Document Similarity for Texts of Varying Lengths via Hidden Topics

Hongyu Gong[*] **Tarek Sakakini**[*] **Suma Bhat**[*] **Jinjun Xiong** [†]
[*]University of Illinois at Urbana-Champaign, USA
[†]T. J. Watson Research Center, IBM
[*]{hgong6, sakakini, spbhat2}@illinois.edu [†]jinjun@us.ibm.com

Abstract

Measuring similarity between texts is an important task for several applications. Available approaches to measure document similarity are inadequate for document pairs that have non-comparable lengths, such as a long document and its summary. This is because of the lexical, contextual and the abstraction gaps between a long document of rich details and its concise summary of abstract information. In this paper, we present a document matching approach to bridge this gap, by comparing the texts in a common space of hidden topics. We evaluate the matching algorithm on two matching tasks and find that it consistently and widely outperforms strong baselines. We also highlight the benefits of the incorporation of domain knowledge to text matching.

1 Introduction

Measuring the similarity between documents is of key importance in several natural processing applications including information retrieval (Salton and Buckley, 1988), book recommendation (Gopalan et al., 2014), news categorization (Ontrup and Ritter, 2002) and essay scoring (Landauer, 2003). A range of document similarity approaches have been proposed and effectively used in recent applications including (Lai et al., 2015; Bordes et al., 2015). Central to the tasks discussed above is the assumption that the documents being compared are of comparable lengths.

Advances in language processing approaches to transform natural language understanding, such as text summarization and recommendation, have generated new requirements for comparing documents. For instance, summarization techniques (extractive and abstractive) are capable of automatically generating textual summaries by converting a long document of several hundred words into a condensed text of only a few words while preserving the core meaning of the original text (Kedzie and McKeown, 2016). Conceivably, a related aspect of summarization is the task of bidirectional matching of a summary and a document or a set of documents, which is the focus of this study. The document similarity considered in this paper is between texts that have significant differences not only in length, but also in the abstraction level (such as a definition of an abstract concept versus a detailed instance of that abstract concept).

Table 1: A Sample Concept-Project Matching

Concept
Heredity: Inheritance and Variation of Traits
All cells contain genetic information in the form of DNA molecules. Genes are regions in the DNA that contain the instructions that code for the formation of proteins.
Project
Pedigree Analysis: A Family Tree of Traits
Do you have the same hair color or eye color as your mother? When we look at members of a family it is easy to see that some physical characteristics or traits are shared. To start this project, you should draw a pedigree showing the different members of your family. Ideally you should include multiple people from at least three generations.

As an illustration, consider the task of matching a **Concept** with a **Project** as shown in Table 1. Here a **Concept** is a grade-level science curriculum item and represents the summary. A **Project**, listed in a collection of science projects, represents the document. Projects typically are long texts including an introduction, materials and procedures, whereas science concepts are much shorter in comparison having a title and a concise and abstract description. The concepts and projects are described in detail in Section 5.1. The matching

Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Long Papers), pages 2341–2351
Melbourne, Australia, July 15 - 20, 2018. ©2018 Association for Computational Linguistics

task here is to automatically suggest a hands-on project for a given concept in the curriculum, such that the project can help reinforce a learner's basic understanding of the concept. Conversely, given a science project, one may need to identify the concept it covers by matching it to a listed concept in the curriculum. This would be conceivable in the context of an intelligent tutoring system.

Challenges to the matching task mentioned above include: 1) The mismatch in the relative lengths of the documents being compared – a long piece of text (henceforth termed *document*) and a short piece of text (termed *summary*) – gives rise to the vocabulary mismatch problem, where the document and the summary do not share a majority of terms. 2) The context mismatch problem arising because a document provides a reasonable amount of text to infer the contextual meaning of a term, but a summary only provides a limited context, which may or may not involve the same terms considered in the document. These challenges render existing approaches to comparing documents–for instance, those that rely on document representations (e.g., Doc2Vec (Le and Mikolov, 2014))–inadequate, because the predominance of non-topic words in the document introduces noise to its representation while the summary is relatively noise-free, rendering Doc2Vec inadequate for comparing them.

Our approach to the matching problem is to allow a multi-view generalization of the document, where multiple hidden topics are used to establish a common ground to capture as much information of the document and the summary as possible and use this to score the relevance of the pair. We empirically validate our approach on two tasks – that of project-concept matching in grade-level science and that of scientific paper-summary matching – using both custom-made and publicly available datasets. The main contributions of this paper are:

1. We propose an embedding-based hidden topic model to extract topics and measure their importance in long documents.
2. We present a novel geometric approach to compare documents with differing modality (a long document to a short summary) and validate its performance relative to strong baselines.
3. We explore the use of domain-specific word embeddings for the matching task and show the explicit benefit of incorporating domain knowledge in the algorithm.
4. We make available the first dataset[1] on project-concept matching in the science domain to help further research in this area.

2 Related Works

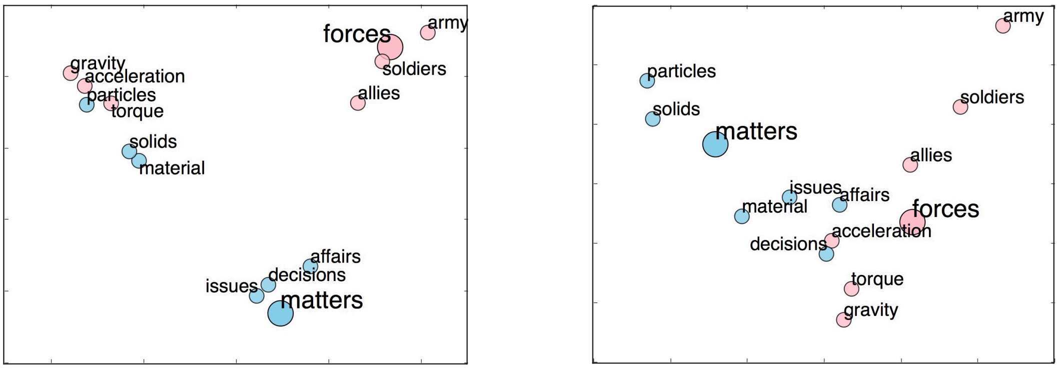
Document similarity approaches quantify the degree of relatedness between two pieces of texts of comparable lengths and thus enable matching between documents. Traditionally, statistical approaches (e.g., (Metzler et al., 2007)) and vector-space-based methods (including the robust Latent Semantic Analysis (LSA) (Dumais, 2004)) have been used for text similarity. More recently, neural network-based methods have been used for document representation and these include average word embeddings (Mikolov et al., 2013), Doc2Vec (Le and Mikolov, 2014), Skip-Thought vectors (Kiros et al., 2015), recursive neural network-based methods (Socher et al., 2014), LSTM architectures (Tai et al., 2015), and convolutional neural networks (Blunsom et al., 2014).

Considering works that avoid using an explicit document representation for comparing documents, the state-of-the-art method is Word Mover's Distance (WMD), which relies on pre-trained word embeddings (Kusner et al., 2015). Given these embeddings, the WMD defines the distance between two documents as the best transport cost of moving all words from one document to another within the space of word embeddings. The advantages of WMD are that it is hyper-parameter free and achieves high retrieval accuracy on document classification tasks with documents of comparable lengths. However, it is computationally expensive for long documents (Kusner et al., 2015).

Clearly, what is lacking in prior literature is a study of document similarity approaches that match documents with widely different sizes. It is this gap in literature that we expect to fill by way of this study.

Latent Variable Models. Latent variable models including count-based and probabilistic models have been studied in many previous works. Count-based models such as Latent Semantic Indexing (LSI) compare two documents based on their combined vocabulary (Deerwester et al., 1990). When

[1] Our code and data are available at: https://github.com/HongyuGong/Document-Similarity-via-Hidden-Topics.git



(a) word geometry of general embedding

(b) word geometry of science domain embeddings

Figure 1: Two key words "forces" and "matters" are shown in red and blue respectively. Red words represent different senses of "forces", and blue words carry senses of "matters". "forces" mainly refers to "army" and "matters" refers to "issues" in general embedding of (a), whereas "forces" shows its sense of "gravity" and "matters" shows the sense of "solids" in science-domain embedding of (b)

documents have highly mismatched vocabularies such as those that we study, relevant documents might be classified as irrelevant. Our model is built upon word-embeddings which is more robust to such a vocabulary mismatch.

Probabilistic models such as Latent Dirichlet Analysis (LDA) define topics as distributions over words (Blei et al., 2003). In our model, topics are low-dimensional real-valued vectors (more details in Section 4.2).

3 Domain Knowledge

Domain information pertaining to specific areas of knowledge is made available in texts by the use of words with domain-specific meanings or senses. Consequently, domain knowledge has been shown to be critical in many NLP applications such as information extraction and multi-document summarization (Cheung and Penn, 2013a), spoken language understanding (Chen et al., 2015), aspect extraction (Chen et al., 2013) and summarization (Cheung and Penn, 2013b).

As will be described later, our distance metric for comparing a document and a summary relies on word embeddings. We show in this work, that embeddings trained on a science-domain corpus lead to better performance than embeddings on the general corpus (WikiCorpus). Towards this, we extract a science-domain sub-corpus from the WikiCorpus, and the corpus extraction will be detailed in Section 5.

To motivate the domain-specific behavior of polysemous words, we will qualitatively explore how domain-specific embeddings differ from the general embeddings on two polysemous science terms: *forces* and *matters*. Considering the fact that the meaning of a word is dictated by its neighbors, for each set of word embeddings, we plot the neighbors of these two terms in Figure 1 on to 2 dimensions using Locally Linear Embedding (LLE), which preserves word distances (Roweis and Saul, 2000). We then analyze the sense of the focus terms–here, *forces* and *matters*.

From Figure 1(a), we see that for the word *forces*, its general embedding is close to *army*, *soldiers*, *allies* indicating that it is related with violence and power in a general domain. Shifting our attention to Figure 1(b), we see that for the same term, its science embedding is closer to *torque*, *gravity*, *acceleration* implying that its science sense is more about physical interactions. Likewise, for the word *matters*, its general embedding is surrounded by *affairs* and *issues*, whereas, its science embedding is closer to *particles* and *material*, prompting that it represents substances. Thus, we conclude that domain specific embeddings (here, science), is capable of incorporating domain knowledge into word representations. We use this observation in our document-summary matching system to which we turn next.

4 Model

Our model that performs the matching between document and summary is depicted in Figure 2. It is composed of three modules that perform preprocessing, document topic generation, and relevance measurement between a document and a summary. Each of these modules is discussed below.

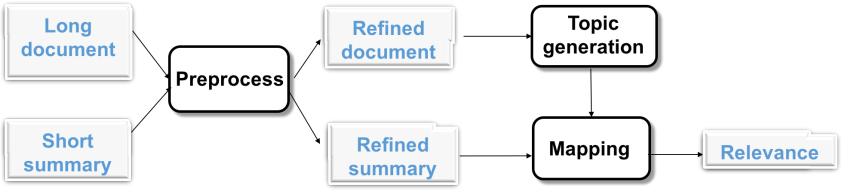


Figure 2: The system for document-summary matching

4.1 Preprocessing

The preprocessing module tokenizes texts and removes stop words and prepositions. This step allows our system to focus on the content words without impacting the meaning of original texts.

4.2 Topic Generation from Documents

We assume that a document (a long text) is a structured collection of words, with the 'structure' brought about by the composition of topics. In some sense, this 'structure' is represented as a set of hidden topics. Thus, we assume that a document is generated from certain hidden "topics", analogous to the modeling assumption in LDA. However, unlike in LDA, the "topics" here are neither specific words nor the distribution over words, but are are essentially a set of *vectors*. In turn, this means that words (represented as vectors) constituting the document structure can be generated from the hidden topic vectors.

Introducing some notation, the word vectors in a document are $\{\mathbf{w}_1, \dots, \mathbf{w}_n\}$, and the hidden topic vectors of the document are $\{\mathbf{h}_1, \dots, \mathbf{h}_K\}$, where $\mathbf{w}_i, \mathbf{h}_k \in \mathbb{R}^d$, $d = 300$ in our experiments.

Linear operations using word embeddings have been empirically shown to approximate their compositional properties (e.g. the embedding of a phrase is nearly the sum of the embeddings of its component words) (Mikolov et al., 2013). This motivates the linear reconstruction of the words from the document's hidden topics while minimizing the reconstruction error. We stack the K topic vectors as a topic matrix $\mathbf{H} = [\mathbf{h}_1, \dots, \mathbf{h}_K](K < d)$. We define the reconstructed word vector $\tilde{\mathbf{w}}_i$ for the word $\mathbf{w}_i$ as the optimal linear approximation given by topic vectors: $\tilde{\mathbf{w}}_i = \mathbf{H}\tilde{\boldsymbol{\alpha}}_i$, where

$$\tilde{\boldsymbol{\alpha}}_i = \underset{\boldsymbol{\alpha}_i \in \mathbb{R}^K}{\operatorname{argmin}} \|\mathbf{w}_i - \mathbf{H}\boldsymbol{\alpha}_i\|_2^2. \tag{1}$$

The reconstruction error E for the whole document is the sum of each word's reconstruction error and is given by: $E = \sum_{i=1}^{n} \|\mathbf{w}_i - \tilde{\mathbf{w}}_i\|_2^2$. This being a function of the topic vectors, our goal is to find the optimal $\mathbf{H}^*$ so as to minimize the error E:

$$\begin{aligned} \mathbf{H}^* &= \underset{\mathbf{H} \in \mathbb{R}^{d \times K}}{\operatorname{argmin}}\ E(\mathbf{H}) \\ &= \underset{\mathbf{H} \in \mathbb{R}^{d \times K}}{\operatorname{argmin}} \sum_{i=1}^{n} \min_{\boldsymbol{\alpha}_i} \|\mathbf{w}_i - \mathbf{H}\boldsymbol{\alpha}_i\|_2^2, \end{aligned} \tag{2}$$

where $\|\cdot\|$ is the Frobenius norm of a matrix.

Without loss of generality, we require the topic vectors $\{\mathbf{h}_i\}_{i=1}^K$ to be orthonormal, i.e., $\mathbf{h}_i^T \mathbf{h}_j = 1_{(i=j)}$. As we can see, the optimization problem (2) describes an optimal linear space spanned by the topic vectors, so the norm and the linear dependency of the vectors do not matter. With the orthonormal constraints, we simplify the form of the reconstructed vector $\tilde{\mathbf{w}}_i$ as:

$$\tilde{\mathbf{w}}_i = \mathbf{H}\mathbf{H}^T \mathbf{w}_i. \tag{3}$$

We stack word vectors in the document as a matrix $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_n]$. The equivalent formulation to problem (2) is:

$$\begin{aligned} \min_{\mathbf{H}} \quad & \|\mathbf{W} - \mathbf{H}\mathbf{H}^T\mathbf{W}\|_2^2 \\ \text{s.t.} \quad & \mathbf{H}^T\mathbf{H} = \mathbf{I}, \end{aligned} \tag{4}$$

where $\mathbf{I}$ is an identity matrix.

The problem can be solved by Singular Value Decomposition (SVD), using which, the matrix $\mathbf{W}$ can be decomposed as $\mathbf{W} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$, where $\mathbf{U}^T\mathbf{U} = \mathbf{I}, \mathbf{V}^T\mathbf{V} = \mathbf{I}$, and $\boldsymbol{\Sigma}$ is a diagonal matrix where the diagonal elements are arranged in a decreasing order of absolute values. We show in the supplementary material that the first K vectors in the matrix $\mathbf{U}$ are exactly the solution to $\mathbf{H}^* = [\mathbf{h}_1^*, \dots, \mathbf{h}_K^*]$.

We find optimal topic vectors $\mathbf{H}^* = [\mathbf{h}_1^*, \dots, \mathbf{h}_K^*]$ by solving problem (4). We note that these topic vectors are not equally important, and we say that one topic is more important than another if it can reconstruct words with smaller error. Define E_k as the reconstruction error when we only use topic vector $\mathbf{h}_k^*$ to reconstruct the document:

$$E_k = \|\mathbf{W} - \mathbf{h}_k^* \mathbf{h}_k^{*T} \mathbf{W}\|_2^2. \tag{5}$$

Now define i_k as the *importance* of topic $\mathbf{h}_k^*$, which measures the topic's ability to reconstruct the words in a document:

$$i_k = \|\mathbf{h}_k^{*T} \mathbf{W}\|_2^2 \tag{6}$$

We show in the supplementary material that the higher the importance i_k is, the smaller the reconstruction error E_k is. Now we normalize i_k as $\bar{i}_k$ so that the importance does not scale with the norm of the word matrix W, and so that the importances of the K topics sum to 1. Thus,

$$\bar{i}_k = i_k / (\sum_{j=1}^{K} i_j). \tag{7}$$

The number of topics K is a hyperparameter in our model. A small K may not cover key ideas of the document, whereas a large K may keep trivial and noisy information. Empirically we find that $K = 15$ captures most important information from the document.

4.3 Topic Mapping to Summaries

We have extracted K topic vectors $\{\mathbf{h}_k^*\}_{k=1}^K$ from the document matrix $\mathbf{W}$, whose importance is reflected by $\{\bar{i}_k\}_{k=1}^K$. In this module, we measure the relevance of a document-summary pair. Towards this, a summary that matches the document should also be closely related with the “topics” of that document. Suppose the vectors of the words in a summary are stacked as a $d \times m$ matrix $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_m]$, where $\mathbf{s}_j$ is the vector of the j-th word in a summary. Similar to the reconstruction of the document, the summary can also be reconstructed from the documents' topic vectors as shown in Eq. (3). Let $\tilde{\mathbf{s}}_j^k$ be the reconstruction of the summary word $\mathbf{s}_j$ given by one topic $\mathbf{h}_k^*$: $\tilde{\mathbf{s}}_j^k = \mathbf{h}_k^* \mathbf{h}_k^{*T} \mathbf{s}_j$.

Let $r(\mathbf{h}_k^*, \mathbf{s}_j)$ be the relevance between a topic vector $\mathbf{h}_k^*$ and summary word $\mathbf{s}_j$. It is defined as the cosine similarity between $\tilde{\mathbf{s}}_j^k$ and $\mathbf{s}_j$:

$$r(\mathbf{h}_k^*, \mathbf{s}_j) = \mathbf{s}_j{}^T \tilde{\mathbf{s}}_j^k / (\|\mathbf{s}_j\|_2 \cdot \|\tilde{\mathbf{s}}_j^k\|_2). \tag{8}$$

Furthermore, let $r(\mathbf{h}_k^*, \mathbf{S})$ be the relevance between a topic vector and the summary, defined to be the average similarity between the topic vector and the summary words:

$$r(\mathbf{h}_k^*, \mathbf{S}) = \frac{1}{m} \sum_{j=1}^{m} r(\mathbf{h}_k^*, \mathbf{s}_j). \tag{9}$$

The relevance between a topic vector and a summary is a real value between 0 and 1.

As we have shown, the topics extracted from a document are not equally important. Naturally, a summary relevant to more important topics is more likely to better match the document. Therefore, we define $r(\mathbf{W}, \mathbf{S})$ as the relevance between the document $\mathbf{W}$ and the summary $\mathbf{S}$, and $r(\mathbf{W}, \mathbf{S})$ is the sum of topic-summary relevance weighted by the importance of the topic:

$$r(\mathbf{W}, \mathbf{S}) = \sum_{k=1}^{K} \bar{i}_k \cdot r(\mathbf{h}_k^*, \mathbf{S}), \tag{10}$$

where $\bar{i}_k$ is the importance of topic $\mathbf{h}_k^*$ as defined in (7). The higher $r(\mathbf{W}, \mathbf{S})$ is, the better the summary matches the document.

We provide a visual representation of the documents as shown in Figure 3 to illustrate the notion of hidden topics. The two documents are from science projects: a genetics project, *Pedigree Analysis: A Family Tree of Traits* (ScienceBuddies, 2017a), and a weather project, *How Do the Seasons Change in Each Hemisphere* (ScienceBuddies, 2017b). We project all embeddings to a three-dimensional space for ease of visualization.

As seen in Figure 3, the hidden topics reconstruct the words in their respective documents to the extent possible. This means that the words of a document lie roughly on the plane formed by their corresponding topic vectors. We also notice that the summary words (*heredity* and *weather* respectively for the two projects under consideration) lie very close to the plane formed by the hidden topics of the relevant project while remaining away from the plane of the irrelevant project. This shows that the words in the summary (and hence the summary itself) can also be reconstructed from the hidden topics of documents that match the summary (and are hence ‘relevant’ to the summary). Figure 3 visually explains the geometric relations between the summaries, the hidden topics and the documents. It also validates the representation power of the extracted hidden topic vectors.

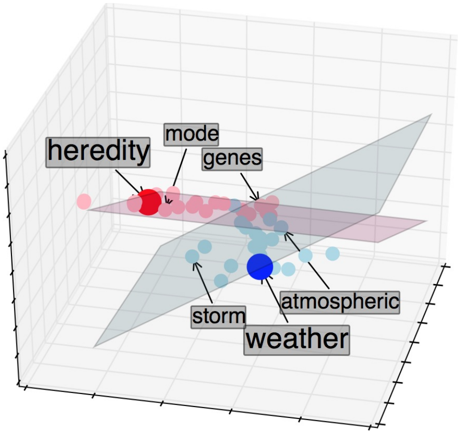


Figure 3: Words *mode* and *genes* from the document on genetics and words *storm* and *atmospheric* from document on weather are represented by pink and blue points respectively. Linear space of hidden topics in genetics form the pink plane, where summary word *heredity* (the red point) roughly lies. Topic vectors of the document on weather form the blue plane, and the summary word *weather* (the darkblue point) lies almost on the same plane.

5 Experiments

In this section, we evaluate our document-summary matching approach on two specific applications where texts of different sizes are compared. One application is that of concept-project matching useful in science education and the other is that of summary-research paper matching.

Word Embeddings. Two sets of 300-dimension word embeddings were used in our experiments. They were trained by the Continuous Bag-of-Words (CBOW) model in word2vec (Mikolov et al., 2013) but on different corpora. One training corpus is the full English WikiCorpus of size 9 GB (Al-Rfou et al., 2013). The second consists of science articles extracted from the WikiCorpus. To extract these science articles, we manually selected the science categories in Wikipedia and considered all subcategories within a depth of 3 from these manually selected root categories. We then extracted all articles in the aforementioned science categories resulting in a science corpus of size 2.4 GB. The word vectors used for documents and summaries are both from the pretrained word2vec embeddings.

Baselines We include two state-of-the-art methods of measuring document similarity for comparison using their implementations available in gensim (Řehůřek and Sojka, 2010).

(1) **Word movers' distance (WMD)** (Kusner et al., 2015). WMD quantifies the distance between a pair of documents based on word embeddings as introduced previously (c.f. Related Work). We take the negative of their distance as a measure of document similarity (here between a document and a summary).

(2) **Doc2Vec** (Le and Mikolov, 2014). Document representations have been trained with neural networks. We used two versions of doc2vec: one trained on the full English Wikicorpus and a second trained on the science corpus, same as the corpora used for word embedding training. We used the cosine similarity between two text vectors to measure their relevance.

For a given document-summary pair, we compare the scores obtained using the above two methods with that obtained using our method.

5.1 Concept-Project matching

Science projects are valuable resources for learners to instigate knowledge creation via experimentation and observation. The need for matching a science concept with a science project arises when learners intending to delve deeper into certain concepts search for projects that match a given concept. Additionally, they may want to identify the concepts with which a set of projects are related.

We note that in this task, science concepts are highly concise summaries of the core ideas in projects, whereas projects are detailed instructions of the experimental procedures, including an introduction, materials and a description of the procedure, as shown in Table 1. Our matching method provides a way to bridge the gap between abstract concepts and detailed projects. The format of the concepts and the projects is discussed below.

Concepts. For the purpose of this study we use the concepts available in the Next Generation Science Standards (NGSS) (NGSS, 2017). Each concept is accompanied by a short description. For example, one concept in life science is *Heredity: Inheritance and Variation of Traits*. Its description is *All cells contain genetic information in the form of DNA molecules. Genes are regions in the DNA that contain the instructions that code for the formation of proteins.* Typical lengths of concepts are around 50 words.

Projects. The website Science Buddies (Science-Buddies, 2017c) provides a list of projects from a variety of science and engineering disciplines such

Table 2: Classification results for the Concept-Project Matching task. All performance differences were statistically significant at $p = 0.01$.

method	topic_science	topic_wiki	wmd_science	wmd_wiki	doc2vec_science	doc2vec_wiki
precision	$\mathbf{0.758 \pm 0.012}$	0.750 ± 0.009	0.643 ± 0.070	0.568 ± 0.055	0.615 ± 0.055	0.661 ± 0.084
recall	$\mathbf{0.885 \pm 0.071}$	0.842 ± 0.010	0.735 ± 0.119	0.661 ± 0.119	0.843 ± 0.066	0.737 ± 0.149
fscore	$\mathbf{0.818 \pm 0.028}$	0.791 ± 0.007	0.679 ± 0.022	0.595 ± 0.020	0.695 ± 0.019	0.681 ± 0.032

as physical sciences, life sciences and social sciences. A typical project consists of an abstract, an introduction, a description of the experiment and the associated procedures. A project typically has more than 1000 words.

Dataset. We prepared a representative dataset 537 pairs of projects and concepts involving 53 unique concepts from NGSS and 230 unique projects from Science Buddies. Engineering undergraduate students annotated each pair with the decision whether it was a good match or not and received research credit. As a result, each concept-project pair received at least three annotations, and upon consolidation, we considered a concept-project pair to be a good match when a majority of the annotators agreed. Otherwise it was not considered a good match. The ratio between good matches and bad matches in the collected data was $44 : 56$.

Classification Evaluation. Annotations from students provided the ground truth labels for the classification task. We randomly split the dataset into tuning and test instances with a ratio of $1 : 9$. A threshold score was tuned on the tuning data, and concept-project pairs with scores higher than this threshold were classified as a good matches during testing. We performed 10-fold cross validation, and report the average precision, recall, F1 score and their standard deviation in Table 2.

Our topic-based metric is denoted as "topic", and the general-domain and science-domain embeddings are denoted as "wiki" and "science" respectively. We show the performance of our method against the two baselines while varying the underlying embeddings, thus resulting in 6 different combinations. For example, "topic_science" refers to our method with science embeddings. From the table (column 1) we notice the following: 1) Our method *significantly outperforms* the two baselines by a wide margin (≈10%) in both the general domain setting as well as the domain-specific setting. 2) Using science domain-specific word embeddings instead of the general word embeddings results in the *best performance across all algorithms*. This performance was observed despite the word embeddings being trained on a *significantly smaller* corpus compared to the general domain corpus.

Besides the classification metrics, we also evaluated the directed matching from concepts to projects with ranking metrics.

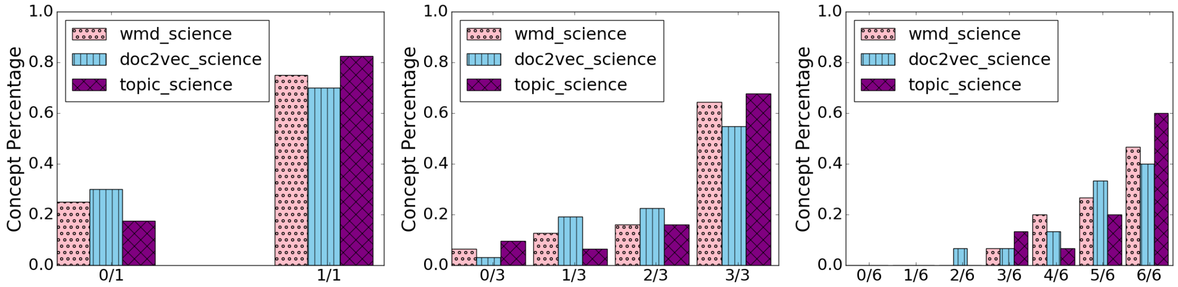
Ranking Evaluation Our collected dataset resulted in having a many-to-many matching between concepts and projects. This is because the same concept was found to be a good match for multiple projects and the same project was found to match many concepts. The previously described classification task evaluated the bidirectional concept-project matching. Next we evaluated the directed matching from concepts to projects, to see how relevant these top ranking projects are to a given input concepts. Here we use precision@k (Radlinski and Craswell, 2010) as the evaluation metric, considering the percentage of relevant ones among top-ranking projects.

For this part, we only considered the methods using science domain embeddings as they have shown superior performance in the classificaiton task. For each concept, we check the precision@k of matched projects and place it in one of k+1 bins accordingly. For example, for k=3, if only two of the three top projects are a correct match, the concept is placed in the bin corresponding to $2/3$. In Figure 4, we show the percentage of concepts that fall into each bin for the three different algorithms for k=1,3,6.

We observe that recommendations using the hidden topic approach fall more in the high value bin compared to others, performing consistently better than two strong baselines. The advantage becomes more obvious at precision@6. It is worth mentioning that wmd_science falls behind doc2vec_science in the classification task while it outperforms in the ranking task.

5.2 Text Summarization

The task of matching summaries and documents is commonly seen in real life. For example, we



(a) Precision@1

(b) Precision@3

(c) Precision@6

Figure 4: Ranking Performance of All Methods

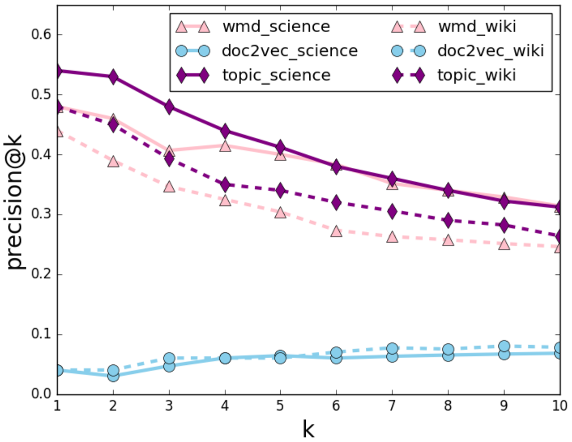


Figure 5: Summary-Article Matching

use an event summary "Google's AlphaGo beats Korean player Lee Sedol in Go" to search for relevant news, or use the summary of a scientific paper to look for related research publications. Such matching constitutes an ideal task to evaluate our matching method between texts of different sizes.

Dataset. We use a dataset from the CL-SciSumm Shared Task (Jaidka et al., 2016). The dataset consists of 730 ACL Computational Linguistics research papers covering 50 categories in total. Each category consists of a reference paper (RP) and around 10 citing papers (CP) that contain citations to the RP. A human-generated summary for the RP is provided and we use the 10 CP as being relevant to the summary. The matching task here is between the summary and all CPs in each category.

Evaluation. For each paper, we keep all of its content except the sections of experiments and acknowledgement (these sections were omitted because often their content is often less related to the topic of the summary). The typical summary length is about 100 words, while a paper has more than 2000 words. For each topic, we rank all 730 papers in terms of their relevance generated by our method and baselines using both sets of embeddings. For evaluation, we use the information retrieval measure of precision@k, which considers the number of relevant matches in the top-k matchings (Manning et al., 2008). For each combination of the text similarity approaches and embeddings, we show precision@k for different k's in Figure 5. We observe that our method with science embedding achieves the best performance compared to the baselines, once again showing not only the benefits of our method but also that of incorporating domain knowledge.

6 Discussion

Analysis of Results. From the results of the two tasks we observe that our method outperforms two strong baselines. The reason for WMD's poor performance could be that the many uninformative words (those unrelated to the central topic) make WMD overestimate the distance between the document-summary pair. As for doc2vec, its single vector representation may not be able to capture all the key topics of a document. A project could contain multifaceted information, e.g., a project to study how climate change affects grain production is related to both environmental science and agricultural science.

Effect of Topic Number. The number of hidden topics K is a hyperparameter in our setting. We empirically evaluate the effect of topic number in the task of concept-project mapping. Figure 6 shows the F1 scores and the standard deviations at different K. As we can see, optimal K is 18. When K is too small, hidden topics are too few to capture key information in projects. Thus we can see that the increase of topic number from 3 to 6 brings a big improvement to the performance. Topic numbers larger than the optimal value degrade the performance since more topics incorporate noisy information. We note that the perfor-

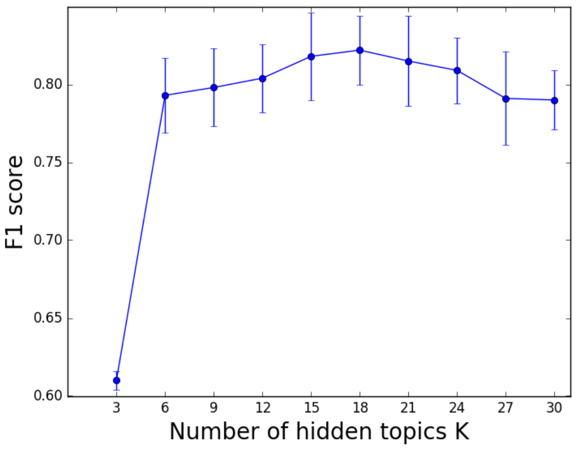


Figure 6: F1 score on concept-project matching with different topic numbers K

mance changes are mild when the number of topics are in the range of [18, 31]. Since topics are weighted by their importance, the effect of noisy information from extra hidden topics is mitigated.

Interpretation of Hidden Topics. We consider the summary-paper matching as an example with around 10 papers per category. We extracted the hidden topics from each paper, reconstructed words with these topics as shown in Eq. (3), and selected the words which had the smallest reconstruction errors. These words are thus closely related to the hidden topics, and we call them *topic words* to serve as an interpretation of the hidden topics. We visualize the cloud of such topic words on the set of papers about word sense disambiguation as shown in Figure 7. We see that the words selected based on the hidden topics cover key ideas such as *disambiguation*, *represent*, *classification* and *sentence*. This qualitatively validates the representation power of hidden topics. More examples are available in the supplementary material.

We interpret this to mean that proposed idea of multiple hidden topics captures the key information of a document. The extracted "hidden topics" represent the essence of documents, suggesting the appropriateness of our relevance metric to measure the similarity between texts of different sizes. Even though our focus in this study was the science domain we point out that the results are more generally valid since we made no domain-specific assumptions.

Varying Sensitivity to Domain. As shown in the results, the science-domain embeddings improved the classification of concept-project matching for the topic-based method by 2% in F1-score, WMD by 8% and doc2vec by 1%, thus underscoring the importance of domain-specific word embeddings.

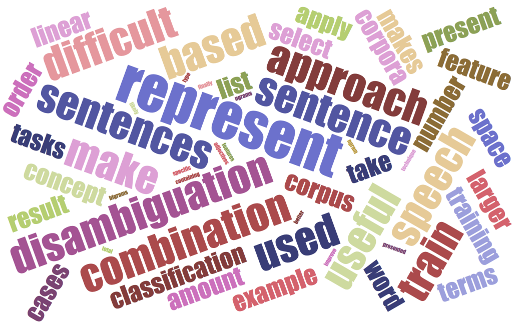


Figure 7: Topic words from papers on word sense disambiguation

Doc2vec is less sensitive to the domain, because it provides document-level representation. Even if some words cannot be disambiguated due to the lack of domain knowledge, other words in the same document can provide complementary information so that the document embedding does not deviate too much from its true meaning.

Our method, also a word embedding method, is not as sensitive to domain as WMD. It is robust to the polysemous words with domain-sensitive semantics, since hidden topics are extracted in the document level. Broader contexts beyond just words provide complementary information for word sense disambiguation.

7 Conclusion

We propose a novel approach to matching documents and summaries. The challenge we address is to bridge the gap between detailed long texts and its abstraction with hidden topics. We incorporate domain knowledge into the matching system to gain further performance improvement. Our approach has beaten two strong baselines in two downstream applications, concept-project matching and summary-research paper matching.

Acknowledgments

This work is supported by IBM-ILLINOIS Center for Cognitive Computing Systems Research (C3SR) - a research collaboration as part of the IBM AI Horizons Network. We thank ACL anonymous reviewers for their constructive suggestions. We thank Mathew Monfort for helping deploy annotation tasks, and thank Jong Yoon Lee for the dataset collection.

References

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual nlp. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192, Sofia, Bulgaria. Association for Computational Linguistics.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Phil Blunsom, Edward Grefenstette, and Nal Kalchbrenner. 2014. A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics.

Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. 2015. Large-scale simple question answering with memory networks. *arXiv preprint arXiv:1506.02075*.

Yun-Nung Chen, William Yang Wang, Anatole Gershman, and Alexander I Rudnicky. 2015. Matrix factorization with knowledge graph propagation for unsupervised spoken language understanding. In *ACL (1)*, pages 483–494.

Zhiyuan Chen, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. 2013. Exploiting domain knowledge in aspect extraction. In *EMNLP*, pages 1655–1667.

Jackie Chi Kit Cheung and Gerald Penn. 2013a. Probabilistic domain modelling with contextualized distributional semantic vectors. In *ACL (1)*, pages 392–401.

Jackie Chi Kit Cheung and Gerald Penn. 2013b. Towards robust abstractive multi-document summarization: A caseframe analysis of centrality and domain. In *ACL (1)*, pages 1233–1242.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391.

Susan T Dumais. 2004. Latent semantic analysis. *Annual review of information science and technology*, 38(1):188–230.

Prem K Gopalan, Laurent Charlin, and David Blei. 2014. Content-based recommendations with poisson factorization. In *Advances in Neural Information Processing Systems*, pages 3176–3184.

Kokil Jaidka, Muthu Kumar Chandrasekaran, Sajal Rustagi, and Min-Yen Kan. 2016. Overview of the cl-scisumm 2016 shared task. In *In Proceedings of Joint Workshop on Bibliometric-enhanced Information Retrieval and NLP for Digital Libraries (BIRNDL 2016)*.

Chris Kedzie and Kathleen McKeown. 2016. Extractive and abstractive event summarization over streaming web text. In *IJCAI*, pages 4002–4003.

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International Conference on Machine Learning*, pages 957–966.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *AAAI*, volume 333, pages 2267–2273.

Thomas K Landauer. 2003. Automatic essay assessment. *Assessment in education: Principles, policy & practice*, 10(3):295–308.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196.

Christopher D Manning, Prabhakar Raghavan, Hinrich Schütze, et al. 2008. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge.

Donald Metzler, Susan Dumais, and Christopher Meek. 2007. Similarity measures for short segments of text. In *European Conference on Information Retrieval*, pages 16–27. Springer.

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

NGSS. 2017. Available at: `https://www.nextgenscience.org`. Accessed: 2017-06-30.

Jorg Ontrup and Helge Ritter. 2002. Hyperbolic self-organizing maps for semantic navigation. In *Advances in neural information processing systems*, pages 1417–1424.

Filip Radlinski and Nick Craswell. 2010. Comparing the sensitivity of information retrieval metrics. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 667–674. ACM.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New*

Challenges for NLP Frameworks, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.

Sam T Roweis and Lawrence K Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523.

ScienceBuddies. 2017a. Available at: https://www.sciencebuddies.org/science-fair-projects/project-ideas/Genom_p010/genetics-genomics/pedigree-analysis-a-family-tree-of-traits. Accessed: 2017-06-30.

ScienceBuddies. 2017b. Available at: https://www.sciencebuddies.org/science-fair-projects/project-ideas/Weather_p006/weather-atmosphere/how-do-the-seasons-change-in-each-hemisphere. Accessed: 2017-06-30.

ScienceBuddies. 2017c. Available at: http://www.sciencebuddies.org. Accessed: 2017-06-30.

Richard Socher, Andrej Karpathy, Quoc V Le, Christopher D Manning, and Andrew Y Ng. 2014. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1556–1566.