *Does structure play a role in identifying supporting facts for multi-hop Question Answering?* We further break down the question in the following research hypotheses:

- **RH1:** Existing machine comprehension models benefit from reducing the context to a small number of sentences necessary to answer a question.

- **RH2:** Models operating on a graph-structured representation perform better, supporting the identification of relevant facts when compared to a baseline that uses a sequential strategy.

We seek to provide evidence for those claims by conducting the following experiments:

- **Experiment 1:** investigate how a representative state-of-the-art MC model performs on different passages with varying coherency and length.

- **Experiment 2:** evaluate the capability of the proposed approach to identify supporting facts in a question answering scenario where the relevant facts are distributed across multiple documents.

Specific tests are performed to identify contributing features and compare the overall performance of the approach with a sequential baseline reported in the literature.

HotpotQA We ran the experiments over the recently proposed HotpotQA dataset (Yang et al., 2018), which requires MC models to find supporting passages in a large set of documents, and perform multi-hop reasoning to arrive at the correct answer. HotpotQA provides 105,547 first paragraphs extracted from Wikipedia articles, and corresponding question-answer pairs created by human annotators. Questions are designed to only be answerable by combining information from two articles and require to bridge documents via a concept or entity mentioned in both articles. A subset of questions require a comparison of similar concepts concerning their common or differing properties. Furthermore, the dataset provides labels for supporting sentences, making it possible to perform quantitative analysis on the retrieval of supporting facts.

In all of the reported experiments, if not stated otherwise, training is performed on the HotpotQA *training set* while the evaluation is performed on the *development set* in the *distractor setting*. In order to address this setting, a system has to retrieve the answer and the supporting facts for a given question by reasoning over a set of ten documents. Only two of the supplied documents are guaranteed to contain the information that is sufficient and necessary to answer the question. The remaining eight documents are similar documents retrieved by an information retrieval model (hence the name *distractor*).

3.1 State-of-the-Art Machine Comprehension Performance

This experiment is designed to investigate the capabilities of single passage MC models to retrieve the correct answer when provided with a context of varying size and coherency. For this analysis we adopt BERT (Devlin et al., 2018), a neural transformer architecture (Vaswani et al., 2017) constituting the state-of-the-art latent representation for various NLP tasks.

The publicly available model is pre-trained in an unsupervised manner on a large text corpus with the objective of language modelling and next sentence prediction. Fine-tuning this model to specific NLP tasks has shown to achieve state-of-the-art-results for many NLP tasks, among others question answering and machine reading comprehension (Devlin et al., 2018). To that end, we fine-tune the model on the training split of HotpotQA and evaluate it on the evaluation split. Before training, we manually remove all the questions that cannot be answered by retrieving a continuous passage in the supporting facts (e.g. we exclude comparison questions that typically require yes/no type of answers).

We evaluate the performance of BERT with supporting facts only, and then progressively enrich the context by a rising number of sentences retrieved by the filtering algorithm (Sec. 2.3). The results of this experiment are reported in Table 1.

Note that these results can not be interpreted as a resilient comparison baseline as *(1)* we don't optimise the set of hyper-parameters associated with the model training and *(2)* we ensure the existence of supporting facts in the evaluation, since we are interested in the intrinsic performance of BERT in answer retrieval.

Unsurprisingly, the best results are achieved when the context provided to BERT is composed of supporting facts only. Conversely, the performance of the model gradually deteriorates when distracting information is added to the context.

These results reinforce our assumption that a module capable of identifying the correct set of supporting facts represents a fundamental component in a multi-hop QA pipeline. Moreover, this component may be complementary to downstream machine comprehension models, constituting a valid support to improve overall performances in answer retrieval.

3.2 Supporting Facts Identification

We compare the DGN model on the task of identifying supporting facts against the neural baseline reported in (Yang et al., 2018). In order to suit the task, the baseline architecture extends the state-of-the-art answer passage retrieval model (Seo et al., 2016) by an additional recurrent layer that classifies whether a sentence is a supporting fact or not. The model is trained jointly and under strong supervision on the objectives of retrieving both answer and supporting facts. We replicate the experiment on our infrastructure in order to obtain more detailed measures, such as precision an recall. The results of the evaluation are reported in Table 2.

Table 1: F1 and exact match (EM) score of BERT when evaluated on answer retrieval over contexts of varying size.

Sentences	SF only	+5	+10	+20	+25	+30
F1	**0.75**	0.60	0.52	0.44	0.42	0.42
EM	**0.60**	0.47	0.40	0.33	0.32	0.31

Table 2: Supporting facts identification: Harmonic mean (**F1**), **P**recision and **R**ecall

Model	F1	P	R
Baseline (Yang et al., 2018)	66.66	-	-
Baseline Replication (Answer + SF)	65.28	**63.28**	67.43
Baseline Replication (SF only)	46.44	48.80	44.31
DGN (best)	**68.02**	61.51	**76.07**
DGN (no edge types)	45.84	-	-

The experiments show that the DGN model outperforms the baseline in terms of F1 score (≈2% improvement compared to the results reported in the paper, ≈3% improvement compared to our replication), and recall (≈14% improvement over our replication). However, the baseline implementation has a higher precision. We attribute that to the fact that the baseline optimises for both answer extraction and supporting facts retrieval.

In general we observe that recall is higher than precision throughout the experiments. Compared to the DGN model, the baseline is less penalised when the retrieved answer still matches the expected answer, even if retrieved from an unrelated sentence spuriously. in the absence of the answer selection optimisation criterion, the DGN model is only penalised if it fails to predict the correct supporting facts. This forces the model to prioritise recall over precision during training. Adding a weight to the loss calculation as an additional hyperparameter can balance the precision and recall metric.

We don't evaluate DGN on the task of answer retrieval, since the proposed architecture focuses on the classification of the relevant supporting facts. The task of jointly retrieving answer and supporting facts is left for future work.

Table 3: **P**recision, **R**ecall and **F1** score of the DGN model with different values of k assigned to the prefiltering algorithm. The best values are highlighted.

	k = 20	**k = 25**	**k = 30**	**full**
Prefiltering (R)	84.72	90.23	94.29	100.00
DGN (F1)	63.40	67.38	**68.02**	66.00
DGN (P)	57.83	**61.92**	61.51	53.37
DGN (R)	70.17	73.90	76.07	**86.47**

3.3 Analysis

In order to understand the interaction of the key contributing parts of the architecture, we analyse the behaviour of the full pipeline in different settings. Specifically, we measure the DGN performance when trained and evaluated on the output of the filtering step. During the training, we ensure the existence of the supporting facts in the input graph of the DGN model. We then evaluate it on the development set by performing prediction on the subset retrieved by the filtering algorithm. The results reported in Table 3 take into account the combined performance of the full pipeline with different hyperparameters assigned to the prefiltering algorithm.

Firstly, we observe the increase of recall with the increasing number of retrieved sentences. This fact is unsurprising and it is in line with the higher recall score of the filtering module. More sentences means broader coverage, and thus higher recall even before executing DGN prediction.

Secondly, across the experiments, we observe that $k = 30$ is the best number of sentences for the model to learn from. This is confirmed by the best precision and overall F1 score obtained when training and predicting on the top 30 sentences. Moreover, we observed that the application of the filtering algorithm sensibly speeds up the training, decreasing at the same time the amount of memory required to store matrices and weights of the graph network. The application of a light filtering is then justified both in terms of performance and computational complexity.

Regarding the baseline model, we aim to analyse the impact of multi-task learning, where the model is jointly trained to retrieve supporting facts and the final answer. We observe a significant drop in performance (≈20% F1 score) when we optimise the baseline only for supporting facts identification (see *Baseline Replication* in Table 2). This observation is perfectly in line with the literature. (Hashimoto et al., 2016) report improvements on low level tasks when jointly optimised with higher level tasks in a hierarchical learning setting. Regarding multi-hop QA, the identification of supporting facts directly depends on the answer being predicted correctly and vice-versa. A plausible future work may be to understand whether DGN can benefit from a similar multi-task learning setup.

Finally, we investigate the role of the semantic information expressed explicitly in the Document Graph. To that end, we train the DGN model using the same configuration of the best performing model without edge type information. This results in a notable drop of F1 score (see Table 2) reinforcing the evidence that explicit semantic information encoded in relational form contributes towards the performance of the model. A promising future direction will be to investigate whether different types of semantic representation benefit the performance of the model and to what extent.

4 Related Work

State-of-the-art approaches for Open-Domain Question Answering over large collections of documents employ a combination of character-level models, self-attention (Wang et al., 2017), and bi-attention (Seo et al., 2016) to operate over unstructured paragraphs without exploiting any structured text representation. Despite these methods have demonstrated impressive results reaching in some cases super-human performances (Seo et al., 2016; Chen et al., 2017; Yu et al., 2018), recent studies have raised important concerns related to generalisation (Wiese et al., 2017; Dhingra et al., 2017) complex reasoning (Welbl et al., 2018) and explainability (Yang et al., 2018). Specifically, the lack of structured representation makes it hard for current Machine Comprehension models to find meaningful patterns in large corpora, generalise beyond the training domain and justify the answer.

Research efforts towards the creation of message-passing architectures with relational inductive bias (Battaglia et al., 2018) have enabled machine learning algorithms to incorporate graphical structures in their training process. These models, trained over explicit entities and relations, have the potential to boost generalisation, interpretability and abstract reasoning capabilities. A variety of Graph Neural Network architectures have already demonstrated remarkable results in a large set of applications ranging from Computer Vision, Physical Systems and Protein-Protein Interaction (Zhou et al., 2018).

Our research is in line with recent trends in Question Answering prone to explore message-passing architectures over graph-structured representation of documents to enhance performance and overcome challenges involved in dealing with unstructured text. (Sun et al., 2018) fuse text corpus with manually-curated knowledge bases to create heterogeneous graphs of KB facts and text sentences. Their model, GRAFT-Net, built upon Graph Convolutional Networks (Schlichtkrull et al., 2018), is used to propagate information between heterogeneous nodes in the graph and perform binary classification on entity nodes to select the answer. Differently from the proposed approach, the latter work focuses on links between whole paragraphs and external entities in a Knowledge Base. Moreover, GRAFT-Net is designed for single-hop Question Answering, assuming that the question is always about a single entity.

The proposed approach is similar to (De Cao et al., 2018) and (Song et al., 2018), where the aim is to answer complex questions that require the integration of multiple text passages. However, our research is focused on the identification of supporting facts instead of answer retrieval.

Another line of research focuses on narrowing down the context for later Machine Comprehension models by selecting relevant passages as supporting facts. Work in that direction includes (Watanabe et al., 2017) which present a neural information retrieval system to retrieve a sufficiently small paragraph and (Geva and Berant, 2018) which employ a Deep Q-Network (DQN) to solve the task by learning to navigate over an intra-document tree. A similar approach is chosen by (Clark and Gardner, 2017). However, instead of operating on document structure, they adopt a sampling technique to make the model more robust towards multi-paragraph documents. These approaches are not directly comparable to our work since they focus either on single paragraphs or intra-document (local) structure.

Strongly related to our work is (Yang et al., 2018) which presents HotpotQA, a novel dataset for multi-hop QA. The authors highlight the importance of identifying supporting facts for improving reasoning and explainability of current systems. We compare the proposed architecture with the baseline described in their paper. The model is based on a state-of-the-art MC model (Seo et al., 2016) that adopts a sequential reading strategy to identifying supporting facts from large collections of documents.

5 Conclusion

In this paper, we investigated the role played by interlinked sentence representation for complex, multi-hop question answering under the focus of supporting facts identification, i.e. retrieving the minimum set of facts required to answer a given question. We emphasise that this problem is worth pursuing, showing that the performance of state-of-the-art models substantially deteriorates as the size of the accompanying context increases.

We present Document Graph Network (DGN), a novel approach for selecting supporting facts in a multi-hop QA pipeline. The model operates over explicit relational knowledge, connecting documents and sentences extracted from large text corpora. We adopt a pre-filtering step to limit the number of nodes and train a customised Graph Gated Neural Network directly on the extracted representation.

We train and evaluate the DGN model on a newly proposed dataset for complex, multi-hop question answering over unstructured text. The evaluation shows that DGN outperforms a baseline adopting a sequential reading strategy. Additionally, we show that when trained to retrieve just supporting facts, the performance of the baseline degrades by $\approx$20%.

Perhaps most importantly, we highlight a way to combine structured and distributional sentence representation models and propose further research lines in that direction. As future work, we aim to investigate the role and impact of different structured sentence representation models within the inference process, linking it with the Open Information Extraction (Cetto et al., 2018; Niklaus et al., 2018) and sentence simplification (Niklaus et al., 2019, 2017) literature.

We believe that further research can be dedicated to inject richer structured knowledge in the model, allowing for fine-grained message passing and improved representation learning. Another important line of research will focus on the implementation of advanced mechanisms and techniques to scale the approach to massive text corpora such as the whole Wikipedia.

Acknowledgements

The authors would like to express their gratitude towards members of the AI Systems lab at the University of Manchester for many fruitful and intense discussions.

References

Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. 2018. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.

Matthias Cetto, Christina Niklaus, André Freitas, and Siegfried Handschuh. 2018. Graphene: Semantically-linked propositions in open information extraction. *Prooceedings of COLING*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Christopher Clark and Matt Gardner. 2017. Simple and effective multi-paragraph reading comprehension. *arXiv preprint arXiv:1710.10723*.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2018. Question answering by reasoning across documents with graph convolutional networks. *arXiv preprint arXiv:1808.09920*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Bhuwan Dhingra, Kathryn Mazaitis, and William W Cohen. 2017. Quasar: Datasets for question answering by search and reading. *arXiv preprint arXiv:1707.03904*.

Mor Geva and Jonathan Berant. 2018. Learning to search in long documents using document structure. *arXiv preprint arXiv:1806.03529*.

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2016. A joint many-task model: Growing a neural network for multiple nlp tasks. *arXiv preprint arXiv:1611.01587*.

Divyansh Kaushik and Zachary C Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. *arXiv preprint arXiv:1808.04926*.

Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. 2018. Bilinear attention networks. In *Advances in Neural Information Processing Systems*, pages 1571–1581.

Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. 2015. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*.

Medhini Narasimhan, Svetlana Lazebnik, and Alexander Schwing. 2018. Out of the box: Reasoning with graph convolution nets for factual visual question answering. In *Advances in Neural Information Processing Systems*, pages 2659–2670.

Christina Niklaus, Bernhard Bermeitinger, Siegfried Handschuh, and André Freitas. 2017. A sentence simplification system for improving relation extraction.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2018. A survey on open information extraction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3866–3878, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2019. Transforming complex sentences into a semantic hierarchy. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3415–3427, Florence, Italy. Association for Computational Linguistics.

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.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, pages 593–607. Springer.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Linfeng Song, Zhiguo Wang, Mo Yu, Yue Zhang, Radu Florian, and Daniel Gildea. 2018. Exploring graph-structured passage representation for multihop reading comprehension with graph neural networks. *arXiv preprint arXiv:1809.02040*.

Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4231–4242.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 189–198.

Yusuke Watanabe, Bhuwan Dhingra, and Ruslan Salakhutdinov. 2017. Question Answering from Unstructured Text by Retrieval and Comprehension.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association of Computational Linguistics*, 6:287–302.

Georg Wiese, Dirk Weissenborn, and Mariana Neves. 2017. Neural domain adaptation for biomedical question answering. *arXiv preprint arXiv:1706.03610*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. *arXiv preprint arXiv:1804.09541*.

Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, and Maosong Sun. 2018. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*.