Latent Space Embedding for Retrieval in Question-Answer Archives

Deepak P[1] **Dinesh Garg**[2] **Shirish Shevade**[3]
[1]Queen's University Belfast, Northern Ireland, UK deepaksp@acm.org
[2]Indian Institute of Technology Gandhinagar, India dgarg@iitgn.ac.in
[3]Indian Institute of Science, Bangalore, India shirish@csa.iisc.ernet.in

Abstract

Community-driven Question Answering (CQA) systems such as Yahoo! Answers have become valuable sources of reusable information. CQA retrieval enables usage of historical CQA archives to solve new questions posed by users. This task has received much recent attention, with methods building upon literature from translation models, topic models, and deep learning. In this paper, we devise a CQA retrieval technique, *LASER-QA*, that embeds question-answer pairs within a unified latent space preserving the local neighborhood structure of question and answer spaces. The idea is that such a space mirrors semantic similarity among questions as well as answers, thereby enabling high quality retrieval. Through an empirical analysis on various real-world QA datasets, we illustrate the improved effectiveness of LASER-QA over state-of-the-art methods.

1 Introduction

Community-based Question Answering (CQA) services such as Yahoo! Answers[1], Quora[2], StackOverflow[3], and Baidu Zhidao[4] have become a dependable source of knowledge to solve common user problems. These allow a user to post queries such as *how* and *why* questions that seek descriptive solutions and opinions as answers. Over time, these services build up a large archive of question-answer knowledge that may be leveraged to solve new user questions. The CQA retrieval problem, that has received much recent attention, is about addressing this opportunity. CQA retrieval methods focus on finding historical archived knowledge (questions, answers or QA pairs) that are relevant to a newly posed user question. The central technical challenge that differentiates CQA retrieval from other general purpose IR tasks is that of the need to address the lexical gap (aka *lexical chasm*) in QA archives. Lexical chasm means that text fragments in questions (e.g., *disk full*) may lead to semantically correlated content in answers (e.g., *format*). This QA-correlation is different from semantic relatedness such as *synonymy* and *antonymy*; in the above example, the correlation is due to *disk full* issues often leading to solution involving disk *format*ing. Explicit correlation modelling, using statistical translation models, have met with much success in CQA retrieval.

Table 1: Example CQA Pairs

#	QA	Cause
1	**Q:** My internet connection is not working, my router shows the "Internet" led blinking in red.	Router Authentication Issue
	A: Please go to the router login page and re-login with broadband credentials; click "connect" and you should be on the internet.	
2	**Q:** My internet connection is not working, only the power led is lit in the router.	Router Loose Connection
	A: Can you check whether the broadband cable is plugged in. Maybe, the broadband cable is not connected properly.	

In this paper, we take a neighborhood preserving learning approach, and learn a unified representation for QA pairs in an abstract *latent space*. Consider two example CQA pairs from a technical support forum presented in Table 1; the intuitive causes listed alongside are external to the dataset. Though the questions are reasonably similar lexically, they pertain to very different issues

[1] https://answers.yahoo.com/
[2] https://www.quora.com/
[3] http://stackoverflow.com/
[4] https://en.wikipedia.org/wiki/Baidu_Knows

Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 866–876
Copenhagen, Denmark, September 7–11, 2017.

as illustrated by the wide disparity in the answers posed to them. We model QA-pairs in a unified space that preserves the similarity neighborhood in question and answer spaces. In this example, the wide divergence in answer-space similarity neighborhoods between the two QAs would pull them apart, so they live in different parts of the latent space, reflecting the dissimilarity between their causes. Thus, our contribution in this paper is a neighborhood-preserving method for CQA retrieval, LASER-QA, expanding to **LA**tent-**S**pace **E**mbedding for **R**retrieval in **QA** archives.

2 Related Work

The three main CQA retrieval tasks target retrieving (a) related past questions (Zhou et al., 2015), (b) potentially usable past answers (Shtok et al., 2012), and (c) past question-answer pairs (Xue et al., 2008). Techniques for CQA typically use one of: (i) statistical translation models, (ii) topic models and (iii) neural networks. A fourth class target exploiting metadata such as question categories and author data, or domain-specific syntactic information, and are not as applicable in the absence of such information.

In the interest of keeping this section focused on retrieval, we do not cover other tasks that have been addressed for CQA, such as QA-pair discovery (Deepak and Visweswariah, 2014), clustering (Deepak, 2016) and auxiliary IR tasks such as query suggestions (Deepak et al., 2013).

2.1 Translation Model based Techniques

Translation models (Brown et al., 1990) take parallel corpora, collections of document pairs expressing the same thing in different natural languages, and learn correlations between words/phrases; for example, $p(f|e)$ quantifies the probability of an english word *e* getting translated to a french word *f* in an English-French translation system. Though question-answer pairs do not semantically qualify as parallel corpora, usage of translation models treating them so(Xue et al., 2008) have led to retrieval accuracy improvements. Simplistically, a high probability for $p(format|disk)$ leads to retrieval models boosting the score of a answer containing the word *format* to respond to a user query involving a *disk* problem. Later methods have improved upon them by phrase-level (Zhou et al., 2011) and entity-level (Singh, 2012) modelling as well as by unimportant word removal (Lee et al., 2008) and differential treatment of concepts (Park and Croft, 2015). Recent work has even explored using a different language (e.g., Chinese) to enrich questions (Zhang et al., 2015).

2.2 Topic Model based Techniques

Topic models (Blei et al., 2003) have been used to retrieve topically similar questions (Cai et al., 2011) with usage of the solution side leading to further improvements (Ji et al., 2012). They have been combined with language modeling whereby question and answer parts are modeled to have been generated from *paired* latent topics, but in "question and answer languages" (Zhang et al., 2014). We will use such paired topic modelling, called TBLM, as a baseline in our experimental study.

2.3 Topic+Translation Models

Hybrid methods build upon topic and translation models by interpolating the separate scorings. Due to the usage of a combination of multiple types of parameterized models, the results of such "pipeline methods" have been observed to be hard to reproduce (Qiu et al., 2013).We use a recent hybrid scoring method, called TopicTRLM (Zhou et al., 2015), as a baseline in our experimental study.

2.4 Deep Learning Methods

Neural networks such as DBNs (Wang et al., 2011; Hu et al., 2013) and more sophisticated neural pipelines (Shen et al., 2015) have been explored for CQA retrieval. A recent work (Nakov et al., 2016a) trains a neural network to discriminate between good and bad comments for a question. Using neural networks for retrieval within question datasets (not involving answers) has also been a subject of recent interest (e.g., (Bogdanova et al., 2015; Das et al., 2016)). The most recent method for generic QA-pair processing, which we will call as AENN (Zhou et al., 2016), trains separate auto-encoders for question and answer corpora, and induces correlatedness of intermediate representations in a fine-tuning step. In our empirical analysis, we will use the AENN approach from (Zhou et al., 2016) as a baseline.

2.5 Auxiliary-information based Methods

This category of methods target to exploit specific kinds of auxiliary information that are potentially

available with CQA data. Techniques have considered usage of question categories (Cao et al., 2009; Zhou et al., 2014), the split between question title and description (Qiu et al., 2013), and assumptions of the question syntax (Duan et al., 2008). While such information is available in many systems, QA information from systems such as forums and chat-based customer support sometimes have very little information other than just QA-pairs. We target a general scenario where such metadata is not expected as a pre-requisite, as in the case of most techniques from other categories.

3 Problem Statement

Let $\mathcal{D} = \{(\boldsymbol{q}_1, \boldsymbol{a}_1), \ldots, (\boldsymbol{q}_n, \boldsymbol{a}_n)\}$ be QA pairs from a CQA archive where answer $\boldsymbol{a}_i$ is associated with question $\boldsymbol{q}_i$; for cases involving multiple answers for a question, the question would be replicated for each answer. For a new question $\boldsymbol{q}$, the CQA retrieval problem is about devising a scoring function $f(\boldsymbol{q}, (\boldsymbol{q}_i, \boldsymbol{a}_i))$ that quantifies the relevance of each $(\boldsymbol{q}_i, \boldsymbol{a}_i)$ pair from $\mathcal{D}$ to the new question $\boldsymbol{q}$. Having devised a scoring function, retrieval is trivially accomplished by choosing an ordered set of top-t QA pairs from $\mathcal{D}$ in accordance with their $f(\cdot, \cdot)$ scores.

3.1 Evaluation

In the datasets that we use, we have labels indicating which QAs are related/relevant to a particular question. Thus, the quality of the scoring function can be evaluated using traditional information retrieval metrics (Robertson and Zaragoza, 2007) such as Precision, MAP, MRR, and NDCG when measured against such labellings. In addition, we will use one more metric, namely **Success Rate**, *the fraction of questions for which at least one related question is ranked among the top-t*, in evaluation.

4 LASER-QA: The Proposed Technique

Our method, LASER-QA, embeds QA pairs in $\mathcal{D}$, within a unified space of desired dimensionality.

$$\{(\boldsymbol{q}_1, \boldsymbol{a}_1), \ldots, (\boldsymbol{q}_n, \boldsymbol{a}_n)\} \xrightarrow{LASER-QA} \{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n\}$$

where, $\boldsymbol{u}_i \in \mathbb{R}^d$ is a vector space embedding in the latent space $\mathbb{R}^d$. As we will illustrate, LASER-QA targets to preserve the local similarity structures in the question and answer spaces within the unified embedding. Having built the embedding of QA pairs, cosine similarity between vectors in $\mathbb{R}^d$ is used for scoring:

$$f(\boldsymbol{q}, (\boldsymbol{q}_i, \boldsymbol{a}_i)) = \frac{\boldsymbol{u}^\top \boldsymbol{u}_i}{\|\boldsymbol{u}\| \|\boldsymbol{u}_i\|}, \quad (1)$$

where, $\boldsymbol{u} \in \mathbb{R}^d$ is the embedding of the new question $\boldsymbol{q}$; we will outline the embedding of single questions into $\mathbb{R}^d$ in a later section.

Our motivation behind LASER-QA stems from the idea of Local Linear Embedding (LLE) (Saul and Roweis, 2000); further, the choice of local neighborhood preservation is motivated by pervasive usage of local neighbors (i.e., k-NN retrieval) in case-based reasoning frameworks (De Mantaras et al., 2005) that seek to reuse structured problem-solution data.

4.1 Data Representation

We use the *tf-idf* vector representation for each question (denoted as $\boldsymbol{x}_i$) and each answer ($\boldsymbol{y}_i$) in $\mathcal{D}$. The *tf-idf* vectors are in $\mathbb{R}^D$ where D denotes the size of the vocabulary. The question and answer tf-idf vectors are arranged as columns to form matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$, both of size $D \times n$. Recall, the *latent space* would be a Euclidean space of dimension d, and typically, we have $d < D$. Our method is intentionally designed to not rely on the specifics of the representation used, and thus can make use of any vector representation of text data. Note that our latent space embeddings in $\mathbb{R}^d$ are evidently unrelated to distributional text embeddings (e.g., (Mikolov et al., 2013)) and are complementary in that such embeddings could be used as an alternative input representation for $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$.

4.2 Regularized Reconstruction Coefficients

For any question $\boldsymbol{x}_i$, let $\mathcal{N}_k(\boldsymbol{x}_i)$ denote the set of top-k nearest questions to the question $\boldsymbol{x}_i$, proximity assessed using cosine similarity of vectors in $\mathbb{R}^D$; analogously, $\mathcal{N}_k(\boldsymbol{y}_i)$ denotes the top-k nearest answers to $\boldsymbol{y}_i$. Much like the representation, the similarity measure may also be replaced as appropriate. Inspired by LLE (Saul and Roweis, 2000), we model the local neighborhood geometry around $\boldsymbol{x}_i$ using *reconstruction coefficient* w^q_{ij} for each question $\boldsymbol{x}_j \in \mathcal{N}_k(\boldsymbol{x}_i)$. We intend to learn the co-efficients such that $\boldsymbol{x}_i$ may be reconstructed well as a linear combination of the neighbors using the co-efficients. Thus, these co-efficients are computed by minimizing, for every question

$\boldsymbol{x}_i$, the *regularized reconstruction penalty (RRP)* given below:

$$\text{RRP}(\boldsymbol{x}_i) = \frac{1}{2}\left\|\boldsymbol{x}_i - \sum_{\boldsymbol{x}_j \in \mathcal{N}_k(\boldsymbol{x}_i)} w_{ij}^q \boldsymbol{x}_j\right\|^2 + \frac{\lambda}{2}\sum_{\boldsymbol{x}_j \in \mathcal{N}_k(\boldsymbol{x}_i)} \left(w_{ij}^q\right)^2 \quad (2)$$

The first term denotes the approximation error in reconstructing $\boldsymbol{x}_i$ as a linear combination of its k nearest neighbors using weights w_{ij}^q. The second term is an L2 regularization term weighted with a non-negative hyperparameter λ, which we set to 0.01 in our experiments. We replaced the sum-to-one constraint in (Saul and Roweis, 2000) by L2 regularization since the former produces large swings in magnitude on either sides of 0.0 (note co-efficients are not constrained to be non-negative) on high-dimensional spaces such as our tf-idf space, leading to stability concerns.

By explicitly assigning $w_{ij}^q = 0 \; \forall \boldsymbol{x}_j \notin \mathcal{N}_k(\boldsymbol{x}_i)$, we rewrite the above problem as:

$$\min_{\boldsymbol{w}_i^q} \; \frac{1}{2}\boldsymbol{w}_i^{q\top}\left(\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I}\right)\boldsymbol{w}_i^q - \boldsymbol{w}_i^{q\top}\boldsymbol{X}^\top\boldsymbol{x}_i + \frac{1}{2}\boldsymbol{x}_i^\top\boldsymbol{x}_i \quad \text{subject to} \;\; w_{ij}^q = 0 \; \forall j \notin \mathcal{N}_k(\boldsymbol{x}_i) \quad (3)$$

where $\boldsymbol{I}$ is an $n \times n$ identity matrix and $\boldsymbol{w}_i^q$ is a column vector of size n comprising reconstruction coefficients vector for $\boldsymbol{x}_i$. It can be shown that the nonzero entries of the optimal coefficient vector is:

$$(\boldsymbol{X}_i^\top\boldsymbol{X}_i + \lambda\boldsymbol{I}_k)^{-1}\boldsymbol{X}_i^\top\boldsymbol{x}_i \quad (4)$$

where $\boldsymbol{I}_k$ is an identity matrix of the size k and matrix $\boldsymbol{X}_i$ is a $D \times k$ matrix obtained from the matrix $\boldsymbol{X}$ by retaining only those columns which are neighbors of $\boldsymbol{x}_i$. Note, the above matrix inverse is well-defined since the matrix is *positive definite* by construction. Once we find these optimal coefficient vectors for all questions (answers), we stack them together column-wise and obtain a matrix, $\boldsymbol{W}^q$ ($\boldsymbol{W}^a$) of size $n \times n$, called the reconstruction coefficient matrix for questions (answers). These two matrices $\boldsymbol{W}^q$ and $\boldsymbol{W}^a$ capture the local geometry of the questions and answers in the QA-archive $\mathcal{D}$.

4.3 Embedding into Latent Space $\mathbb{R}^d$

In this step, we use the $\boldsymbol{W}^q$ and $\boldsymbol{W}^a$ matrices to do the transformation of the QA pairs, the $(\boldsymbol{x}_i, \boldsymbol{y}_i)$s to $\boldsymbol{u}_i$s. Building upon LLE, we develop a scheme to preserve the local neighborhood structure around $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ in learning the $\boldsymbol{u}_i$.

$$\min_{\boldsymbol{U}} \; \alpha\sum_{i=1}^{n}\|\boldsymbol{u}_i - \boldsymbol{U}\boldsymbol{w}_i^q\|^2 + (1-\alpha)\sum_{i=1}^{n}\|\boldsymbol{u}_i - \boldsymbol{U}\boldsymbol{w}_i^a\|^2$$
$$\text{subject to:} \;\; \sum_{i=1}^{n}\boldsymbol{u}_i = 0$$
$$\boldsymbol{U}\boldsymbol{U}^\top = (n\text{-}1)\boldsymbol{I}_d \quad (5)$$

where, $\boldsymbol{U}$ is a $d \times n$ matrix whose i^{th} column is equal to $\boldsymbol{u}_i$. $\alpha \in [0, 1]$ is a weighting parameter that allows to trade-off between question and answer spaces. At $\alpha = 1$, the embedding $\boldsymbol{u}_i$ will try to maximally align with question $\boldsymbol{x}_i$ and vice versa. Our constraints, like the analogous ones in LLE, ensure origin-centered mean solutions and avoid degenerate solutions, respectively (Pang et al., 2005). The first constraint is soft in that any optimal solution disregarding the constraint can be shifted to ensure origin-centering.

Towards capturing the optimal solution for Eq. 5, we define three $n \times n$ symmetric matrices,

$$\boldsymbol{Q} = (\boldsymbol{I} - \boldsymbol{W}^q)(\boldsymbol{I} - \boldsymbol{W}^q)^\top \quad (6)$$
$$\boldsymbol{A} = (\boldsymbol{I} - \boldsymbol{W}^a)(\boldsymbol{I} - \boldsymbol{W}^a)^\top \quad (7)$$
$$\boldsymbol{Z} = \alpha\boldsymbol{Q} + (1-\alpha)\boldsymbol{P}, \; \alpha \in [0, 1] \quad (8)$$

Theorem 1. *If the eigenvalues of the matrix $\boldsymbol{Z}$ are arranged in the descending order and the eigenvectors corresponding to the last d eigenvalues are denoted by $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_d\}$, then, the optimal solution for Eq. (5), denoted by $\boldsymbol{U}^*$ is:*

$$\boldsymbol{U}^* = \begin{pmatrix} \boldsymbol{v}_1^\top \\ \boldsymbol{v}_2^\top \\ \cdots \\ \boldsymbol{v}_d^\top \end{pmatrix} \quad (9)$$

Further, origin centering is achieved by the following transformation:

$$\boldsymbol{U}^*_{centered} = \boldsymbol{U}^* - \boldsymbol{U}^*\boldsymbol{e}\boldsymbol{e}^\top \quad (10)$$

where $\boldsymbol{e}$ is a a vector of all $1's$.

Proof: First, observe that the objective function of Eq. (5) can be rewritten in a compact form:

$$\alpha \left\| \boldsymbol{U} - \boldsymbol{U}\boldsymbol{W}^q \right\|_F^2 + (1-\alpha) \left\| \boldsymbol{U} - \boldsymbol{U}\boldsymbol{W}^a \right\|_F^2 \quad (11)$$

where, $\| \cdot \|_F$ denotes the Frobenius norm. Now, keeping the first constraint aside, we fold in the second constraint using Lagrange multipliers yielding the following Lagrangian $L(\boldsymbol{U}, \boldsymbol{\Lambda})$.

$$L(\boldsymbol{U}, \boldsymbol{\Lambda}) = \alpha \left\| \boldsymbol{U} \text{-} \boldsymbol{U}\boldsymbol{W}^q \right\|_F^2 + (1\text{-}\alpha) \left\| \boldsymbol{U} \text{-} \boldsymbol{U}\boldsymbol{W}^a \right\|_F^2 + \boldsymbol{e}^\top \left(\boldsymbol{\Lambda} \circ \left(\boldsymbol{U}\boldsymbol{U}^\top \text{-} (n\text{-}1)\boldsymbol{I}_d \right) \right) \boldsymbol{e} \quad (12)$$

where $\boldsymbol{e}$ is an all 1's vector. In this Lagrangian,

- Matrix $\boldsymbol{\Lambda}$ is a $d \times d$ symmetric matrix denoting the Lagrange multipliers for the second constraint. Note, the last term is a compact representation of $d^2/2$ equality constraints.
- The symbol $\circ$ denotes the Hadamard products (element wise product) of two matrices.

For any matrix $\boldsymbol{M}$, we have $\|\boldsymbol{M}\|_F^2 = \mathrm{Tr}(\boldsymbol{M}\boldsymbol{M}^\top)$ where $\mathrm{Tr}(.)$ is the trace. Thus, we can rewrite the first two terms of Eq.(12) as:

$$\alpha \, \mathrm{Tr}\left((\boldsymbol{U} - \boldsymbol{U}\boldsymbol{W}^q)(\boldsymbol{U} - \boldsymbol{U}\boldsymbol{W}^q)^\top \right) + (1-\alpha) \, \mathrm{Tr}\left((\boldsymbol{U} - \boldsymbol{U}\boldsymbol{W}^a)(\boldsymbol{U} - \boldsymbol{U}\boldsymbol{W}^a)^\top \right)$$

A slight re-arrangement yields:

$$\alpha \, \mathrm{Tr}\left(\boldsymbol{U}(\boldsymbol{I} - \boldsymbol{W}^q)(\boldsymbol{I} - \boldsymbol{W}^q)^\top \boldsymbol{U}^\top \right) + (1-\alpha) \, \mathrm{Tr}\left(\boldsymbol{U}(\boldsymbol{I} - \boldsymbol{W}^a)(\boldsymbol{I} - \boldsymbol{W}^a)^\top \boldsymbol{U}^\top \right)$$

The Q and A space components are now separated out into the first and second terms. We now simplify the notation using Eq. (6) and (7) to:

$$L(\boldsymbol{U}, \boldsymbol{\Lambda}) = \alpha \, \mathrm{Tr}\left(\boldsymbol{U}\boldsymbol{Q}\boldsymbol{U}^\top \right) + (1\text{-}\alpha) \, \mathrm{Tr}\left(\boldsymbol{U}\boldsymbol{A}\boldsymbol{U}^\top \right) + \boldsymbol{e}^\top \left(\boldsymbol{\Lambda} \circ \left(\boldsymbol{U}\boldsymbol{U}^\top \text{-} (n\text{-}1)\boldsymbol{I}_d \right) \right) \boldsymbol{e}$$

Recall the following for any matrices $\boldsymbol{A}, \boldsymbol{B}, \& \, \boldsymbol{C}$.

1. $\boldsymbol{A} \circ (\boldsymbol{B} - \boldsymbol{C}) = (\boldsymbol{A} \circ \boldsymbol{B}) - (\boldsymbol{A} \circ \boldsymbol{C})$
2. $\boldsymbol{e}^\top (\boldsymbol{A} \circ \boldsymbol{B}) \, \boldsymbol{e} = \mathrm{Tr}(\boldsymbol{A}\boldsymbol{B}^\top) = \mathrm{Tr}(\boldsymbol{A}^\top \boldsymbol{B})$

This allows us to rewrite Eq. (13) as:

$$L(\boldsymbol{U}, \boldsymbol{\Lambda}) = \mathrm{Tr}\left(\alpha \boldsymbol{U}\boldsymbol{Q}\boldsymbol{U}^\top + (1-\alpha)\boldsymbol{U}\boldsymbol{A}\boldsymbol{U}^\top + \boldsymbol{\Lambda}\boldsymbol{U}\boldsymbol{U}^\top - (n-1)\boldsymbol{\Lambda} \right) \quad (13)$$

To find an optimal $\boldsymbol{U}$, we differentiate $L(\boldsymbol{U}, \boldsymbol{\Lambda})$ w.r.t $\boldsymbol{U}$ and equate to zero. This leads to:

$$\frac{\partial L(\boldsymbol{U}, \boldsymbol{\Lambda})}{\partial \boldsymbol{U}} = 2\alpha \boldsymbol{U}\boldsymbol{Q} + 2(1-\alpha)\boldsymbol{U}\boldsymbol{A} + 2\boldsymbol{\Lambda}\boldsymbol{U} = 0$$

The above follows from standard matrix properties (Petersen and Pedersen, 2012). Re-arranging:

$$(\alpha \boldsymbol{Q} + (1-\alpha)\boldsymbol{A})\, \boldsymbol{U}^\top = -\boldsymbol{U}^\top \boldsymbol{\Lambda} \quad (14)$$

One possible solution of the above equation could be constructed in the following manner.

1. Let $\boldsymbol{Z} = \alpha \boldsymbol{Q} + (1-\alpha)\boldsymbol{A}$
2. Compute the Eigen decomposition of $\boldsymbol{Z}$
3. Find the lowest (i.e., bottom) d Eigen values, and take the corresponding Eigen vectors.
4. Form a matrix $\boldsymbol{U}$ by stacking the selected Eigen vectors row-wise.

While any subset of d eigenvectors (and their eigenvalues) would be a solution for Eq. (14), we would take the bottom d eigenvectors for minimizing the objective; this is so since the objective becomes $\mathrm{Tr}(-\boldsymbol{\Lambda})$ when Eq. (14) holds. The matrix constructed above is the optimal $\boldsymbol{U}^*$ in Eq. (9). This completes the proof. ■

The first constraint in Eq.(5) is then applied to centre the vectors around the origin using Eq.(10).

4.4 Embedding a new Question in $\mathbb{R}^d$

To use the historical $\boldsymbol{u}_i$ vectors to retrieve historical QAs against a new question (vector) $\boldsymbol{q}$, we need to embed the latter in the same space $\mathbb{R}^d$. This is achieved using the same structure as applied in forming the embedding; we start with identifying, from $\mathcal{D}$, the k-nearest questions to $\boldsymbol{q}$. The reconstruction co-efficient vector $\boldsymbol{w}^q$ is then learnt using Eq. (2). Finally, we obtain the embedding $\boldsymbol{u}$ for $\boldsymbol{x}$ as a $\boldsymbol{w}^q$-weighted linear combination of the $\mathbb{R}^d$ embeddings corresponding to the k-nearest neighbors. This is captured in steps 9-11 in Algorithm 1 given in the next section.

Algorithm 1: LASER-QA Algorithm

input : $\mathcal{D} = \{(\boldsymbol{q}_1, \boldsymbol{a}_1), \ldots, (\boldsymbol{q}_n, \boldsymbol{a}_n)\}$ (CQA corpus) & query $\boldsymbol{q}$

output : Top-t relevant QA pairs from $\mathcal{D}$

Offline Phase

1 Use appropriate data representation to form vector-pairs $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ for every QA $(\boldsymbol{q}_i, \boldsymbol{a}_i)$;

2 Compute the reconstruction coefficient matrices $\boldsymbol{W}^q$ and $\boldsymbol{W}^a$ (Ref. Section 4.2);

3 $\boldsymbol{Q} \leftarrow (\boldsymbol{I} - \boldsymbol{W}^q)(\boldsymbol{I} - \boldsymbol{W}^q)^\top$;

4 $\boldsymbol{A} \leftarrow (\boldsymbol{I} - \boldsymbol{W}^a)(\boldsymbol{I} - \boldsymbol{W}^a)^\top$;

5 $\boldsymbol{Z} \leftarrow \alpha\boldsymbol{Q} + (1-\alpha)\boldsymbol{A}$;

6 $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_d\} \leftarrow$ Bottom d eigenvectors of matrix $\boldsymbol{Z}$;

7 $\boldsymbol{U}^* \leftarrow \begin{pmatrix} \boldsymbol{v}_1^\top \\ \boldsymbol{v}_2^\top \\ \ldots \\ \boldsymbol{v}_d^\top \end{pmatrix}$;

8 $\boldsymbol{U}^*_{\text{centered}} = \boldsymbol{U}^* - \boldsymbol{U}^*\boldsymbol{e}\boldsymbol{e}^\top$;

Query-time (Online) Phase

9 $\boldsymbol{x} \leftarrow$ Vector representation of the query $\boldsymbol{q}$;

10 $\boldsymbol{w}_x \leftarrow$ Vector of size n capturing the reconstruction coefficients for $\boldsymbol{x}$;

11 $\boldsymbol{u} \leftarrow \boldsymbol{U}^*_{\text{centered}}\boldsymbol{w}_x$;

12 Output top-t QA pairs based by computing the following scores

$$f(\boldsymbol{q}, (\boldsymbol{q}_i, \boldsymbol{a}_i)) = \frac{\boldsymbol{u}^\top \boldsymbol{u}_i}{\|\boldsymbol{u}\|\|\boldsymbol{u}_i\|}$$

4.5 LASER-QA Algorithm

The details of the LASER-QA technique from the previous sections are summarized in Algorithm 1, with the offline (Steps 1-8) and query-time phase (Steps 9-12) clearly demarcated. It may be noted that, LASER-QA, being an optimization-based method, preserves Q/A-space local neighborhoods on a best-effort basis and does not offer guarantees on the fraction of local neighbors preserved from either spaces in the $\mathbb{R}^d$ embedding.

4.5.1 Generalizability of LASER-QA

LASER-QA can be easily extended to incorporate other kinds of information that might be available along with QA pairs such as images, votes (e.g., Blurtit[5], Quora and Yahoo! Answers) tags (Quora), categories (answers.com[6] and Yahoo! Answers) or comments (Quora and Blurtit). Consider data in the form of triplets $(\boldsymbol{q}_i, \boldsymbol{a}_i, \boldsymbol{m}_i)$ where $\boldsymbol{m}_i$ represents the extra information. The $\boldsymbol{m}_i$ vectors are subjected to the same form of processing as $\boldsymbol{q}_i$ and $\boldsymbol{a}_i$ vectors, leading to the $\boldsymbol{W}^m$ and $\boldsymbol{M}$ matrices. Line 5 in Algorithm 1 would then change to:

$$\boldsymbol{Z} \leftarrow \alpha_q\boldsymbol{Q} + \alpha_a\boldsymbol{A} + \alpha_m\boldsymbol{M} \tag{15}$$

where the different αs denote interpolation weights that need to be set appropriately. The remainder of the LASER-QA steps remain identical to Algorithm 1. It may be noted that α_m could be set to a low value if the utility of the extra information is deemed to be low.

4.5.2 Scalability of LASER-QA

We now analyze the scalability of LASER-QA, separately analysing the (a) one-time offline phase, and (b) query-time phase.

Offline Phase: This is a one-time operation at the system design time, involving matrix multiplications followed by eigen-decomposition. Our matrices being sparse, multiplications are fast and worst-case quadratic[7] in n. The Eigendecomposition is $\mathcal{O}(n^3)$, but being a fundamental matrix operation, very efficient implementations exist (especially for symmetric matrices such as ours) with sub-second response times for n of the order of thousands (in packages such as Eigen[8] and LAPACK[9]), trendlines illustrating that Eigendecompositions with even n of the order of millions are easy. The embeddings of all vectors are then indexed using conventional multi-dimensional indexes and/or locality sensitive hashing to aid querying.

Online/Query-time Phase: This encompasses (a) an IR query to find the k most similar questions, (b) solving for the k reconstruction co-efficients in Eq. 4 and forming the embedding, and (c) simply querying for top-t nearest neighbors over indexes built at design-time. The main query-time overhead (vis-a-vis conventional information retrieval) is the additional query over the multi-dimensional index; this construction ensures fast sub-linear response times for the online phase.

Scalability against other methods: In contrast to LASER-QA, it is notable that the baselines

[5] http://www.blurtit.com/

[6] http://www.answers.com/

[7] https://goo.gl/RQ1m0V

[8] https://goo.gl/phMJv9

[9] https://goo.gl/rJjBY6 (Fig 3.1)

employ expensive query-time operations; specifically, it is unclear as to how query-time retrieval using Eq.4 in the TBLM paper (Zhang et al., 2014) and Eq.1 in Topic-TRLM paper (Zhou et al., 2015) could be completed in sub-linear time.

5 Experimental Evaluation

5.1 Datasets and Experimental Setup

Datasets: We use two recent datasets in our evaluation, CQADupStack (Hoogeveen et al., 2015) and SemEval2016-Task3 (Nakov et al., 2016b). The former has a manually labelled set of *related* questions to every question, whereas the latter has relevance labels associated with answers (appearing as *comments*); these labellings make automated evaluation possible. Among the 12 subsets in CQADupStack, owing to scalability issues of the AENN baseline, we choose the three smaller subsets from CQADupStack, namely *webmasters* (1299 QAs), *android* (2193), and *gis* (3726) for a full comparative evaluation. Each of these are split into two halves, with one portion used for the training (that is, learning the statistical model such as LASER-QA, translation model, etc.) and the other one used for the testing (the 50:50 split ensures a sizeable test set). The *related* labellings are used only for evaluation purposes; however, since only training pairs are retrieved within this setup, *related* labellings across QAs in the testing set would be missed, artificially lowering the recall of all the methods in our evaluation. In a recent analysis (Hoogeveen et al., 2016), CQADupStack authors quantify the incompleteness of labeling in the dataset. Such issues further artificially reduce retrieval accuracies as estimated from our automated evaluation. The SemEval2016 dataset, on the other hand, has an implicit test-train split. We use the subset of the data categorized under *Qatar Living Lounge*, the largest category (which is 27% of the full dataset), for our experiments. All 'comments' that are labelled relevant to the associated question are paired together as QA-pairs to form a training set of 1366 pairs, with the test questions from the dataset used as is.

Baselines: As detailed in Section 2, we compare against three baselines (a) TBLM (Zhang et al., 2014) (topic model approach), (b) Topic-TRLM (Zhou et al., 2015) (topic+translation models), and (b) AENN (Zhou et al., 2016) (deep learning). TBLM requires an *answer quality signal* that we set to unity. We use author-recommended parameter settings for TBLM and TopicTRLM. Since AENN learns a latent space representation (though a separate one for questions and answers unlike LASER-QA), the evaluation w.r.t LASER-QA is a direct comparison of the quality of the respective latent spaces. The AENN method requires training triplets, i.e., [question, answer, other answer]; we populate the *other answer* part using the answer of a related question. This gives AENN an advantage as it uses relations among training pairs that are unavailable to other methods. For AENN, quality measures peaked around 2000 (for *webmasters* and *gis*) and 3000 (for *android* and *SemEval2016*) for latent space dimensionality; our results are from such settings.

LASER-QA Parameters: We set $k = 15$ and $\alpha = 0.8$, the latter ensuring that the question space is given more importance. We always set d to the number of eigen vectors in $\boldsymbol{Z}$, equalling $|\mathcal{D}|$. We will separately study LASER-QA trends against parameter variations as well.

Evaluation Metrics: We use Precision, Success Rate (SR) (Ref. Sec 3), MAP and NDCG (Robertson and Zaragoza, 2007) for our evaluation. Precision simply measures the fraction of related documents among the top-t that were retrieved. Due to this rank-agnostic construction, precision is unable to incentivize for putting the relevant results at the top of the result instead of deeper down. In contrast, MAP and NDCG are rank-aware metrics. MAP[10] computes the average of precisions computed at rank positions where a relevant result is returned. NDCG is another rank-aware metric[11] that discounts the appearance of the revelant result based on it's rank in the result set. We assess statistical significance using randomization tests (Smucker et al., 2007).

5.2 Evaluation Results and Insights

Table 2 summarizes the comparative evaluation across varying t (best results boldfaced). The following observations are notable:

- LASER-QA outperforms the other methods across datasets. This is followed by Topic-TRLM, TBLM and then AENN.
- LASER-QA's margin is highest at (small) values of t that are typical of scenarios involving human perusal of results. As t in-

[10] https://goo.gl/xr7NnD
[11] https://goo.gl/26Pcct

Table 2: Retrieval Results (• & ∘ denote statistical significance at p-value < 0.01 & < 0.05 respectively)

Dataset →	webmasters (#QAs=1299)				android (#QAs=2193)				gis (#QAs=3726)				SemEval2016 (#QAs=1366)			
	t=5				t=5				t=5				t=5			
Method	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG
LASER-QA	**0.022•**	**0.101•**	**0.079•**	**0.082•**	**0.026∘**	**0.127•**	**0.094•**	**0.102•**	**0.023•**	**0.111•**	**0.085•**	**0.089•**	**0.141∘**	**0.407•**	**0.266∘**	**0.265∘**
TBLM	0.009	0.043	0.031	0.034	0.017	0.080	0.059	0.061	0.008	0.041	0.034	0.035	0.084	0.259	0.158	0.158
TTRLM	0.012	0.056	0.030	0.033	0.022	0.101	0.061	0.067	0.016	0.079	0.048	0.052	0.067	0.201	0.134	0.139
AENN	0.008	0.035	0.023	0.026	0.007	0.033	0.012	0.012	0.005	0.026	0.016	0.017	0.030	0.148	0.060	0.059
	t=10				t=10				t=10				t=10			
Method	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG
LASER-QA	**0.013•**	**0.116•**	**0.080•**	**0.088•**	**0.015∘**	**0.148•**	**0.097•**	**0.110•**	**0.013•**	**0.123•**	**0.086•**	**0.095•**	**0.125•**	**0.556•**	**0.291•**	**0.292•**
TBLM	0.006	0.050	0.031	0.036	0.010	0.089	0.060	0.066	0.005	0.046	0.034	0.037	0.070	0.296	0.156	0.163
TTRLM	0.010	0.093	0.036	0.043	0.014	0.124	0.065	0.076	0.010	0.098	0.050	0.060	0.069	0.383	0.152	0.160
AENN	0.005	0.043	0.024	0.029	0.005	0.047	0.014	0.018	0.003	0.030	0.016	0.019	0.036	0.259	0.069	0.078
	t=20				t=20				t=20				t=20			
Method	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG
LASER-QA	**0.007**	**0.121**	**0.080•**	**0.092•**	**0.008**	**0.157**	**0.097•**	**0.116•**	**0.007**	**0.126**	**0.087•**	**0.100•**	**0.079**	**0.605**	**0.296•**	**0.327•**
TBLM	0.003	0.050	0.031	0.038	0.006	0.100	0.060	0.070	0.003	0.052	0.034	0.038	0.051	0.333	0.157	0.175
TTRLM	0.006	0.118	0.037	0.052	**0.008**	0.149	0.066	0.085	0.006	0.119	0.052	0.067	0.059	0.519	0.149	0.182
AENN	0.003	0.053	0.025	0.032	0.003	0.066	0.015	0.023	0.002	0.039	0.017	0.021	0.023	0.321	0.074	0.104
	t=50				t=50				t=50				t=50			
Method	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG	Prec	SR	MAP	NDCG
LASER-QA	**0.003**	0.125	**0.081•**	**0.096•**	0.003	0.166	**0.097•**	**0.121•**	0.002	0.129	**0.087•**	**0.103•**	0.033	0.630	**0.295•**	**0.359•**
TBLM	0.001	0.056	0.031	0.040	0.003	0.112	0.060	0.074	0.001	0.060	0.035	0.040	0.026	0.346	0.151	0.187
TTRLM	**0.003**	**0.145∘**	0.038	0.061	**0.004∘**	**0.177**	0.066	0.093	**0.003•**	**0.152•**	0.052	0.074	**0.039**	**0.667**	0.140	0.214
AENN	0.002	0.070	0.025	0.035	0.002	0.088	0.015	0.028	0.001	0.065	0.017	0.025	0.016	0.407	0.066	0.107

Dataset	#QAs	t=5 Prec	t=5 SR	t=5 MAP	t=5 NDCG	t=50 Prec	t=50 SR	t=50 MAP	t=50 NDCG
stats	4004	**0.016•**	**0.076•**	**0.057•**	**0.060•**	0.002	0.096	**0.058•**	**0.071•**
programmers	4107	**0.020•**	**0.096•**	**0.068•**	**0.075•**	0.002	0.115	**0.069•**	**0.088•**
wordpress	4744	**0.019•**	**0.091•**	**0.069•**	**0.074•**	0.002	0.112	**0.070•**	**0.085•**
physics	5044	**0.025•**	**0.120•**	**0.088•**	**0.094•**	0.003	0.148	**0.090•**	**0.111•**
mathematica	5084	**0.018•**	**0.087•**	**0.067•**	**0.072•**	0.002	0.116	**0.069•**	**0.084•**
unix	5330	**0.023•**	**0.115•**	**0.089•**	**0.094•**	0.003	0.137	**0.091•**	**0.107•**
gaming	6398	**0.034•**	**0.166•**	**0.130•**	**0.137•**	0.004	0.189	**0.132•**	**0.155•**
english	6668	**0.024•**	**0.115•**	**0.090•**	**0.095•**	**0.003**	**0.130**	**0.092•**	**0.107•**

Table 3: LASER-QA Results (Boldfacing and Statistical Significance indications from comparison with TopicTRLM and TBLM) over Larger Categories in CQADupStack

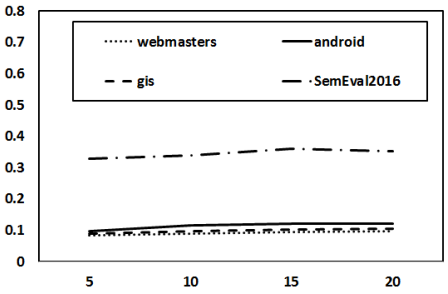

Figure 1: NDCG (Y-axis) v/s. k

creases way beyond the training neighborhood size (i.e., 15), LASER-QA is seen to deteriorate gracefully (as expected).

- LASER-QA performance peaks on rank-aware metrics such as MAP and NDCG (even at $t = 50$), indicating it's high effectiveness in producing relevant results at the top.

These observations underline the effectiveness of LASER-QA as a CQA retrieval method. It may be noted that LASER-QA uses compact representations ($d < 2000$), as compared to vocabulary space representations that are typically ≥ 5000.

Trends at High t: The performance trends at high values of t are explained by the usage of the local neighborhood (of size k) in LASER-QA latent space learning. Going down the result list much beyond k reveals expected, but graceful, decline in accuracy. For automated processing scenarios that necessitate large t, a correspondingly high k may be used in learning. It is notable that LASER-QA's focus on local neighborhood manifests as enhanced accuracy at the top of the result set.

LASER-QA Analysis on Larger CQADupStack Datasets: Owing to scalability issues of AENN that disallows a full evaluation over larger categories in CQADupStack, we present LASER-QA results over them in Table 3 to illustrate the consistency in trends. Boldfacing and statistical significance have the same semantics as earlier, with the comparison performed against only TopicTRLM and TBLM.

5.3 LASER-QA Parameter Analysis

We now analyze the NDCG trends (NDCG being the most popular IR metric) across LASER-QA parameters, i.e., k, α and d, varying each one separately keeping the default choice for others.

- **Varying k:** Figure 1 plots NDCG against values of k from $\{5, 10, 15, 20\}$. As may be seen, the accuracy is seen to improve with increasing k in the lower ranges, while sat-

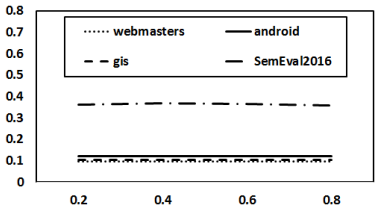

Figure 2: NDCG (Y-axis) v/s varying values of α

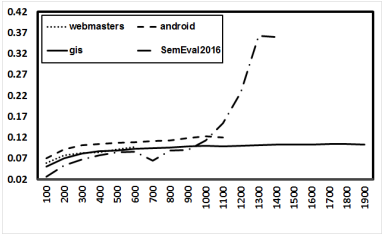

Figure 3: NDCG (Y-axis) v/s varying values of d

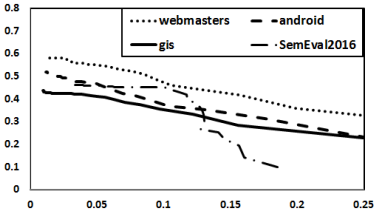

Figure 4: Precision (Y-axis) v/s Recall (X-axis)

urating beyond 15. The trends are seen to be similar across datasets.

- **Varying α:** The retrieval accuracies were seen to be stable across a wide range of values of α such as illustrated in Figure 2. This shows that LASER-QA is not very sensitive to α.

- **Varying d:** The size of the latent space, d, forms a critical parameter for LASER-QA. Given the LASER-QA construction, this space is limited by the number of eigenvectors in the matrix $\boldsymbol{Z}$ which is $n \times n$. This means, d is limited above by n, the size of the training dataset. Table 3 plots the accuracies with varying values of d, with the upper end different for different datasets due to the dependence on the training dataset size. The plots indicate that the performance improves steadily with increasing values of d. The performance saturates beyond 400 for the topically coherent CQADupStack datasets. The *Qatar Living Lounge* category in SemEval2016, unlike the CQADupStack categories, is more diverse discussing issues ranging from massage centres to immigration. Thus, LASER-QA is able to make use of much more dimensions to model the complexity involved.

To summarize, LASER-QA is not very sensitive to α and is best run with $k \geq 15$ and values of $d \approx n$.

Finally, the precision-recall curve with varying values of t is presented in Figure 4. As may be observed, LASER-QA exhibits a gradual degradation of precision with increasing t correlated with a corresponding improvement in recall. The diversity in the SemEval2016 dataset manifests as a sharper precision drop at high t, as the result set starts to transcend sub-topic boundaries.

6 Conclusions

We considered the problem of CQA retrieval – the task of retrieving relevant historical QA pairs in response to a new question. We formulated a method that builds upon the ideas from local linear embedding to use collective corpus level information across historical QA pairs to embed them in a latent space. In contrast to the mainstream paradigm in literature, we do not explicitly model lexical correlations; instead, we learn an embedding of QA pairs in a way that the local neighborhood in question and answer spaces are preserved. LASER-QA provides a single-model based solution in lieu of learning separate models (e.g., topic and translation models) which are then interpolated to a final scoring; the latter approach has been observed to have reproducibility issues (Qiu et al., 2013). We analyzed our method empirically against state of the art methods from across families of CQA retrieval methods that use topic models, translation models and deep learning. Our empirical results confirm that the LASER-QA method significantly outperforms the baselines on all IR metrics of interest, underlining the effectiveness of our modelling.

Future Work: A study on the correlation between the kNNs in the LASER-QA embedded space and the original Question and Answer spaces would provide insights into the extent of correlation between manifolds in the original spaces. Further, we would like to see how LASER-QA generalizes to beyond text; one immediate scenario of interest is to explore how pictures and multimedia within QAs may be leveraged within LASER-QA.

Acknowledgments

A part of this work for Dinesh Garg was supported by the SERB-DST Early Career Research Grant No. ECR/2016/002035.

References

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

Dasha Bogdanova, Cícero Nogueira dos Santos, Luciano Barbosa, and Bianca Zadrozny. 2015. Detecting semantically equivalent questions in online user forums. In *CoNLL*, volume 123, page 2015.

Peter F Brown, John Cocke, Stephen A Della Pietra, Vincent J Della Pietra, Fredrick Jelinek, John D Lafferty, Robert L Mercer, and Paul S Roossin. 1990. A statistical approach to machine translation. *Computational linguistics*, 16(2):79–85.

Li Cai, Guangyou Zhou, Kang Liu, and Jun Zhao. 2011. Learning the latent topics for question retrieval in community QA. In *IJCNLP*, volume 11, pages 273–281.

Xin Cao, Gao Cong, Bin Cui, Christian Søndergaard Jensen, and Ce Zhang. 2009. The use of categorization information in language models for question retrieval. In *CIKM*, pages 265–274. ACM.

Arpita Das, Harish Yenala, Manoj Kumar Chinnakotla, and Manish Shrivastava. 2016. Together we stand: Siamese networks for similar question retrieval. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Ramon Lopez De Mantaras, David McSherry, Derek Bridge, David Leake, Barry Smyth, Susan Craw, Boi Faltings, Mary Lou Maher, MICHAEL T COX, Kenneth Forbus, et al. 2005. Retrieval, reuse, revision and retention in case-based reasoning. *The Knowledge Engineering Review*, 20(03):215–240.

P Deepak. 2016. Mixkmeans: Clustering question-answer archives. In *EMNLP*, pages 1576–1585.

P Deepak, Sutanu Chakraborti, and Deepak Khemani. 2013. Query suggestions for textual problem solution repositories. In *ECIR*, pages 569–581. Springer.

P Deepak and Karthik Visweswariah. 2014. Unsupervised solution post identification from discussion forums. In *ACL (1)*, pages 155–164.

Huizhong Duan, Yunbo Cao, Chin-Yew Lin, and Yong Yu. 2008. Searching questions by identifying question topic and question focus. In *ACL*, pages 156–164.

Doris Hoogeveen, Karin M Verspoor, and Timothy Baldwin. 2015. CQADupStack: A benchmark data set for community question-answering research. In *Proceedings of the 20th Australasian Document Computing Symposium*, page 3. ACM.

Doris Hoogeveen, Karin M Verspoor, and Timothy Baldwin. 2016. Cqadupstack: Gold or silver?

Haifeng Hu, Bingquan Liu, Baoxun Wang, Ming Liu, and Xiaolong Wang. 2013. Multimodal DBN for predicting high-quality answers in cQA portals. In *ACL (2)*, pages 843–847.

Zongcheng Ji, Fei Xu, Bin Wang, and Ben He. 2012. Question-answer topic model for question retrieval in community question answering. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2471–2474. ACM.

Jung-Tae Lee, Sang-Bum Kim, Young-In Song, and Hae-Chang Rim. 2008. Bridging lexical gaps between queries and questions on large online Q&A collections with compact translation models. In *EMNLP*, pages 410–418.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119.

Preslav Nakov, Lluís Màrquez, and Francisco Guzmán. 2016a. It takes three to tango: Triangulation approach to answer ranking in community question answering. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1586–1597.

Preslav Nakov, Lluís Màrquez, Alessandro Moschitti, Walid Magdy, Hamdy Mubarak, Abed Alhakim Freihat, Jim Glass, and Bilal Randeree. 2016b. Semeval-2016 task 3: Community question answering. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 525–545.

Yanwei Pang, Lei Zhang, Zhengkai Liu, Nenghai Yu, and Houqiang Li. 2005. Neighborhood preserving projections (NPP): A novel linear dimension reduction method. In *ICIC*, pages 117–125.

Jae Hyun Park and W Bruce Croft. 2015. Using key concepts in a translation model for retrieval. In *SIGIR*, pages 927–930. ACM.

K.B. Petersen and M. S. Pedersen. 2012. The matrix cookbook.

Xipeng Qiu, Le Tian, and Xuanjing Huang. 2013. Latent semantic tensor indexing for community-based question answering. In *ACL (2)*, pages 434–439.

Stephen Robertson and Hugo Zaragoza. 2007. On rank-based effectiveness measures and optimization. *Information Retrieval*, 10(3):321–339.

Lawrence K Saul and Sam T Roweis. 2000. An introduction to locally linear embedding. *unpublished. Available at: http://www. cs. toronto. edu/~ roweis/lle/publications. html*.

Yikang Shen, Wenge Rong, Zhiwei Sun, Yuanxin Ouyang, and Zhang Xiong. 2015. Question/answer matching for CQA system via combining lexical and sequential information. In *AAAI*.

Anna Shtok, Gideon Dror, Yoelle Maarek, and Idan Szpektor. 2012. Learning from the past: answering new questions with past answers. In *WWW*, pages 759–768. ACM.

Amit Singh. 2012. Entity based Q&A retrieval. In *EMNLP-CoNLL*, pages 1266–1277. Association for Computational Linguistics.

Mark D. Smucker, James Allan, and Ben Carterette. 2007. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 623–632, New York, NY, USA. ACM.

Baoxun Wang, Bingquan Liu, Xiaolong Wang, Chengjie Sun, and Deyuan Zhang. 2011. Deep learning approaches to semantic relevance modeling for chinese question-answer pairs. *ACM Transactions on Asian Language Information Processing (TALIP)*, 10(4):21.

Xiaobing Xue, Jiwoon Jeon, and W Bruce Croft. 2008. Retrieval models for question and answer archives. In *SIGIR*, pages 475–482. ACM.

Kai Zhang, Wei Wu, Haocheng Wu, Zhoujun Li, and Ming Zhou. 2014. Question retrieval with high quality answers in community question answering. In *CIKM*, pages 371–380.

Weinan Zhang, Zhaoyan Ming, Yu Zhang, Ting Liu, and Tat-Seng Chua. 2015. Exploring key concept paraphrasing based on pivot language translation for question retrieval. In *AAAI*, pages 410–416.

Guangyou Zhou, Li Cai, Jun Zhao, and Kang Liu. 2011. Phrase-based translation model for question retrieval in community question answer archives. In *ACL(1)*, pages 653–662.

Guangyou Zhou, Yubo Chen, Daojian Zeng, and Jun Zhao. 2014. Group non-negative matrix factorization with natural categories for question retrieval in community question answer archives. In *COLING*, pages 89–98.

Guangyou Zhou, Yin Zhou, Tingting He, and Wensheng Wu. 2016. Learning semantic representation with neural networks for community question answering retrieval. *Knowledge-Based Systems*, 93:75–83.

Tom Chao Zhou, Michael Rung-Tsong Lyu, Irwin King, and Jie Lou. 2015. Learning to suggest questions in social media. *Knowledge and Information Systems*, 43(2):389–416.