The Computer Journal Advance Access published September 8, 2016

doi:10.1093/comjnl/bxw036

A Hybrid Approach for Question Retrieval in Community Question Answering

Long Chen*, Joemon M. Jose, Haitao Yu, and Fajie Yuan

School of Computing Science, University of Glasgow, Glasgow G12 8QQ, UK
**Corresponding author: long.chen@glasgow.ac.uk*

Community Question Answering (CQA) services, such as Yahoo! Answers and WikiAnswers, have become popular with users as one of the central paradigms for satisfying users' information needs. The task of question retrieval aims to answer one's query directly by finding the most relevant questions (together with their answers) from an archive of past questions. However, questions are always short text that there is a lexical gap between the queried question and the past questions. Furthermore, the underlying intents of two questions could be very different even if they bear a close lexical resemblance. To alleviate these problems, we present a hybrid approach that blends several language modelling techniques for question retrieval, namely, the Classic (query-likelihood) Language Model, the state-of-the-art Translation-based Language Model, and our proposed Semantic-based Language Model and Intent-based Language Model. The semantics of each candidate question is derived by a Probabilistic Topic Model, which makes use of local and global semantic graphs for capturing the hidden interactions among entities (e.g. people, places and concepts) in question–answer pairs. Experiments on two real-world data sets show that our approach can significantly outperform existing ones.

Keywords: question retrieval; language model; user intent; Semantic-based Topic Model; DBpedia

Received 24 November 2015; revised 19 May 2016
Handling editor: Fionn Murtagh

1. INTRODUCTION

In recent years, Community Question Answering (CQA) services, such as Yahoo! Answers, WikiAnswers, Quora and Stack Overflow have become popular knowledge sources for Internet users to access useful information online. When a user submits a new question (i.e. *query*) in CQA, the system would usually check whether similar questions have already been asked and answered before. If so, the user's query could be resolved directly by returning those archived questions (i.e. *documents*) with their corresponding answers.

Identifying similar questions in CQA archives is still a difficult task, since the questions asked by users are always short texts which would lead to a lexical gap between the queried question and the past questions. To address the above limitations, researchers have proposed the use of translation models [18, 31] to capture the semantic relations between words. While useful in general, the effectiveness of such models largely depends on the availability of high-quality parallel corpora (e.g. *question-answer pairs* in this context). Furthermore, simple word-level analysis model cannot handle cases where entities (e.g. *people, places* and *concepts*) are expressed by multi-word phrases.

To go beyond the mere word-level analysis, this paper proposes a question retrieval framework on the basis of semantic graphs, which makes use of an *external* resource (namely *DBpedia*) as the background knowledge to overcome the problem of the lexical gap between the queried question and the past questions. As a simple illustration, Fig. 1 shows a piece of global semantic graph produced by our proposed knowledge-rich approach[1] (*cf.* Section 4.3.1) from two questions: 'Why Kobe is better than Jordan?' and 'Who is better Kobe Bryant or Lebron James?'. From this graph, one can easily see that 'Basketball' and 'BasketballPlayer' are the central entities (i.e. *concepts* in *DBPedia*) of these two questions, even though these entities didn'(t) appear in the original questions. Furthermore, the rich

[1] https://github.com/long4glasgow/Semantics-based-Question-Retrieval

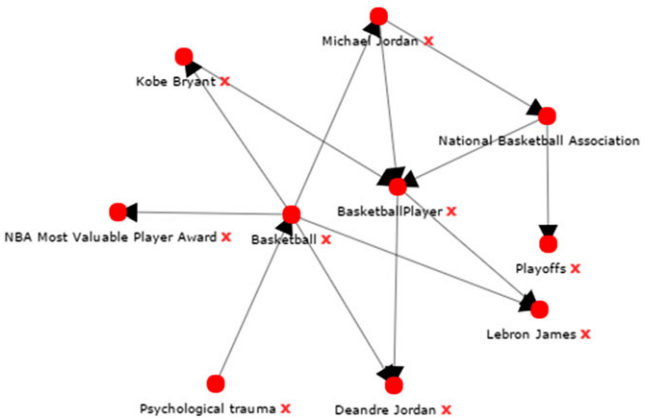

FIGURE 1. A piece of global semantic graph generated from two questions in Yahoo! Answers.

semantic connections among entities could be exploited to help inferring the latent topics of each question. Therefore, we would like to learn the interrelationships between entities in the global semantic graph, which would allow effective information sharing among all questions in the archive.

A local semantic graph for a question can also contribute for its answers, the inference of a question's latent topics could benefit from examining its similar questions based on their individual local semantic graphs. In other words, if two questions have a large degree of overlap in terms of their semantic entities and relations, they probably bear a close topical resemblance to each other. Therefore, we also construct local semantic graphs for each question–answer pair in the archive with the hope to utilize their semantic similarities.

In this work, a novel topic modelling framework is proposed, namely, Semantic Graph-based Topic Model (SGTM), which can seamlessly incorporate entities and relations from the semantic graphs into topic modelling. We then present a hybrid approach that blends several language modelling techniques for question retrieval, namely, the Classic (query-likelihood) Language Model, the state-of-the-art Translation-based Language Model, and our proposed Intent-based Language Model and Semantics-based Language Model. Experiments on two real-world data sets show that the hybrid approach to question retrieval outperforms existing approaches significantly.

2. RELATED WORK

2.1. Topic modelling

The techniques of topic modelling, such as Probabilistic Latent Semantic Analysis (PLSA) [14] and Latent Dirichlet Allocation (LDA) [29], provide an elegant mathematical way to analyze large volumes of unlabelled text. Recently, studies, such as Author-Topic Model (ATM) [27] and Contextual Focused Topic Model [9], try to integrate some outside information of network structure with topic modelling, but they mostly focus on homogeneous networks rather than heterogeneous networks (which are the networks consisted of different types of objects). Entity-Topic Model [20] combined LDA with entity–document relations, which is somewhat similar to our idea. However it assumes that an edge (entity–document) created in exactly the same way as a word, whereas our approach directly takes into account several types of relations (entity–document and entity–entity relations) through regularized propagation.

In Mei's seminal work [25], a homogeneous network was employed as a biased regularizer to overcome the overfitting problem of topic modelling. Deng *et al.* [11] combined PLSA [14] (see Section 4.2) with the regularizer learned from a heterogeneous network. However, it was originally designed for academic networks, and thus did not utilize the contextual information from any knowledge repositories. In addition, their framework only incorporates the heterogeneous network (i.e. relations among entities), while the homogeneous network (i.e. relations between entity pairs with weight) is completely ignored, whereas we consider both of them in our proposed framework.

2.2. Knowledge rich approaches

The recent advances in knowledge-rich approaches (e.g. DBPedia[2] and Knowledge Graph[3]) provide new techniques to gain insight into the semantic structure of a question archive. While enormous success has been made in several NLP tasks such as document similarity [26], topic labelling [17] and query expansion [10], the feasibility and effectiveness of such knowledge-rich approaches in topic modelling and question retrieval are mostly unknown.

Instead, we pruned our semantic graphs by filtering and weighting the edges (see Section 4.3.2). This is similar to the logics of Ref. [26], but their work attempts to produce graph-representation of documents for the task of document ranking, while we aim to construct semantic graphs for the task of topic modelling and question retrieval.

2.3. Question retrieval

The technique of language modelling has proved to be the best performing for question retrieval in CQA. The state-of-the-art approach utilized the Classic (query-likelihood) Language Model [32] (see Section 5.1) together with the Translation-based Language Model [18, 31] (see Section 5.2). We [1] have recently reported an approach to question retrieval based on the Question–Answer Topic Model, which uses semantic graph to learn the latent topics underlying the surface text of

[2] http://wiki.dbpedia.org/http://wiki.dbpedia.org/
[3] https://developers.google.com/freebase/

question–answer pairs. However, in that work the user intent of question–answer pairs is completely ignored, whereas in this work, we will investigate how to effectively combine the semantic-based approach with the intent-based approach. Furthermore, since the semantic graph based approach is quite computationally heavy, we will also take into account the time and space complexities in this work.

Cao *et al*. [6] have proposed a Category-based Language Model for question retrieval, but it requires users to manually label one topic category to each of their query questions, which is not always feasible. Instead of using the manually labelled categories, Cao *et al*. [5] later on improved their framework using category-based entries extracted from Wikipedia, but their Category-based Model cannot be applied straightforwardly to incorporate topics into question retrieval, because it assumes that a question belongs to one and only one topic, while we believe that a question can have multiple topics (each to a certain degree). In this paper, we address these concerns by proposing a Semantic-based Topic Model (*cf.* Section 4.3), which incorporate topic-question probabilities from the semantic graphs into topic modelling, where each question corresponds to a mixture of topics.

3. PRELIMINARIES

In this section, we formally introduce several concepts and notations.

DEFINITION 1. (Question). *A* **question** d *in a archive* D *is a sequence of words* $w_1 w_2 \cdots w_{|d|}$, *where* w_i *is a word from a fixed vocabulary. Following a common simplification in most work in information retrieval [14], we consider each question (i.e. the title) as a bag of words, and use* $n(d, w)$ *to denote the number of occurrence of word* w *in* d.

DEFINITION 2. (Entity). *An entity* e *in our system, given that we are using DBPedia resource URIs, can be either an instance or a concept in DBPedia. The former are concrete entries of DBPedia (e.g. dbpedia*: *Barrack_Obama), while the latter are the classes found within the DBPedia Ontology (i.e. the types of instances such as people, places and organizations).*

DEFINITION 3. (Semantic graph). *A semantic graph* G *consists of* V *the set of entities in the text data and* E *the set of edges representing the relations between entities. For instance, an edge* $\langle u, v \rangle$ *is a binary directed relation from entity* u *to entity* v, *where we use* $w(u, v)$ *to denote the weight of* $\langle u, v \rangle$.

We call the semantic graph built from the entire corpus (the archive of past questions) the *global* semantic graph, and those built from individual documents (i.e. questions and its answers) *local* semantic graphs

4. TOPIC MODELS

In this section, we propose a propagation algorithm to combine semantic graphs with the textual information for topic modelling, namely *Regularized Propagation*. The goal of this algorithm is to estimate the probabilities of topics for documents as well as other associated entities, in order to improve the performance of topic modelling.

4.1. Probabilistic topic models

In PLSA [14], an unobserved topic variable $z_k \in \{z_1, \ldots, z_K\}$ is inferred from the occurrences of different words $w_j \in \{w_1, \ldots, w_M\}$ in a particular document $d_i \in \{d_1, \ldots, d_N\}$. The joint probability of an observed pair (d,w) can be expressed as

$$P(d_i, w_j) = P(d_i) \sum_{k=1}^{K} P(w_j|z_k) P(z_k|d_i) \tag{1}$$

where $P(w_j|z_k)$ is the probability of word w_j occurring in topic z_k, and $P(z_k|d_j)$ is the probability of topic z_k for document d_i. The model parameters can be estimated by maximizing the log likelihood of the document collection D

$$L(D) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log \sum_{k=1}^{K} P(w_j|z_k) P(z_k|d_i) \tag{2}$$

through Expectation-Maximization (EM) [14].

PLSA provides a simplistic solution to find topics of documents. There is no guarantee that documents are associated with similar topics. Furthermore, in PLSA there is no constraint on the parameters $\theta_{ki} = P(z_k|d_i)$, the number of which grows linearly with the data. Therefore, the model tends to overfit the data. LDA [29] which is essentially the Bayesian version of PLSA, can alleviate the overfitting problem, but it still ignores the semantic relationships among documents. In this paper, we propose to use the semantic graphs of the document collection to enhance PLSA topic modelling.

4.2. Author topic model

Unlike PLSA, which associates each document with a distribution over topics, the ATM links each author with a distribution over topics and assumes that each document contains a mixture of multi-authors topic mixtures.

Specifically, given a document set $D = d_1, d_2, \ldots, d_{|D|}$, an author set $\mathbb{A} = a_1, a_2, \ldots, a_A$. Based on PLSA, one can define the following joint model for predicting:

$$P(w_i|d_j) = \sum_k P(w_i|z_k) P(z_k|d_j) \tag{3}$$

$$P(w_i|a_l) = \sum_k P(w_i|z_k) P(z_k|a_l) \tag{4}$$

where all the decompositions share a common topic $P(w_i|z_k)$. The model parameters can be estimated by maximizing the log likelihood of both the document collection D and the author collection $\mathbb{A}$.

$$L_A = \sum_i (\rho \sum_j N_{ij} \log \sum_k P(w_i|z_k) P(z_k|d_j) + \sigma \sum_l A_{il} \log \sum_k P(w_i|z_k) P(z_k|a_l)) \quad (5)$$

$$\rho + \sigma = 1 \quad (6)$$

through the EM algorithm described in Ref. [12]. The ATM allows for inferences about authors as well as documents, however, it does not consider the relationships between authors; furthermore, it is essentially a homogeneous networks since the rich information from the external resources is not exploited.

4.3. Semantic graph-based topic model

Generally speaking, an entity $e \in V$ is likely to be relevant to a specific topic of a question if its adjacent entities in the semantic graph are largely relevant to that topic. Following this intuition, we define a regularization term for the learning of the topic model:

$$R_V(G) = (1 - \mu) \sum_{i=1}^{|D|} \sum_{k=1}^{K} (P(z_k|d_i) - \sum_{e_i \in V_l} \sum_{e_u \in V_{d_i}} P(z_k|e_i) P(e_i|e_u))^2 + \mu \sum_{i=1}^{|D|} \sum_{k=1}^{K} (P(z_k|d_i) - \sum_{e_j \in V_g} \sum_{e_u \in V_{d_i}} P(z_k|e_j) P(e_j|e_u))^2 \quad (7)$$

where μ is a bias parameter, which strikes the optimal balance between the local semantic graph and the global semantic graph. V_l and V_g are two set of entities derived by local semantic graph and global semantic graph (see Section 4.3.1), respectively. $P(z_k|e_i)$ and $P(z_k|e_j)$ are estimated in the same way as $P(z_k|d_i)$ by using the EM algorithm (see Section 5.4.1). $w(e_i|e_u)$ is the weight between entity e_i and e_u in local semantic graph, $w(e_j|e_u)$ is the weight between entity e_j and e_u in global semantic graph (see Section 4.3.2).

To incorporate both the textual information and the semantic graph into the topic model, we define a *regularized propagation* framework by adding the regularization term with weight λ to the log-likelihood as follows:

$$L'_{rp}(D) = -(1 - \lambda) L(D) + \lambda R_V(G_D) \quad (8)$$

where R_V is defined in Equation (7). By minimizing R_V, we will smooth the topic distribution on the semantic graphs, where adjacent entities would have similar topic distributions. On the other hand, by minimizing $-L(C)$, we will find $P(z_i|d)$ and $P(w|z_i)$, which fit the text as much as possible. The parameter $\lambda \in [0, 1]$ is set to control the balance between the log-likelihood and the smoothness of topic distributions over the semantic graphs. When $\lambda = 0$, the objective function backs off to the standard PLSA. Minimizing $L'_{rp}(D)$ would only give us the topics, which best fit the content of the document collection D. When $\lambda = 1$, the objective function backs off to R_V, which can be deemed as a 'loss function' to measure how well the topic distributions on the semantic graph are consistent with the topic distribution on the documents. This is related to the objective of spectral clustering (e.g. ratio cut [7]). By minimizing $L'_{rp}(D)$, we can extract question clusters that making use of not only the text content of documents but also the structure of semantic graphs.

4.3.1. *Mapping questions into semantic graphs*

When computing $P(e_j|e_u)$ in the above SGTM, the method of Ref. [26] is adopted to construct the semantic graph. We start with a set of input entities C, which is found by using the off-the-shelf entity recognition tool DBpedia Spotlight.[4]

We then create a directed graph G as follows: (i) we define the set of entities V of G to be made up of all input entities, i.e. we set $V := C$ and (ii) we connect the entities in V based on the directed paths found between them in DBpedia. Specifically, the set of entities in V is expanded into a graph by conducting a depth-first search along the DBpedia graph and adding all the visited relations and entities, to a certain limit. So the finally constructed semantic graph consists of all the 'seed' entities identified from the documents together with all the edges found along the paths up to maximal length L that connect them. In this work, we set $L = 2$, as we find that the model with $L > 2$ tends to produce very large graphs and introduce lots of noise.

Figure 2 illustrates an example of a semantic graph generated from the set of entities {**db:Kobe Bryant, db:Michael Jordan**}, which are found in the question 'Why Kobe is better than Jordan?' Starting from these seed entities, we conduct a depth-first search to add relevant intermediate entities and relations to G (e.g. **dbo:BasketballPlayer** and **dbr:NBA**). As a result, we obtain a semantic graph with additional entities and edges, which provide us with rich knowledge about the original entities. Please note that we create two kinds of semantic graphs, namely, local semantic graph and global semantic graph. A local semantic graph is built for an individual question (associated with its answers) in order to detect its pairwise contextual similarities with other questions. The global semantic graph is constructed from the entities in the entire question archive, in order to capture the global contextual information of all existing questions.

[4] https://github.com/dbpedia-spotlight/dbpedia-spotlight

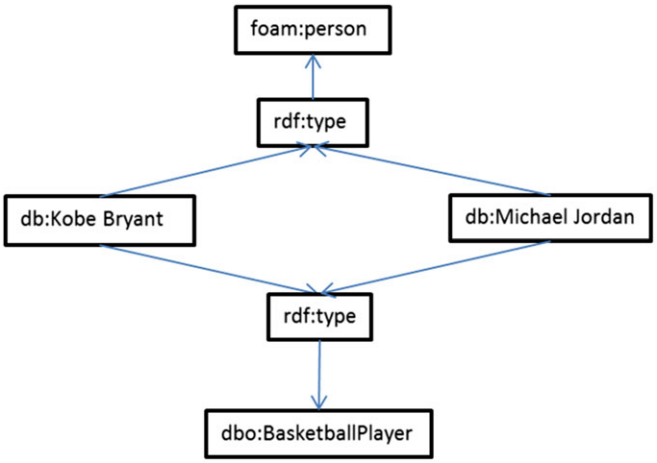

FIGURE 2. A sample semantic graph.

4.3.2. Semantic relation weighting

So far, we simply traverse a set of input entities from DBpedia graph. However, DBpedia ontology contains semantic relations at different levels, which may not be equally informative. For example, in Fig. 2, the seed entities **db:Kobe Bryant** and **db:Michael Jordan** can be connected through both **rdf:type dbo:BasketballPlayer** and **rdf:type foam:person**, but the later is less informative since it can apply to a large number of entities (i.e. all people in DBpedia). We can use real-valued weights to quantify the level of correlation between entities in the graph, and our weighting scheme will reward those edges that are most specific to the entities connected by them. Therefore, we define the weighting function as

$$W = -\log(P(W_{\text{Pred}})) \tag{9}$$

where W is the weight of an edge, $P(W_{\text{Pred}})$ is the probability that the predicate W_{Pred} (such as **rdf:type**) is describing the specific semantic relation. This measure is based on the hypothesis that specificity is a good estimator for relevance. We can compute the document frequency for each type of predicates, as we have the whole DBpedia database available and are able to query for all possible realizations of the variable X_{Pred}. $P(W_{\text{Pred}})$ is then defined in a similar way as the tf-idf [24] representation of W_{Pred}. In our example, an edge labelled with **rdf:type** will accordingly get a weight W, which is considerably lower than those labelled with **dbr:BasketballPlayer**.

There are often multiple relations between two entities, so the relation with the highest weight will be selected as the final edge. For instance, in the above example, **db:Kobe Bryant** and **db:Michael Jordan** can be connected by **dbr:BasketballPlayer** and **foaf:person**. The path to **dbr:BasketballPlayer** will be selected since it has a higher weight than **foaf:person**.

5. RETRIEVAL MODELS

5.1. Classic language model

Using the Classic (query-likelihood) Language Model [32] for information retrieval, we can measure the relevance of an archive question d with respect to the query question q as:

$$P_{cla}(q|d) = \prod_{w \in q} P_{cla}(w|d) \tag{10}$$

assuming that each term w in the query q is generated independently by the Unigram Model of document d. The probabilities $P_{cla}(w|d)$ are estimated from the bag of words in document d with Dirichlet prior smoothing.

5.2. Translation-based language model

There are often words mismatch between a query question and archive questions in CQA. For example, 'Where can I see movies for free online' and 'Anyone share me a DVD streaming link?' probably have the same meaning but are expressed in quite different words. It has been demonstrated that this issue could be addressed by the Translation-based Language Model [18, 31]:

$$P_{\text{tra}}(q|d) = \prod_{w \in q} P_{\text{tra}}(w|d) \tag{11}$$

$$P_{\text{tra}}(w|d) = \sum_{t \in d} P(w|t) P(t|d) \tag{12}$$

where $P(w|t)$ represents the probability of a document term t being translated into a query term w. As in Ref. [31], we estimate such word-to-word translation probabilities $P(w|t)$ on a parallel corpus that consists of 200 000 archived question–answer pairs from Yahoo! Answers.

5.3. Intent-based language model

We found that many questions are more inclined to have a social need rather than an informational one. For example, people may ask questions driven by a mindset of empathy, support or affection. In this case, a general social answer may be good enough to resolve questions of some certain patterns. For instance, when users ask a question containing the pattern 'do you agree with me', the best match will be the empathy answers from other people of 'Yes, you are right', regardless of any additional descriptions or links provided in the question. So it is necessary to identify the question type and find questions of similar intent match accordingly.

Let's denote w a word in query question q; let C_k be a class from the question intent taxonomy. Then what we aim to do is to calculate $P_I(w|d)$, which is the probability of a query

question q belonging to the class C_k containing the archive question document d.

$$P_I(w|d) = \sum_{k=1}^{N} P(w|C_k)P(C_k|d) \tag{13}$$

$$P(w|C_k) = \frac{\sum_{d\in C_k} tf(w, d)P(C_k|d)}{\sum_{w'\in d}\sum_{d\in C_k} tf(w', d)P(C_k|d)} \tag{14}$$

Note that the Intent-based Language Model is comprised of two stages, namely, the archive question training stage and the query question training stage. In archive question training stage, we put all the archive questions into the Co-training classifier to learn the $P(C_k|d)$, which is the probability of d being classified as C_k. We put this one as the first step as this is a time-consuming process, and its result serves as the basis for the computation of stage two. In the query question training stage, we calculate the probability of q being generated by C_k, which is the $P(w|C_k)$ in Equation (13). $P(C_k|d)$ is employed in stage two to learn how confidence the classifier it is for the prediction over each archive document in the category. The details regarding the design of Intent-based Language Model can be found in our previous work [8].

5.4. Semantics-based language model

The Semantics-based Language Model proposed by us aims to capture the latent topics in a better way via exploiting the hidden interactions among different entities in the SGTM. It can be descried formally as follows:

$$P_{\text{sem}}(q|d) = \prod_{w\in q} P_{\text{sem}}(w|d) \tag{15}$$

$$P_{\text{sem}}(w|d) = \sum_{k'=1}^{N} P(w|z_{k'})P_E(z_{k'}|d) \tag{16}$$

where $z_{k'}$ represents a latent topics, $P(w|z_{k'})$ is its corresponding Unigram Language Model and $P_E(z_{k'}|d)$ is the probability that the question d belongs to that topic (*cf.* Section 5.4.1).

Compared to the Topic-based Language Model of Ref. [19], the Semantics-based Language Model presented above is more general and more robust, because instead of learning topics solely from the document collection at the word-level, it utilizes the SGTM topics, which are at multiple semantic levels, namely words, entities and the entity relations extracted from the knowledge repository, DBpedia.

5.4.1. *Model fitting with the EM algorithm*

To estimate $P(w|z_k)$ and $P(z_k|d)$ in the above model, we use the EM algorithm, which alternates two steps, E-step and M-step. The unobserved latent variables in our model include $\phi = P(w_j|z_k)$, $\theta = P(z_k|d_i)$ and $\varphi = P(z_k|e_l)$.

Let us first consider the parameter estimation in PLSA. In E-step, we calculate the posterior probabilities $P(z_k|d_i, w_j, e_l)$:

$$P(z_k|d_i, w_j) = \frac{P(w_j|z_k)P(z_k|d_i)}{\sum_{k'=1}^{K} P(w_j|z_{k'})P(z_{k'}|d_i)} \tag{17}$$

$$P(z_k|d_i, e_l) = \frac{P(z_k|e_l)P(e_l|d_i)}{\sum_{k'=1}^{K} P(z_{k'}|e_l)P(e_l|d_i)} \tag{18}$$

In the M-step, we maximize the expected complete data log-likelihood for PLSA:

$$Q_D = \sum_{i=1}^{N}\sum_{j=1}^{M} n(d_i, w_j)\sum_{k=1}^{K} P(z_k|d_i, w_j) \log \sum_{k=1}^{K} P(w_j|z_k)P(z_k|d_i) \tag{19}$$

There is a closed-form solution [11] to maximize Q_D, which are listed in Equations (20), (21) and (22).

$$P(w_j|z_k) = \frac{\sum_{i=1}^{N} n(d_i, w_j)P(z_k|d_i, w_j)}{\sum_{j'=1}^{M}\sum_{i=1}^{N} n(d_i, w_{j'})P(z_k|d_i, w_{j'})} \tag{20}$$

$$P(z_k|d_i) = \frac{\sum_{j=1}^{M} n(d_i, w_j)P(z_k|d_i, w_j)}{\sum_{j'=1}^{M} n(d_i, w_{j'})} \tag{21}$$

$$P(z_k|e_l) = \frac{\sum_{s=1}^{S} n(e_l, w_s)P(z_k|d_i, e_l)}{\sum_{s'=1}^{S} n(e_l, w_{s'})} \tag{22}$$

Thus the model parameters in PLSA could be estimated efficiently using the standard EM algorithm. However, due to the introduction of the regularizer $R_V(G)$, there is no such closed form solutions for parameter estimation in SGTM (Equation (8)). Fortunately, we can use the generalized EM algorithm [25] to maximize the log-likelihood of $L'_{rp}(D)$. In the following, we will explain the model fitting procedure for $L'_{rp}(D)$.

It is easy to see that $L'_{rp}(D)$ and $L(D)$ share the same hidden variables z_k, and therefore could have the same E-step. Since the regularization $R_V(G)$ does not involve the parameter $P(w_j|z_k)$, we can still use the same M-step estimation for $P(w_j|z_k)$ as in Equation (20). Now the problem is how to estimate the parameter values $\phi = P(z_k|d_i)$ and $\theta = P(z_k|e_l)$. Instead of maximizing $P(z_k|d_i)$ and $P(z_k|e_l)$ directly, the generalized EM algorithm tires to improve the expected $P(z_k|d_i)$ as follows. First, we find $\theta^{(1)}_{t+1}$ using Equations (17) and (18), which maximizes Q_D instead of $L'_{rp}(D)$. Then, we start from $\theta^{(1)}_{t+1}$ and try to minimize $R_V(G)$,

which can be done through the Newton–Raphson method [4]. Given a function $f(x)$ and the initial value x_t, the Newton–Raphson updating formula to decrease $f(x)$ is $x_{t+1} = x_t - \xi(f'(x)/f''(x))$, where $0 \leq \xi \leq 1$ is the step parameter. With $\theta_{t+1}^{(1)}$, we can decrease $R(G)$ by updating $P(z_k|d_i)$ in each step.

$$P_E(z_k|d)_{t+1}^{(n+1)} = \xi P(z_k|d)_{t+1}^{(n)} + (1-\xi)\sum_{e\in V_d}\frac{P(z_k|e)}{|V_d|} \quad (23)$$

where $P(z_k|d_i, e_l)$ is obtained from the initial E-step, the step parameter ξ can be interpreted as a controlling factor of smoothing the topic distribution among the adjacent entities. It repeatedly updates $P_E(z_k|d)_{t+1}^{n}$ until $P_E(z_k|d)_{t+1}^{n+1} \leq P_E(z_k|d)_{t+1}^{n}$. We summarize the generalized EM algorithm for parameter estimation in this regularized propagation framework by using generalized EM algorithm in Algorithm 1.

5.5. Mixture model

Since the Classic Language Model, the Translation-based Language Model, the Intent-based Language Model, and the Semantics-based Language Model cover different grained semantic levels, it would be beneficial to combine their strengths for question retrieval. So we can mix the above language models via linear combination:

$$\begin{aligned} P_{mix1}(q|d) &= \alpha P_{cla}(q|d) + \beta P_{tra}(q|d) + \gamma P_{int}(q|d) \\ P_{mix2}(q|d) &= \alpha' P_{cla}(q|d) + \beta' P_{sem}(q|d) + \gamma' P_{int}(q|d) \\ P_{mix3}(q|d) &= \alpha'' P_{cla}(q|d) + \beta'' P_{tra}(q|d) + \gamma'' P_{sem}(q|d) \end{aligned} \quad (24)$$

where α, β and γ are three non-negative weight parameters satisfying $\alpha + \beta + \gamma = 1$. When $\gamma = 0$, the complete mixture model backs off to the current state-of-the-art approach, i.e. the combination of the Classic Language Model and the Translation-based Language Model [31].

Algorithm 1 Model fitting for regularized propagation.

Input : Input data, which includes: $G = (V, E)$ with word occurrences $n(d_i, w_j)$. The number of topics K, Newton step parameter ξ, regularization parameter λ.
Output: $\phi = P(w_j|z_k)$, $\theta = P(z_k|d_i)$, and $\varphi = P(z_k|e_l)$

1 1: Random initialize the probability distribution ϕ_0 and θ_0
2 2: $t \leftarrow 0$;
3 **while** $t < MaxIteration$ **do**
  $E-step$: Calculate $P(z_k|d_i, w_j)$ and $P(z_k|d_i, e_l)$ as in Equations (17) and (18)
  $M-step$:
4 Re−estimate $P(w_j|z_k)$ as in Equation (20)
5 Re−estimate $P(z_k|d_i)$ as in Equation (21)
6 Re−estimate $P(z_k|e_l)$ as in Equation (22)
7 $P(z_k|d_i)_{t+1}^{1} \leftarrow P(z_k|d_i)_{t+1}$
8
9 Calculate $P(z_k|d_i)_{t+1}^{2}$ as in Equation (23)
10 *while* $L'_{rp}(C)_{t+1}^{2} > L'_{rp}(C)_{t+1}^{1}$ *do*
11 $P(z_k|d_i)_{t+1}^{1} \leftarrow P(z_k|d_i)_{t+1}^{2}$
12 Calculate θ_{t+1}^{2}, update $P(z_k|e)_{t+1}$
13
14 *end*
15 *if* $L'_{rp}(C)\left(\theta_{t+1}^{(1)}\right) \geqslant L'_{rp}(C)(\theta_t)$ *then*
16
17 $P(z_k|d_i)_{t+1} \leftarrow P(z_k|d_i)_{t+1}^{1}$
18 update $P(z_k|e)_{t+1}$
19
20 *end*
21 *else*
22 Keep current θ, ϕ
  end
  $t \leftarrow t + 1$
23 **end**

6. EXPERIMENTS

6.1. Experimental setup

We conducted experiments on two real-world CQA data sets. The first data set, YA, comes from Yahoo! Answers. It is part of Yahoo! Labs' Webscope[5] L6 data set that consists of 4 483 032 questions with their answers from 1 January 2006 to 1 January 2007.

The second data set, WA, comes from WikiAnswers. It contains 824 320 questions with their answers collected from WikiAnswers[6] from 1 January 2012 to 1 June 2012.

In order to produce the semantic graphs, we first apply DBpedia Spotlight on each question–answer pair of these two data sets. We use the *subject* field as question part and the *bestanswer* field as the answer part. The text of questions and answers have been preprocessed by case-folding and stopword-removal (using a standard list of 418 common words). Given the disambiguated entities (see Section 4.3.1), we create local and global entity collections, respectively, for constructing local and global semantic graphs. The creation process of entity collections is organized as a pipeline of filtering operations:

(1) The isolated entities, which have no connections with the other members of the entity collection in the

[5] http://webscope.sandbox.yahoo.com/
[6] http://wiki.answers.com/

DBpedia repository, would be removed, since they are almost useless in the topic propagation process.
(2) The infrequent entities, which appear in less than five documents when constructing the global entity collection, would be discarded.
(3) Similar to the previous step, we discard entities that appear less than twice in the question archive when constructing the local entity collections.

6.2. Experiments with topic modelling

We first experimented with topic modelling on 40 000 questions that are randomly sampled from the top 10 categories of YA and WA data set, respectively. The statistics of these two data sets along with their corresponding entities and relations are shown in Table 1. The top 10 categories distributions in these two data sets are shown in Fig. 3.

We randomly split each of the data set into a training set, a validation set, and a test set with the ratio 2:1:1. We learned the parameters of the SGTM as well as several other representative topic models from the training set, tuned the parameters of each model on the validation set and evaluated the performance of each model on the test set. To demonstrate the effectiveness of our SGTM method, we compare it with the following topic modelling techniques:

TABLE 1. Statistics of the YA and WA data sets.

	YA	WA
No. of questions	40 000	40 000
No. of entities (local)	123 263	83 541
No. of entities (global)	27 324	24 750
No. of relations (local)	142 454	159 492
No. of relations (global)	71 719	69 713

- **PLSA:** The baseline approach, which only employs the classic Probabilistic Latent Semantic Analysis [14].
- **ATM:** Author Topic Model, which combines LDA with authorship network [27]. In our experiments, authors are replaced with entities.
- **TMBP:** The state-of-the-art approach, Topic Model with Biased Propagation [11], which combines PLSA with an entity network (without using any external knowledge, such as DBpedia).
- **SGTM:** Our proposed Semantic Graph-based Topic Model (see Section 4.3).

In order to evaluate our proposed topic model and compare it to existing ones, we use two metrics, accuracy (AC) and normalized mutual information (NMI), which are popular for evaluating the effectiveness of clustering methods. The accuracy is defined as $\mathrm{AC} = \sum_1^n \delta(a_i, map(l_i))/n$ [30], where n denotes the total number of questions, $\delta(x, y)$ is the delta function that equals one if $x = y$ and zero otherwise, and $map(l_i)$ is the mapping function that maps each cluster label l_i to the corresponding label from the data corpus. Given two sets of document clusters, C and C', their mutual information is defined as: $\mathrm{MI}(C, C') = \sum_{c_i \in C, c'_j \in C'} p(c_i, c'_j) \cdot \log_2 \left(p(c_i, c'_j) / p(c_i) \cdot p(c'_j) \right)$ [30], where $p(c_i)$ and $p(c'_j)$ are the probabilities that a randomly chosen document belongs to the clusters c_i and c'_j, respectively, and $p(c_i, c'_j)$ is the joint probability that a randomly chosen document belongs to the cluster c_i and c'_j at the same time.

Parameter Setting: For PLSA, we only use question–answer pairs for question clustering with no additional entity information. For ATM, we use symmetric Dirichlet priors in the LDA estimation with $\alpha = 50/K$ and $\beta = 0.01$, which are common settings in the literature. For TMBP, an entity-based heterogeneous network is constructed, and its parameter settings were set to be identical to those in Ref. [11].

As to the algorithm running process, it takes 179 iterations for SGTM to converge. Since SGTM is the most complex one, we expected all the other models after 200 iterations would

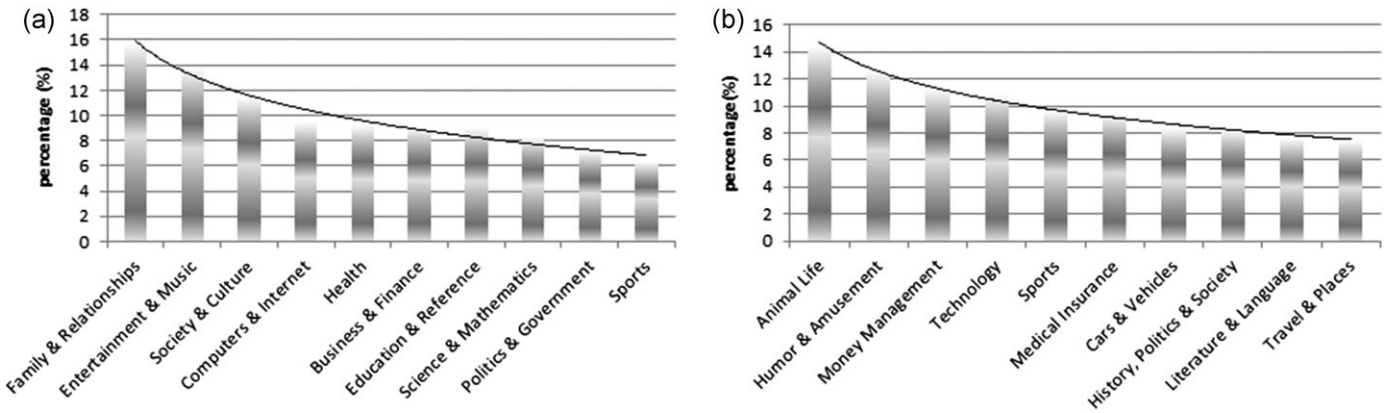

FIGURE 3. (**a**,**b**) The category distribution of YA and WA data sets, respectively.

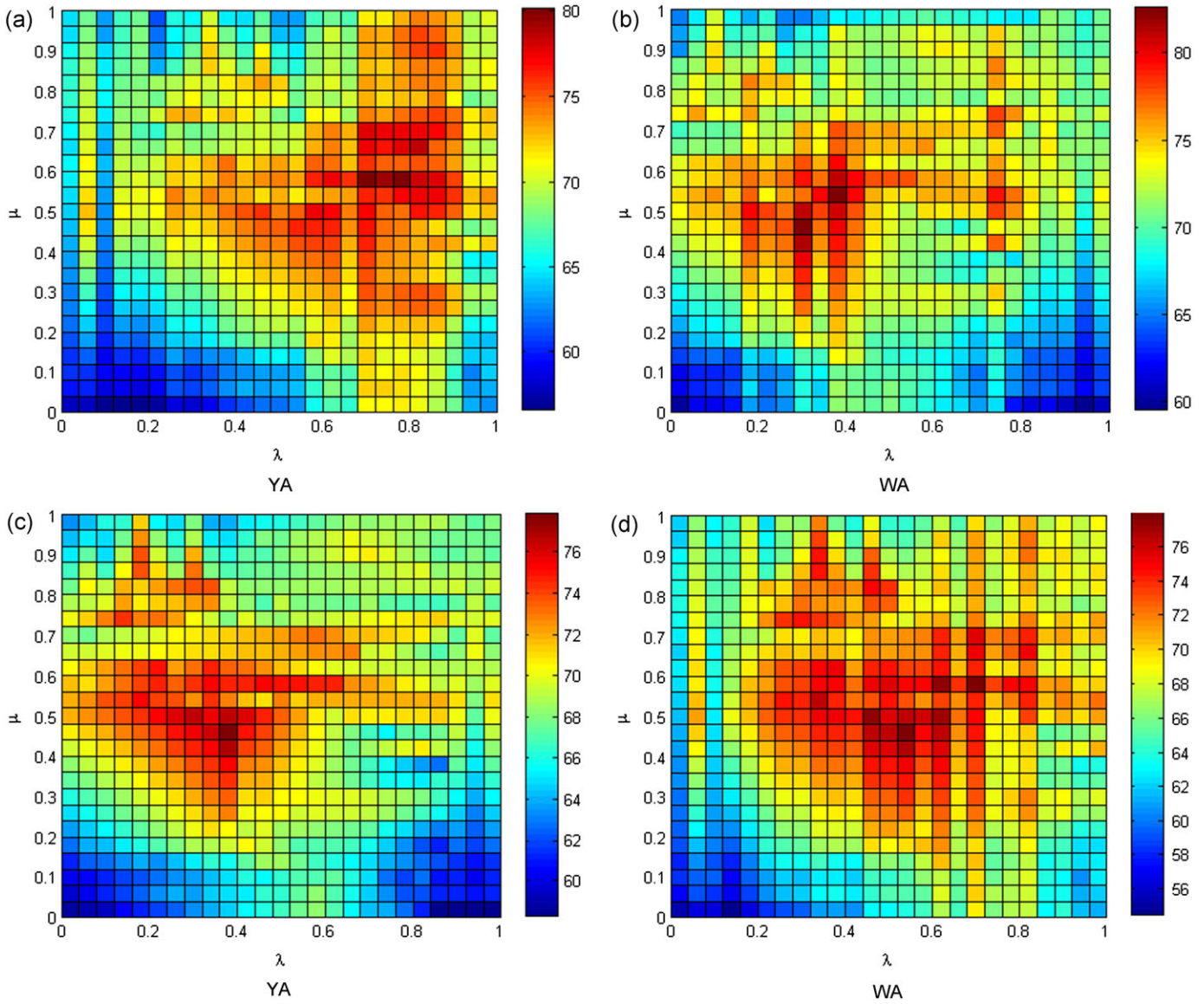

FIGURE 4. (**a**,**b**) The accuracy (%) of SGTM framework with varying parameters λ and μ. (**c**,**d**) The NMI (%) of SGTM framework with varying parameters λ and μ.

arrive at 100% convergence rate. The time-cost for EM training of SGTM is 5.2 hours, which is 1 hour more than TMBP, 2 hours more than ATM and 3.5 hours more than PLSA.

Consistent to our previous setting of top 10 categories, we set the number of topics (K) to be 10 for both YA and WA. Figure 4 shows how the SGTM clustering performance varies with the different parameter values. The essential parameters in the SGTM framework are λ and μ. When $\lambda = 0$, it is the baseline PLSA model. When $\lambda = 1$, it is entirely determined by the graph regulation term R_V (*cf.* Equation (7)). The performance of question clustering was tuned on the validation set and evaluated on the training set through 5-fold validation. The results reported in Fig. 4 are those averaged over the five trials. It can be seen that SGTM with global semantic graphs generally performs better than SGTM with local semantic graphs, which possibly suggests that the global context is more important than the local context for the purpose of question clustering. Furthermore, the best performance is achieved when combining these two with the parameter setting: $\lambda = 0.4$ and $\mu = 0.5$.

Table 2 depicts the question clustering performances of different topic modelling methods. For each method, 20 test runs are conducted on the test set, and the final performance scores were calculated by averaging the scores from those runs. It can be seen that ATM outperforms the baseline PLSA with additional entity network information. As expected, TMBP

TABLE 2. The clustering performance of different methods on (a) YA and (b) WA datasets.

	PLSA	ATM	TMBP	SGTM
(a) YA				
AC	0.662-*-*	0.685-*	0.724-*	0.753
NMI	0.657-*-*	0.732-*	0.765-*	0.819
(b) WA				
AC	0.636-*-*	0.649-*-*	0.652-*	0.694
NMI	0.654-*-*	0.689-*-*	0.717-*	0.734

-*-* and -* indicate the statistical significance of performance decrease from that of SGTM with P-value < 0.01 and P-value < 0.05, respectively.

outperforms ATM since it directly incorporates the heterogeneous network of entities. Our proposed SGTM improves accuracy by 9.1% over the classical PLSA baseline, and 2.9% over the state-of-the-art TMBP on YA data set. A comparison using the paired t-test clearly shows that SGTM outperforms all the other methods PLSA, ATM and TMBP significantly. This indicates that exploiting the semantic graph knowledge greatly improve the performance of topic modelling.

6.3. Experiments with question retrieval

We then experimented with question retrieval with a similar set up in [31]: 50 questions were randomly sampled from the YA and WA data sets, respectively, for testing, and the top archive questions (i.e. search results) returned for each test query question were manually labelled as either relevant or not. We then randomly split each of the data set into a training set and a test set with a ratio 1:1.

For retrieval experiments, we compared the following four approaches:

- the baseline approach, which only employs the Classic Language Model (C);
- the state-of-the-art approach, which combines the Classic Language Model and the Translation-based Language Model (C+T) [31];
- the proposed hybrid approach, which blends the Classic Language Model, the Translation-based Language Model and the Intent-based Language Model (C+T+I) [8];
- the proposed hybrid approach, which blends the Classic Language Model, the Translation-based Language Model and the Semantic-based Language Model (C+T+S);
- the proposed hybrid approach, which blends the Classic Language Model, the Semantic-based Language Model and the Intent-based Language Model (C+S+I).

TABLE 3. The model parameters for different question retrieval approaches.

	C	C + T	C + T + I	C + T + S	C + S + I
α	1	0.3	0.18	0.14	0.20
β	0	0.7	0.42	0.33	0.45
γ	0	0.0	0.40	0.53	0.35

Parameter Setting: All parameter values of retrieval models were tuned based on Precision at 10 (P@10) [24] and Mean Average Precision (MAP) [24]. The model parameters in Equation (24) were tuned on the training data to achieve optimal results, as shown in Table 3. In the mixture models (C+T) and (C+T+I), the ratio between parameter values α and β was same as that in Ref. [31]. As there is no prior knowledge for C + S, and C + S + I, their settings are empirically learned from the training data. All the other parameters were set as identical to that of Section 6.2.

So far, the parameters values of SGTM and TMBP for question retrieval were all derived from the optimal results of Section 6.2, and the number of topics K was empirically set as 10 as we need the data label for calculating the accuracy. However, 10 topics are not necessarily the optimal value for the question retrieval task. Furthermore, it is well known that larger data sets may need more topics in general. Hence, we experiment the Mixture Language Model with different values of K on the training set. As shown in Fig. 5, it is clear that the semantics-based approach (C+S+I) achieves the best result when $K = 40$, and the intent-based approach (C+T+I) achieves the optimal result when $K = 50$.

Given the optimal parameters, the retrieval performances of those approaches on test set, measured by P@10 and MAP, are reported in Table 4. Consistent to the observation in [31], adding the Translation-based Language Model (C+T) brings substantial performance improvement to the Classic Language Model (C). More importantly, it is clear that our proposed hybrid approach incorporating the semantics-based language (C+S+I) outperforms the state-of-the-art approaches (C+T), (C+T+I) and (C+T+S) significantly, according to both P@10 and MAP on YA and WA.

6.4. Time and space complexity

Another natural question is whether our proposed model can fit into the real world scenarios. Therefore, we have also evaluated the overhead of C, C + T, C + T + I, C + T + S, and C + S + I, respectively, by measuring the average runtime per query when retrieving the results of Table 4. The experiments are conducted on a laptop with 16 GB memory and an Intel i7–740QM processor with 1.73 GHz clock rate, 4256 KB L2 Cache and 6 MB L3 Cache. Furthermore, we only use a single sampling thread, and thus only one CPU core is active throughout. We further disabled Turbo Boost to

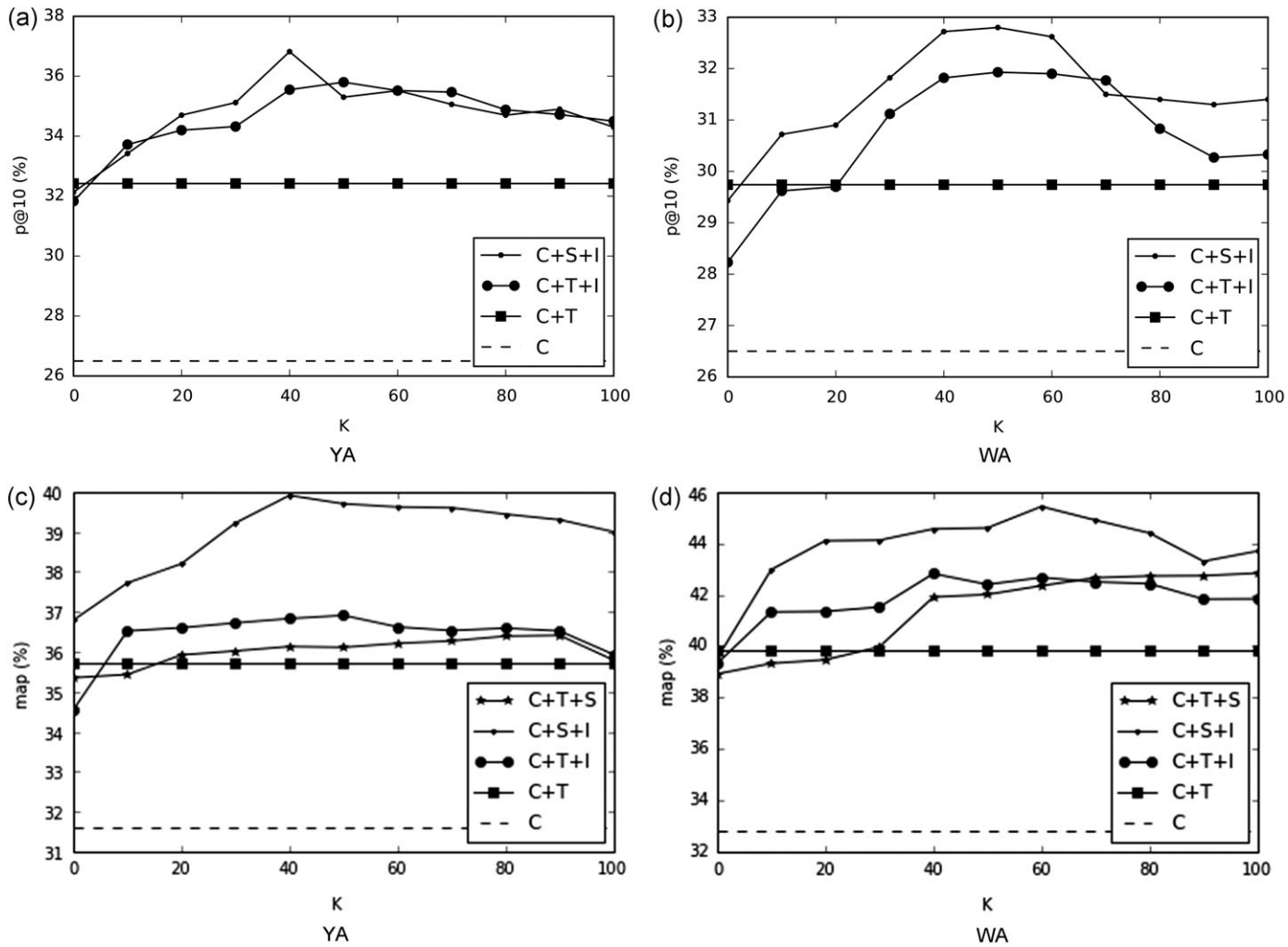

FIGURE 5. (**a**,**b**) The P@10 (%) of retrieval models with varying number of topics K. (**c**,**d**) The NMI (%) of retrieval models with varying number of topics K.

ensure all experiments are run at exactly 1.73 GHz clock rate. All codes are written under the same platform over JAVA Eclipse. As can be observed from Table 5, C + T + S and C + S + I share a similar runtime to the other baseline approaches. This is because once the topic model (SGTM) is successfully constructed and saved, the probabilities of $P(z|d)$ (see Equation (16)) in Semantic-based Language Model for predicting a new question can be quickly obtained from the existing posterior probabilities of $p(w|z)$.

The time complexity for constructing the semantic graph is quite expensive indeed, which is $O(n^2)$ (n is the number of questions in the corpus), since each question needs to be matched against DBpedia with all the other questions in the corpus for constructing local semantic graphs. The time for fitting the SGTM is expensive as well, since the Newton–Raphson's method is adopted in the learning process and it took 179 round for the model to converge on YA data set. The overall pre-processing time in our experiment for SGTM is 27.3 hours. However, once the SGTM has successfully constructed and trained, the execution time per query is comparable to the current baseline approaches. The semantic graphs can also be updated in an incremental fashion. For example, one can update the semantic graph by conducting a

TABLE 4. The experimental results.

	C	C + T	C + T + I	C + T + S	C + S + I
P@10 (YA)	0.265	0.324	0.346	0.349	0.354
MAP (YA)	0.316	0.358	0.422	0.418	0.453*
P@10 (WA)	0.263	0.297	0.335	0.315	0.351*
MAP (WA)	0.328	0.394	0.441	0.432	0.473**

Statistical significance using t-test : ** indicates P-value < 0.01 while * indicates P-value < 0.05.

TABLE 5. The execution time (milliseconds) per query question.

	C	C + T	C + T + I	C + T + S	C + S + I
YA	2135	3378	3347	3138	3349
WA	1812	3030	3238	3122	3227

depth-first search to connect the entities of the new questions with all the existing ones, with the expected time complexity of $O(n)$. Adding a new entity into semantic graphs in our experiment costs as little as 8.3 seconds on average. In addition, when it comes to the storage cost, the global semantic graph took up 36.5 MB space and the local semantic graphs took up 164.5 MB space, which is relatively small considering the overall YA data set size of 3.6 GB.

7. CONCLUSION AND FUTURE WORK

In this paper, we propose a Hybrid Language Model for question retrieval. The new model supersedes the existing category-based language models because (i) question topics are more fine-grained than question categories; (ii) a question resides in only one category but could belong to multiple topics and (iii) the integration of semantic graphs and user intent enables our topic model to capture the hidden contextual semantics of questions.

There are several interesting and promising directions in which this work could be extended. First, when learning the latent topics of questions, we could also include some metadata features such as the questions' categories (which have recently been shown to be useful for question retrieval [6]). Second, SGTM in the current form relies on one of the simplest topic models (PLSA), which makes sense as a first step towards integrating semantic graphs into language model, but of course we could consider using more sophisticated word representation methods, such as word embedding.

REFERENCES

[1] Chen, L., Jose, J.M., Yu, H., Yuan, F. and Zhang, D. (2016) A semantic graph based topic model for question retrieval in community question answering. In *Proc. 9th ACM Int. Conf. Web Search and Data Mining*, pp. 287–296, ACM.

[2] Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R. and Hellmann, S. (2009) Dbpedia-a crystallization point for the web of data. *Web Semant.*, **7**, 154–165.

[3] Blei, D.M., Ng, A.Y., Jordan, M.I. (2003) Latent dirichlet allocation. *J Machine Learn. Res.*, **3**, 993–1022

[4] Cai, D., Mei, Q., Han, J. and Zhai, C. (2008) Modeling hidden topics on document manifold. In Proc. 17th ACM Conf. Information and Knowledge Management, pages 911–920. ACM.

[5] Cao, X., Cong, G., Cui, B. and Jensen, C.S. (2010) A generalized framework of exploring category information for question retrieval in community question answer archives. In *Proc. 19th Int. Conf. World Wide Web*, pp. 201–210. ACM.

[6] Cao, X., Cong, G., Cui, B., Jensen, C.S. and Zhang, C. (2009) The use of categorization information in language models for question retrieval. In *Proc. the 18th ACM Conf. Information and Knowledge Management*, pp. 265–274. ACM.

[7] Chan, P.K., Schlag, M.D.F., Zien, J.Y. (1994) Spectral k-way ratio-cut partitioning and clustering, *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, **13**, 1088–1096

[8] Chen, L., Zhang, D. and Levene, M. (2013) Question retrieval with user intent. In *Proc. 36th Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, pp. 973–976. ACM.

[9] Chen, X., Zhou, M. and Carin, L. (2012) The contextual focused topic model. In *Proc. 18th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 96–104. ACM.

[10] Dalton, J., Dietz, L. and Allan, J. (2014) Entity query feature expansion using knowledge base links. In *Proc. the 37th Int. ACM SIGIR Conf. Research & Development in Information Retrieval*, pp. 365–374. ACM.

[11] Deng, H., Han, J., Zhao, B., Yu, Y. and Xide, C. (2011) Lin. Probabilistic topic models with biased propagation on heterogeneous information networks. In *Proc. 17th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 1271–1279. ACM.

[12] Deng, H., Zhao, B. and Han, J. (2011) Collective topic modeling for heterogeneous networks. In *Proc. 34th Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, pp. 1109–1110. ACM.

[13] Weiwei Guo and Mona Diab. (2011) Semantic topic models: Combining word distributional statistics and dictionary definitions. In *Proc. Conf. Empirical Methods in Natural Language Processing*, pp. 552–561. Association for Computational Linguistics.

[14] Hofmann, T. (2001) Unsupervised learning by probabilistic latent semantic analysis. *Machine Learn*, **42**(1–2), 177–196.

[15] Hong, L., Dom, B., Gurumurthy, S. and Tsioutsiouliklis, K. (2011) A time-dependent topic model for multiple text streams. In *Proc. 17th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 832–840. ACM.

[16] Hörster, E., Lienhart, R. and Slaney, M. (2007) Image retrieval on large-scale image databases. In *Proc. 6th ACM Int. Conf. Image and Video Retrieval*, pp. 17–24. ACM.

[17] Hulpus, I., Hayes, C., Karnstedt, M. and Greene, D. (2013) Unsupervised graph-based topic labelling using dbpedia. In *Proc. 6th ACM Int. Conf. Web Search and Data Mining*, pp. 465–474. ACM.

[18] Jeon, J., Croft, W.B. and Lee, J.H. (2005) Finding similar questions in large question and answer archives. In *Proc. 14th ACM Int. Conf. Information and knowledge Management*, pp. 84–90. ACM.

[19] Ji, Z., Xu, F., Wang, B. and He, B. (2012) Question-answer topic model for question retrieval in community question answering. In *Proc. 21st ACM Int. Conf. Information and Knowledge Management*, pp. 2471–2474. ACM.

[20] Kim, H., Sun, Y., Hockenmaier, J. and Han, J. (2012) Etm: Entity topic models for mining documents associated with entities. In *2012 IEEE 12th Int. Conf.*, *Data Mining (ICDM),* pp. 349–358. IEEE.

[21] Li, F., He, T., Tu, X. and Hu, X. (2012) Incorporating word correlation into tag-topic model for semantic knowledge acquisition. In *Proc. 21st ACM Int. Conf. Information and Knowledge Management*, pp. 1622–1626. ACM.

[22] Li, H., Li, Z., Lee, W.-C. and Lee, D.L. (2009) A probabilistic topic-based ranking framework for location-sensitive domain information retrieval. In *Proc. 32nd Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, pp. 331–338. ACM.

[23] Li, L., Roth, B. and Sporleder, C. (2010) Topic models for word sense disambiguation and token-based idiom detection. In *Proc. 48th Annual Meet. Assoc. Comput. Linguist.*, pp. 1138–1147. Association for Computational Linguistics.

[24] Manning, C.D., Raghavan, P., Schütze, H. (2008) *Introduction to Information Retrieval*, **vol. 1**. Cambridge university press, Cambridge.

[25] Mei, Q., Cai, D., Zhang, D. and Zhai, C.X. (2008) Topic modeling with network regularization. In *Proc. 17th Int. Conf. World Wide Web*, pp. 101–110. ACM.

[26] Schuhmacher, M. and Ponzetto, S.P. (2014) Knowledge-based graph document modeling. In *Proc. 7th ACM Int. Conf. Web Search and Data Mining*, pp. 543–552. ACM.

[27] Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L. and Su, Z. (2008) Arnetminer: extraction and mining of academic social networks. In *Proc. 14th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 990–998. ACM.

[28] Titov, I. and McDonald, R. (2008) Modeling online reviews with multi-grain topic models. In *Proc. 17th Int. Conf. World Wide Web*, pp. 111–120. ACM.

[29] Wei, X. and Croft, W.B. (2006) Lda-based document models for ad-hoc retrieval. In *Proc. 29th Annual Int. ACM SIGIR Conf. Res. Develop. Information Retrieval*, pp. 178–185. ACM.

[30] Xu, W., Liu, X. and Gong, Y. (2003) Document clustering based on non-negative matrix factorization. In *Proc. 26th Annual Int. ACM SIGIR Conf. Res. Develop. Informaion Retrieval*, pp. 267–273. ACM.

[31] Xue, X., Jeon, J. and Croft, W.B. (2008) Retrieval models for question and answer archives. In *Proc. 31st Annual Int. ACM SIGIR Conf. Res. Develop. Information Retrieval*, pp. 475–482. ACM.

[32] Zhai, C.X. (2008). *Statistical Language Models for Information Retrieval* (**vol. 1**), pp. 1–141. Synthesis Lectures on Human Language Technologies.

[33] Zhu, X., Ghahramani, Z., Lafferty, J. (2003) Semi-supervised learning using gaussian fields and harmonic functions. *ICML*, **3**, 912–919.