Document Segmentation for Labeling with Academic Learning Objectives

Divyanshu Bhartiya
IBM Research
Bangalore, India
dibharti@in.ibm.com

Danish Contractor
IBM Research
New Delhi, India
dcontrac@in.ibm.com

Sovan Biswas
IBM Research
Bangalore, India
sobiswa3@in.ibm.com

Bikram Sengupta
IBM Research
Bangalore, India
bsengupt@in.ibm.com

Mukesh Mohania
IBM Research
Melbourne, Australia
mukeshm@au1.ibm.com

ABSTRACT

Teaching in formal academic environments typically follows a curriculum that specifies learning objectives that need to be met at each phase of a student's academic progression. In this paper, we address the novel task of identifying document *segments* in educational material that are relevant for different learning objectives. Using a dynamic programming algorithm based on a vector space representation of sentences in a document, we automatically segment and then label document segments with learning objectives. We demonstrate the effectiveness of our approach on a real-world education data set. We further demonstrate how our system is useful for related tasks of document passage retrieval and QA using a large publicly available dataset. To the best of our knowledge we are the first to attempt the task of segmenting and labeling education materials with academic learning objectives.

Keywords

text segmentation, document labeling, academic learning objectives, unsupervised

1. INTRODUCTION

The rapid growth of cost-effective smart-phones and media devices, coupled with technologies like Learning Content Management Systems, tutoring systems, digital classrooms, MOOC based eLearning systems etc. are changing the way today's students are educated. A recent survey [1] found that there was a 45% year-on-year uptake between 2013 and 2014 of digital content in the classroom and a nearly 82% uptake in the use of digital textbooks. Of the 400,000 K-12 students surveyed, 37% of them reported using online textbooks for their learning needs. Students and teachers frequently search for free and open education resources available online to augment or replace existing learning material. Organizations like MERLOT[2] and the Open Education Consortium[3] offer and promote the use of free learning resources by indexing material available on the web, based only on keywords or user specified meta-data. This makes the identification of the most relevant resources difficult and time consuming. In addition, the use of manually specified meta-data can also result in poor results due to inconsistent meta-data quality, consistency and coverage. Identifying materials most suitable for a learner can be aided by tagging them with learning objectives from different curricula. However, manually labeling material with learning objectives is not scalable since learning standards can contain tens of hundreds of objectives and are prone to frequent revision. Recent work by [3] attempted to address this problem by using external resources such as Wikipedia to expand the context of learning objectives and a *tf-idf* based vector representation of documents and learning objectives. One of the limitations of the system is that it works well only when documents are relatively short in length and relate to a few learning standard objectives. The accuracy of the algorithm reduces when the documents considered are resources such as textbooks due to the dilution of the weights in the *tf-idf* based vector space model. Further, from the perspective of information access, returning a large reference book for a learning objective still burdens the user with the task of identifying the relevant portions of the book. This, therefore, does not adequately address the problem.

In this paper, we address the problem of finding document segments most relevant to learning objectives, using document segmentation [1] and segment ranking. To the best of our knowledge, we are the first to attempt the problem of segmenting and labeling education materials with academic learning objectives.

In summary, our paper makes the following contributions:

- We define the novel task of identifying and labeling document segments with academic learning objectives.

[1] Project Tomorrow, Trends in Digital Learning 2015

[2] http://www.merlot.org

[3] http://www.oeconsortium.org/

- We present the first system that identifies portions of text most relevant for a learning objective in large educational materials. We demonstrate the effectiveness of our approach on a real world education data set. We report a sentence level $F1$ score of 0.6 and a segment level minimal match accuracy@3 of 0.9
- We demonstrate, using a large publicly available dataset, how our methods can also be used for other NLP tasks such as document passage retrieval and QA.

The rest of the paper is organized as follows: In the next section we describe related work, in section 3 we formally describe our problem statement, section 4 describes our algorithm and implementation details and section 5 presents our detailed experiments. Finally, in section 6 we conclude this paper and discuss possible directions of future work.

2. RELATED WORK

Broadly, our work is related to three major areas of natural language research: Text Segmentation, Query Focused Summarization and Document Passage Retrieval. We present a comparison and discussion for each of these areas below:

Text Segmentation: Typically, the problem of automatically chunking text into smaller meaningful units has been addressed by studying changes in vocabulary patterns [6] and building topic models[5]. In [12], the authors adapt the TextTiling algorithm from [6] to use topics instead of words. Most recently, [1] uses semantic word embeddings for the text segmentation task. While supervised approaches tend to perform better, we decided to adapt the state of the art unsupervised text segmentation method proposed in [1], due to the challenges associated with sourcing training data for supervised learning.

Query Focused Summarization: Focused summarization in our context [8], [10] [4] is the task of building summaries of learning materials based on learning objectives. Here, each learning objective can be treated as a *query*, and the learning materials as documents that need to be summarized. However, it is important to note that in the education domain, any such summarization needs to ensure that summarized material is presented in a way that facilitates learning. This poses additional research challenges such as automatically identifying relationships between concepts presented in the material and therefore, in this paper, we do not model our problem as a summarization task. We encourage the reader to consider it as a possible direction for future research.

Document Passage Retrieval: Lastly, document passage retrieval [2] is the task of fetching relevant document passages from a collection of documents based on a user query. However, such tasks typically require the passage boundaries to be well known and therefore, cannot return sub-portions that may be present within a passage or return results that span sub-parts of multiple passages.

3. PROBLEM STATEMENT

Typically, a learning standard consists of a hierarchical organization of learning objectives where learning objectives are grouped by Topic, Course, Subject and Grade. For the purpose of this paper we refer to a "label" as the complete Grade (g) -> Subject (s) -> Course (c) -> Topic (t) -> Learning Objective (l) path in the learning standard.

Given a document $\mathcal{D}$ of length N we would like to identify the most relevant segments $\phi_{ij}^{\{g,s,c,t,l\}}$ for a given label $\{g, s, c, t, l\}$ where i, j denote positions in a document i.e $i, j \in [0, N]$ and $i < j$. In the rest of the paper, we denote the learning objective $\{g, s, c, t, l\}$ as e to ease notation.

Figure 1 shows chapter 2 from the the "College Physics" OpenStax textbook[4]. The segments (demarcated using rectangles) have been identified for two learning objectives INST1 and INST2 and occur in different portions of the book. They can even be a sub-part of an existing section in a chapter as shown for INST1.

The next section describes our algorithm for the problem of segmentation and labeling based on learning objectives.

4. OUR METHOD

We represent each sentence as a unit vector s_i, $(0 \leq i \leq N-1)$ in a Dim dimensional space. The goal of segmentation is to find K splits in a document, denoted by $(x_0, x_1, \ldots, x_K)$, where $x_0 = 0$ and $x_K = N$ and x_i denotes the line number specifying the segment boundary such that if the kth segment contains the sentence s_i, then $x_{k-1} \leq i < x_k$. The discovered segment $\phi_{i,j}$ is the segment between the splits x_i and x_j . Depending on the granularity of the learning objectives and the document collection, the optimal number of splits can be set (See section 5). Let the cost function ψ for a segment $\psi(i, j)$ measure the *internal cohesion* of the segment, $(0 \leq i < j \leq N)$. The segmentation score for K splits $s = (x_0, x_1, \ldots, x_K)$ can then be defined as Ψ :

$$\Psi(s) = \psi(x_0, x_1) + \psi(x_1, x_2) + \ldots + \psi(x_{K-1}, x_K)$$

To find the optimal splits in the document based on the cost function Ψ, we use dynamic programming. The cost of splitting $\Psi(N, K)$ is the cost of splitting 0 to N sentences using K splits. So,

$$\Psi(N, 1) = \psi(0, N)$$

$$\Psi(N, K) = \min_{l<N} \Psi(l, K-1) + \psi(l, N)$$

We define the ψ function as follows:

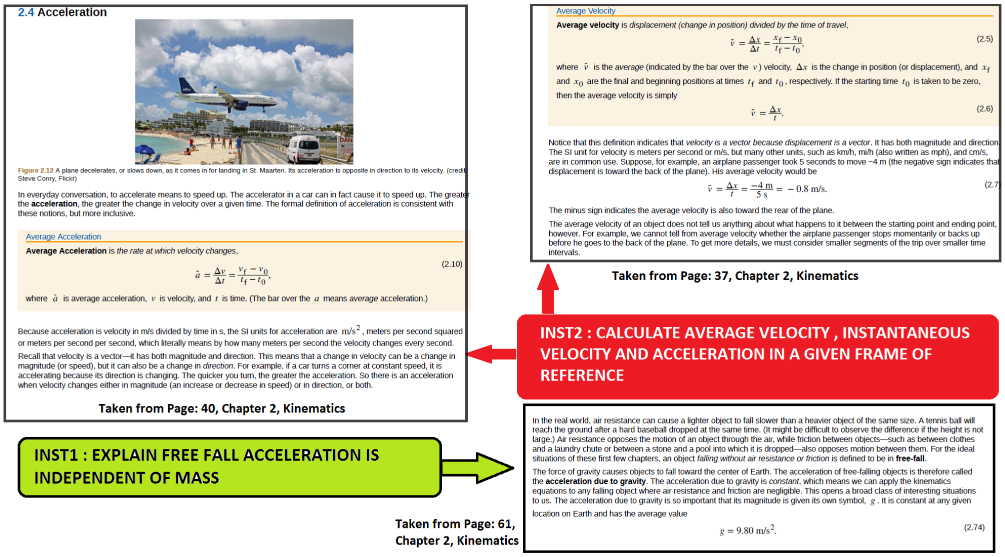
$$\psi(i, j) = \sum_{i \leq l < j} \|s_l - \mu(i, j)\|^2$$

where $\psi(i, j)$ is analogous to the intra-cluster distance in traditional document clustering while $\mu(i, j)$ is a representative vector of the segment. We discuss possible forms of μ later in this section.

Ranking: Each segment is represented as a normalized vector $\mu(i, j)$ and we determine the most relevant segments to a learning objective e by ranking segments in increasing order of similarity based on cosine similarity.

$$\cos(\mu, e) = \sum_{d=1}^{Dim} \mu_d * e_d$$

[4] https://openstax.org/details/college-physics



2.4 Acceleration

Figure 2.12 A plane decelerates, or slows down, as it comes in for landing in St. Maarten. Its acceleration is opposite in direction to its velocity. (credit: Steve Conry, Flickr)

In everyday conversation, to accelerate means to speed up. The accelerator in a car can in fact cause it to speed up. The greater the **acceleration**, the greater the change in velocity over a given time. The formal definition of acceleration is consistent with these notions, but more inclusive.

Average Acceleration

Average Acceleration is *the rate at which velocity changes*,

$$\bar{a} = \frac{\Delta v}{\Delta t} = \frac{v_f - v_0}{t_f - t_0}, \quad (2.10)$$

where $\bar{a}$ is average acceleration, v is velocity, and t is time. (The bar over the a means *average* acceleration.)

Because acceleration is velocity in m/s divided by time in s, the SI units for acceleration are $\mathrm{m/s^2}$, meters per second squared or meters per second per second, which literally means by how many meters per second the velocity changes every second.

Recall that velocity is a vector—it has both magnitude and direction. This means that a change in velocity can be a change in magnitude (or speed), but it can also be a change in *direction*. For example, if a car turns a corner at constant speed, it is accelerating because its direction is changing. The quicker you turn, the greater the acceleration. So there is an acceleration when velocity changes either in magnitude (an increase or decrease in speed) or in direction, or both.

Taken from Page: 40, Chapter 2, Kinematics

Average Velocity

Average velocity is *displacement (change in position) divided by the time of travel*,

$$\bar{v} = \frac{\Delta x}{\Delta t} = \frac{x_f - x_0}{t_f - t_0}, \quad (2.5)$$

where $\bar{v}$ is the *average* (indicated by the bar over the v) velocity, Δx is the change in position (or displacement), and x_f and x_0 are the final and beginning positions at times t_f and t_0, respectively. If the starting time t_0 is taken to be zero, then the average velocity is simply

$$\bar{v} = \frac{\Delta x}{t}. \quad (2.6)$$

Notice that this definition indicates that *velocity is a vector because displacement is a vector*. It has both magnitude and direction. The SI unit for velocity is meters per second or m/s, but many other units, such as km/h, mi/h (also written as mph), and cm/s, are in common use. Suppose, for example, an airplane passenger took 5 seconds to move −4 m (the negative sign indicates that displacement is toward the back of the plane). His average velocity would be

$$\bar{v} = \frac{\Delta x}{t} = \frac{-4\ \mathrm{m}}{5\ \mathrm{s}} = -0.8\ \mathrm{m/s}. \quad (2.7)$$

The minus sign indicates the average velocity is also toward the rear of the plane.

The average velocity of an object does not tell us anything about what happens to it between the starting point and ending point, however. For example, we cannot tell from average velocity whether the airplane passenger stops momentarily or backs up before he goes to the back of the plane. To get more details, we must consider smaller segments of the trip over smaller time intervals.

Taken from Page: 37, Chapter 2, Kinematics

INST2 : CALCULATE AVERAGE VELOCITY , INSTANTANEOUS VELOCITY AND ACCELERATION IN A GIVEN FRAME OF REFERENCE

In the real world, air resistance can cause a lighter object to fall slower than a heavier object of the same size. A tennis ball will reach the ground after a hard baseball dropped at the same time. (It might be difficult to observe the difference if the height is not large.) Air resistance opposes the motion of an object through the air, while friction between objects—such as between clothes and a laundry chute or between a stone and a pool into which it is dropped—also opposes motion between them. For the ideal situations of these first few chapters, an object *falling without air resistance or friction* is defined to be in **free-fall**.

The force of gravity causes objects to fall toward the center of Earth. The acceleration of free-falling objects is therefore called the **acceleration due to gravity**. The acceleration due to gravity is *constant*, which means we can apply the kinematics equations to any falling object where air resistance and friction are negligible. This opens a broad class of interesting situations to us. The acceleration due to gravity is so important that its magnitude is given its own symbol, g. It is constant at any given location on Earth and has the average value

$$g = 9.80\ \mathrm{m/s^2}. \quad (2.74)$$

INST1 : EXPLAIN FREE FALL ACCELERATION IS INDEPENDENT OF MASS

Taken from Page: 61, Chapter 2, Kinematics

Figure 1: This image shows excerpts from chapter 2 Kinematics from the College Physics text book by OpenStax along with the segment boundaries for two learning objectives INST1 and INST2 shown in colors red and green respectively.

We then select the top n ranked segments as the segments relevant to the learning objective. In section 5.3 we describe how the number of splits K as well as the value of n can be chosen empirically given a validation data set.

We now describe different methods of constructing the document and segment vectors:

TF-IDF: Each sentence is represented as a bag of words, the dimensionality being the vocabulary size. Each word in a sentence v_i is weighted by its *tfidf* measure. For a word v_i in the sentence s_k of a document $\mathcal{D}$, the *tfidf* measure is given by :

$$\mathit{tfidf}(v_i)_{s_k,\mathcal{D}} = f(v_i, \mathcal{D}) \log\left(\frac{|D|}{df(v_i)}\right)$$

where $f(v_i, d)$ is the frequency of the word v_i in the document d, $|D|$ being the total number of documents in our corpus and $df(v_i)$ is the number of documents with the word v_i in it. The segment vector $\mu(i,j)$ in this case is the mean of the sentence vectors in that segment.

Word Vector: We represented each sentence as a weighted combination of the word vectors in a sentence. The word-vector w_i for each word v_i is specified using Mikolov's Word2Vec[9]. Each sentence s_i is represented as:

$$s_i = \sum_{v} f(v, d) \log\left(\frac{|D|}{df(v)}\right) w_i$$

The segment vector $\mu(i,j)$ is also the mean vector in this case.

Fisher Vector: Paragraph vectors[7] try to embed the sentences in a fixed dimension, but they require extensive training on the source dataset. Instead we use Fisher Vectors, which have been widely used in the vision community [11] for combining different feature vectors (word vectors in our case), and were recently used for question retrieval by Zhou et.al. [15]. The word vocabulary is modeled as a Gaussian Mixture Model, since a GMM can approximate any continuous arbitrary probability density function. Let $\lambda = \{\theta_j, \mu_j, \Sigma_j, j = 1 \ldots N_G\}$ be the parameters of the GMM with N_G gaussians. Let, $\{w_1, w_2, \ldots, w_T\}$ be the vectors for the words $v_1, v_2, \ldots, v_T$ in the sentence s_i for which we need to construct the fisher vector. We define $\gamma_j(w_t)$ to be the probability that the word w_t is assigned the gaussian j,

$$\gamma_j(w_t) = \frac{\theta_j \mathcal{N}(w_t|\mu_j, \Sigma_j)}{\sum_{u=1}^{N_G} \theta_u \mathcal{N}(w_t|\mu_u, \Sigma_u)}$$

We define the gradient vector as the score for a sentence, $G_\lambda(s_i)$ [13]. To compare two sentences, Fisher Kernel is applied on these gradients,

$$\mathcal{K}(s_i, s_j) = G_\lambda(s_i) F_\lambda^{-1} G_\lambda(s_j)$$

where, F_λ is the Fisher Information Matrix,

$$F_\lambda = E_{x \sim p(x|\lambda)}[G_\lambda(s_i) G_\lambda(s_j)^T]$$

$F_{\lambda}^{-}1$ can be decomposed as $L_\lambda^T L_\lambda$, hence the Fisher Kernel can be decomposed to two normalized vectors, $\Gamma_\lambda(s_i) = L_\lambda G_\lambda(s_i)$. This $\Gamma_\lambda(s_i)$ is the fisher vector for the sentence s_i

$$\Gamma_{\mu_j^d}(s_i) = \frac{1}{T\sqrt{\theta_j}} \sum_{t=1}^{T} \gamma_j(w_t) \left(\frac{w_t^d - \mu_j^d}{\sigma_j^d}\right) \quad (1)$$

$$\Gamma_{\sigma_j^d}(s_i) = \frac{1}{T\sqrt{2\theta_j}} \sum_{t=1}^{T} \gamma_j(w_t) \left[\frac{(w_t^d - \mu_j^d)^2}{(\sigma_j^d)^2} - 1\right] \quad (2)$$

The final fisher vector is the concatenation of all $\Gamma_{\mu_j^d}(s_i)$ and $\Gamma_{\sigma_j^d}(s_i)$ for all $j = 1 \ldots N_G$, $d = 1 \ldots Dim$, hence $2 * N_G * Dim$ dimensional vector. We define the segment vector $\mu(i,j)$ as the fisher vector formed by using the word vectors in the segment, hence giving us a unified representation of the segment.

5. EXPERIMENTS

In this section we evaluate our method for identifying document segments suited for learning objectives.

5.1 Data

We made use of two data sets for our experiments:

AKS labeled Data Set: We use the collection of 110 Science documents used by [3] labeled with 68 learning objectives from the Academic Knowledge and Skills (AKS)[5]. We also used term expansions as described in [3] to increase the context of learning objectives. We further identified document segments (at the sentence level) suitable for the learning standard in each of the documents, where applicable.

To build a collection of documents covering multiple learning objectives, we simulated the creation of large academic documents such as text books, by augmenting each lecture note with 9 randomly selected lecture notes. Thus, for each of the 68 instructions that were covered in our data set, we created 5 larger documents each consisting of 10 documents from the original set, giving us a document collection of 340 large documents, with an average length of 300 sentences.

Dataset	#Docs	#Avg. Sentences	#Avg. Splits
AKS Dataset	340	300	10
WikiQA	8100	180	10

WikiQA Dataset: To show the general applicability of our approach on tasks such as document passage retrieval and QA, we also use the recently released WikiQA data set [14] which consists of 3047 questions sampled from Bing[6] query logs and associated with answers in a Wikipedia summary paragraph. As outlined in the approach above, for each of the questions, we created a larger document by including 9 other randomly selected answer passages. For each of the 2700 questions from the Train and Test collection we created 3 such documents, thus giving us 8100 documents.

5.2 Evaluation Metrics

We define the following metrics for our evaluation:

MRR (Mean Reciprocal Rank) : The MRR is defined as the reciprocal rank of the of the first correct result in a ranked list of candidate results.

P@N (Precision@N): Let the set of sentences in the top N segments identified be Γ^{Sys} and further, let the set of sentences in the gold standard be Γ^{Gold}. The precision@N is given by :

$$P@N = \frac{|\Gamma^{Sys} \cap \Gamma^{Gold}|}{|\Gamma^{Sys}|} \tag{3}$$

R@N (Recall@N):Using the same notation described above, the recall @ N is given by :

$$R@N = \frac{|\Gamma^{Sys} \cap \Gamma^{Gold}|}{|\Gamma^{Gold}|} \tag{4}$$

F1@N (F1 Score @N): The F1 Score@N is given by the harmonic mean of the Precision@N and Recall@N described above. **MMA@N (Minimal Match Accuracy@N)** For a collection of D labeled documents, the minimal match accuracy@N is a relaxed value of precision and is given by:

$$\frac{\sum_i^D \mathbb{1}\{|\Gamma_i^{Sys} \cap \Gamma_i^{Gold}| \geq 1\}}{D} \tag{5}$$

where $\mathbb{1}\{\}$ is the indicator function.

5.3 Experimental Setup

For the AKS dataset, we calculate the *idf* using a collection of 6000 Science documents from Wikibooks[7] and Project Gutenburg[8]. For the WikiQA dataset, *idf* was calculated on the 2700 summaries in the training and test collection. Word vectors and fisher vectors were trained on the full collection of English Wikipedia articles to ensure that the Gaussian Mixture model isn't trained on a skewed dataset and can be used across universally for all kinds of english educational documents. The number of gaussians were selected based on the bayesian information criterion.[9]

Choosing the number of top segments: The number of top ranked segments n and the number of splits K both affect the accuracy of the system. For instance, if we set K to be half the total number of sentences, the resulting segments will be very small. Therefore, the value of n needs to be higher to have adequate coverage (recall). Similarly, choosing very few splits can result in large chunks, which can be problematic if the learning objectives were precise and required finer segments. Thus, the choice of n and K depends on the granularity of specification in the learning objectives as well as the nature of content in the document collection.

We use 20% of the dataset (selected at random) as the validation set for tuning the parameters n and k. By varying both n and K we can determine the value at which the system performance (measured using F1 score) is best. Figure 2 shows the variation in F1 Score for different values of K and n. For clarity of presentation, we only show this for the system using TF-IDF vectors. As can be seen, the $F1$ score is best for 10 splits and choosing the 3 best segments closest to the learning objective i.e $K = 10, n = 3$. Figures 3 and 4 show the individual contributions to the $F1$ score.

5.4 Results

5.4.1 Document Segmentation and Labeling

On performing segmentation on the AKS dataset using all three vector approaches, we observe (table 1) that the tf-idf vector representation works best. We noticed that many

[5] https://publish.gwinnett.k12.ga.us/gcps/home/public/parents/content/general-info/aks

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

[7] http://www.wikibooks.org

[8] http://www.gutenburg.org

[9] An index used for model selection $-2L_m + m\ln n$, where L_m is the maximized likelihood, m are the number of parameters and n is the sample size

		@1			@3			@5		
Query Expansion		**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
No Expansion	TFIDF	0.669	0.359	0.468	0.493	0.698	**0.578**	0.395	0.843	0.538
	WORDVEC	0.462	0.357	0.403	0.331	0.633	0.434	0.284	0.829	0.423
	FISHER	0.476	0.366	0.414	0.342	0.679	0.454	0.284	0.855	0.426
With Expansion	TFIDF	0.686	0.320	0.436	0.545	0.701	**0.613**	0.435	0.856	0.577
	WORDVEC	0.483	0.323	0.387	0.351	0.586	0.439	0.308	0.797	0.444
	FISHER	0.481	0.322	0.386	0.351	0.619	0.448	0.305	0.827	0.445

Table 1: Results on the AKS Labeled Dataset

	MRR	MMA@1	MMA@3	MMA@5
TFIDF	0.78	0.652	**0.905**	0.882
WORDVEC	0.56	0.429	0.635	0.782
FISHER	0.55	0.405	0.620	0.715

Table 2: Segment Level Results on AKS Dataset

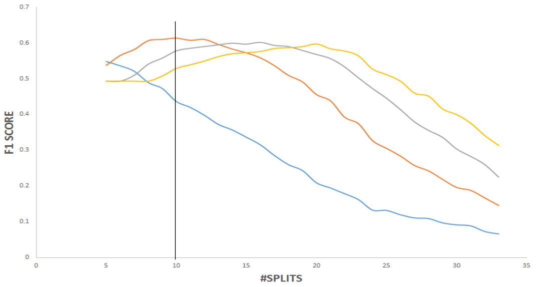


Figure 2: F1 Variation with number of segments at varying depths of retrieval. Best score at 10 segments at depth 3

of the documents in the AKS data set were very well contextualized when changing topics, thus blurring the segment boundaries. For example, in one of the documents which described “Motion in a Straight Line”, the concepts of “velocity”, “acceleration”, “position-time” graphs are intertwined and the topical drift is not easy to observe. As a result, due to the nature of documents in the collection, we hypothesize that the fisher vectors and word vectors which have been trained on large general corpora are unable to adequately distinguish some portions of the text, while the tf-idf vectors which have been tuned on the corpus better reflect the word distributions.

The precision, recall and F1 scores are calculated at the sentence level, thus making it a very strict measure. So we also report segment level accuracy, i.e. how many of the top n segments identified were relevant. A predicted segment is labeled relevant to the external query if at least 70% of the segment overlaps with the gold labeled segments. We evaluate the performance using MRR and MMA@N. Table 2 shows the segment level evaluation of our system.

5.4.2 *Passage Retrieval and QA*

We also conducted experiments with a more discriminative dataset where the topical shift is not as hard to observe. We report (table 3) an MRR of 0.895 and P@1 of 89.4% for the passage retrieval task on each of the documents generated, as described in section 5.1.

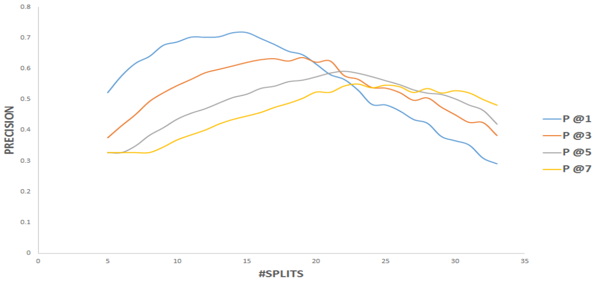


Figure 3: Precision variation with number of segments at varying depths of retrieval. Low values of n and high values of K give high precision. Increasing K while keeping n constant gives a drop in precision.

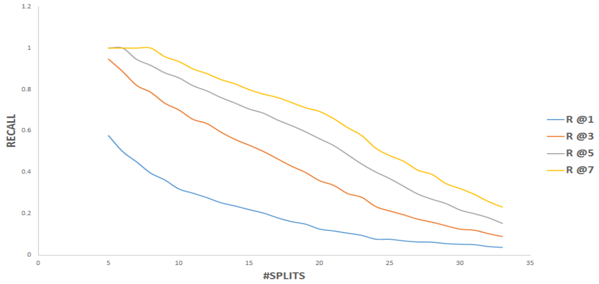


Figure 4: Recall variation with number of segments at varying depths of retrieval. Recall is higher at low values of K and high values of n, and the recall drops considerably as the number of segments K increases.

Further, we also describe our results on the original task, proposed with the data set, of finding the answer in a passage for a question. In our experiments we report results under two conditions: (a) First identifying the best passage and then choosing the best sentence (b) Assuming the best passage is already known and then choosing the best sentence that answers the query (original WikiQA QA task). Table 4 presents results of experiments under both these conditions. It can be seen that our system gives comparable results under both conditions. The state of the art results under condition (b) as reported in the original paper is an MRR of 0.696. Our system, though not designed for the original task, has an MRR score 10% lower than the best system reported.

				@1			@3		
	MRR	**MMA@1**	**MMA@3**	**P**	**R**	**F1**	**P**	**R**	**F1**
TFIDF	0.807	0.797	0.812	0.839	0.893	0.865	0.308	0.958	0.466
WORDVEC	**0.895**	0.877	0.913	**0.894**	0.914	0.904	0.315	0.984	0.478
FISHER	0.865	0.842	0.887	0.863	0.885	0.874	0.298	0.975	0.457

Table 3: WikiQA Passage Retrieval Results

	MRR **Top Segment**	**MRR** **Gold Standard Passage**
TFIDF	0.528	0.495
WORDVEC	0.548	0.586
FISHER	0.577	**0.597**

Table 4: Finding the sentence answering the question: "Top segment" uses our system to select the best passage while "Gold standard passage" uses the actual passage labeled in the data set

6. DISCUSSION AND CONCLUSION

In this paper we described the novel task of automatically segmenting and labeling documents with learning standard objectives. Using a state of the art dynamic programming algorithm for text segmentation, we demonstrate its use for this problem and report a sentence level $F1$ score of 0.613 and segment level MMA@3 of 0.9. We also demonstrated the effectiveness of our approach on document passage retrieval and QA tasks.

Our method is completely unsupervised and only requires a small validation set for parameter tuning. This makes our work general and easily applicable across different geographies and learning standards. Identifying document segments best suited for learning objectives is a challenging problem. For instance, portions of documents that introduce or summarize topics or build a background in an area are very hard to disambiguate for the algorithm due to the lack of observable topic shifts. Developing more sophisticated cohesion and topical diversity measures to address this problem could be an interesting direction of further research.

In future work, we would also like to explore methods that jointly segment and label documents. We also plan to use other methods of vector construction such as paragraph vectors [7] to better represent segments using a training data set as well as semi-supervised text segmentation methods.

7. REFERENCES

[1] A. A. Alemi and P. Ginsparg. Text segmentation based on semantic word embeddings. *arXiv preprint arXiv:1503.05543*, 2015.

[2] C. L. Clarke and E. L. Terra. Passage retrieval vs. document retrieval for factoid question answering. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 427–428. ACM, 2003.

[3] D. Contractor, K. Popat, S. Ikbal, S. Negi, B. Sengupta, and M. K. Mohania. Labeling educational content with academic learning standards. In *Proceedings of the 2015 SIAM International Conference on Data Mining, Vancouver, BC, Canada, April 30 - May 2, 2015*, pages 136–144, 2015.

[4] H. Daumé III and D. Marcu. Bayesian query-focused summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 305–312. Association for Computational Linguistics, 2006.

[5] L. Du, J. K. Pate, and M. Johnson. Topic segmentation in an ordering-based topic model. 2015.

[6] M. A. Hearst. Texttiling: A quantitative approach to discourse segmentation. Technical report, Citeseer, 1993.

[7] Q. V. Le and T. Mikolov. Distributed representations of sentences and documents. *arXiv preprint arXiv:1405.4053*, 2014.

[8] J.-P. Mei and L. Chen. Sumcr: a new subtopic-based extractive approach for text summarization. *Knowledge and information systems*, 31(3):527–545, 2012.

[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean. word2vec, 2014.

[10] Y. Ouyang, W. Li, S. Li, and Q. Lu. Applying regression models to query-focused multi-document summarization. *Information Processing & Management*, 47(2):227–237, 2011.

[11] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *Computer Vision–ECCV 2010*, pages 143–156. Springer, 2010.

[12] M. Riedl and C. Biemann. Text segmentation with topic models. *Journal for Language Technology and Computational Linguistics*, 27(1):47–69, 2012.

[13] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *International journal of computer vision*, 105(3):222–245, 2013.

[14] Y. Yang, W.-t. Yih, and C. Meek. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2013–2018. Citeseer, 2015.

[15] G. Zhou, T. He, J. Zhao, and P. Hu. Learning continuous word embedding with metadata for question retrieval in community question answering. In *Proceedings of ACL*, pages 250–259, 2015.