IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Change-oriented Summarization of Temporal Scholarly Document Collections by Semantic Drift Analysis

NAMAN PAHARIA[1], MUHAMMAD SYAFIQ MOHD POZI[2], AND ADAM JATOWT[3]

[1]IIT Kharagpur, Kharagpur, India (e-mail: namanpaharia.27@iitkgp.ac.in)
[2]School of Computing, Universiti Utara Malaysia, Sintok, 06010, Malaysia (e-mail: syafiq.pozi@uum.edu.my)
[3]University of Innsbruck, Innsbruck, Tirol, Austria (e-mail: adam.jatowt@uibk.ac.at)

Corresponding author: Naman Paharia (e-mail: namanpaharia.27@iitkgp.ac.in), Muhammad Syafiq Mohd Pozi (e-mail: syafiq.pozi@uum.edu.my) and Adam Jatowt (e-mail: adam.jatowt@uibk.ac.at).

This work was supported in part by the Malaysia Ministry of Higher Education under Grant FRGS-RACER/1/2019/SS09/UUM//2.

ABSTRACT The number of scholarly publications has dramatically increased over the last decades. For anyone new to a particular science domain it is not easy to understand the major trends and significant changes that the domain has undergone over time. Temporal summarization and related approaches should be then useful to make sense of scholarly temporal collections. In this paper we demonstrate an approach to analyze the dataset of research papers by providing a high level overview of important changes that occurred over time in this dataset. The novelty of our approach lies in the adaptation of methods used for semantic term evolution analysis. However, we analyze not just semantic evolution of single words independently, but we estimate common semantic drifts shared by groups of semantically converging words. As an example dataset we use the ACL Anthology Reference Corpus that spans from 1974 to 2015 and contains 22,878 scholarly articles.

INDEX TERMS temporal, summarization, ACL, clustering, semantic changes, cluster analysis

I. INTRODUCTION

One of the challenges in big data processing is how to effectively represent and visualize enormous amounts of data that exists in high dimensional spaces, without significant degradation of information, which could lead to misinterpretation of the original data. Text, for example, is one kind of such data that can be considered as highly dimensional data, used to describe objects, events, sentiments, such that all of these can exist in different contexts, that keep also changing over time. As an example, we can observe how text changes over time in datasets of scholarly publications. Scholarly publications are published continuously and their topics tend to change and evolve quite quickly (especially in computer science).

There is a growing interest in the research community to develop automation frameworks to process, analyze, and model publications datasets for generating coherent and sensible summary, given a large size of any particular textual document collections. These are commonly known as automatic text summarization systems [1, 2]. Since the 1950s [3], much work has been done to improve the automatic text summarization so the results would match human-made summaries. The problem is still far from being resolved as issues like sentence redundancy, sentence ordering, domain specificity, multi-modal summary and temporal dimension, semantic changes as well as many more pose significant challenges or affect the quality of summaries generated by automatic text summarization systems.

As times change, so do languages. In order to match evolving contexts, words often acquire new meanings or are used in different contexts. A set of words that might have been hardly associated with each other in the past, maybe now commonly used. New terms are being generated to accurately express or explain an evolving or emerging concept within a new context. This is especially true in scientific scholarly research publications, such as computer science, engineering, or medical related research, as it is common for new words or terms to be coined or derived, so that they can properly illustrate the novelty of the published research works in a concise yet accurate manner.

In this paper, the problem we focus on is how to analyze and visualize evolution of scholarly research based on large

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

corpora such as one consisting of computer science scholarly publications, over a period of time. As such datasets typically cover tens of thousands or more articles it is very difficult to understand the important changes that occurred in the scientific domain over time. We propose a novel change-oriented summarization approach based on word embedding derived evolution, semantic drift clustering as well as customized sentence selection to provide information on key changes that occurred in the data over time. The novel aspect of our work is that we focus on detecting and portraying major semantic changes over time and we adapt concepts from research on word semantic evolution analysis. However, unlike prior works on computational approaches to semantic word change [4, 5] we do not focus on individual words but treat words in aggregate attempting to analyze the drift in the entire temporal document collection. We cluster changing words into changing concepts such that words in the same cluster undergo semantic change of similar direction. This novel concept of grouping words over time allows finding crucial changes on the level of entire corpus. We finally rank the detected clusters based on their importance. Finally, we output the top clusters as well as generate explanatory sentence pairs as another form of the output for better understanding the changes that took place over time in the underlying dataset.

As an underlying dataset for our experiments we use the Association for Computational Linguistics (ACL) dataset, and specifically, the Parcit Structured XML[1] [6] version of the ACL Anthology Reference Corpus [7]. The dataset contains 22,878 scholarly articles that are related to computational linguistics, which spans from the year 1970s to 2015. By analyzing this data we provide answers to questions about what concepts are highly changing and what are their directions of changes with respect to time.

This research is guided in particular by the following main research questions:

1) Which important concepts in the domain of the collection have been semantically changing over time?
2) Can we provide explanations for the important changes? That is, for each drifted crucial concept, can we find an explanation given that outlines and emphasizes the trajectory of the drift and its effect.
3) How to represent these changes in the form of summary?

To sum up, we make the following contributions in this paper:

1) We first construct a vector based word semantic shift representation by capturing temporal semantic difference of terms, wherein each terms' semantics is represented in different time periods.
2) We then formulate a concept-based clustering mechanism by grouping terms that are exhibiting highly correlated semantic differences in the different time periods, based on various distance functions.
3) We next propose a sentence selection mechanism to represent and differentiate the meanings of generated clusters.
4) We finally evaluate the resulting clusters in real world through human evaluation analysis, and we identify the best distance function to be used with our proposed change-oriented temporal based summarization method.

The remainder of this paper is organized as follows. Section II examines the progress and recent directions in the summarization task in general and temporal summarization in particular. Section III proposes a summarization methodology used in this research, including data collection and preprocessing task, text representation, and our approach for capturing, representing and analyzing the temporal semantic changes. Section IV describes the experimental settings, followed by Section V, which describes the experimental results. Then, in Section VI we describe the limitations encountered in this research. Finally, Section VII concludes the paper.

II. RELATED WORKS

A. OVERVIEW

Automatic text summarization is a task to summarize a single document, or multiple documents "such that the result conveys important information from the original text(s), and that is no longer than half of the original text(s) and usually, significantly less than that" [8]. Summaries are useful in today's information retrieval tasks due to the abundance of data to be consumed in a short time. For example, for user queries, usually search engines generate snippets as the previews of the returned documents [9].

In general, two common approaches were widely used and implemented in generating a sensible summary [2]: abstraction approach [10, 11], and extraction approach [12, 13]. Abstractive summarization methods aim to convey specific information, available from any particular document in newly generated text. On the other hand, extractive summarization methods work by identifying relevant portions of content in the document (these could be portions of titles, lead sentences, cue words) in order to summarize the document.

Text summarizers can also be grouped into single-document summarizers [14] or multi-document summarizers [15]. A single-document summarization method creates a summary of a single document, while multi-document summarization method makes a summary of multiple documents or their clusters, forming a single output summary. In addition to that, an automatic text summarizer can also be classified as a generic summarizer or user- or query-oriented summarizer. The former provides an unbiased overview of a particular document [16] while the latter provides a summary that tallies with user requirements, such as topic-focused query, user profile and so on [17]. Last, but not least, summarizers can also be grouped into supervised learning based and unsupervised learning based methods. A supervised learning based approach requires annotated data as it usually needs

[1] https://acl-arc.comp.nus.edu.sg/archives/acl-arc-160301-parscit

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

to classify which sentences should be presented in summary, and which one is not [18, 19]. On the other hand, a unsupervised learning based approach does not depend on any training data. Usually, a clustering task is performed to group portions of text components into similar clusters prior to the summarization task [20, 21].

It can be seen that much of the work in automatic text summarization does not involve the temporal aspect of the corpora within the summarization framework. Temporal based text summarization works are mostly based on real world events, such as police reports, news, stories, and product reviews [22, 23, 24]. Temporal summarization approaches for scholarly literature however, have not been properly established yet. The existing works in scholarly literature are mostly on identifying temporal scientific concepts [25, 21, 26]. This is understandable, as scientific literature usually closely relates to previous findings in science, while news and other real world events can be very dynamic and easily drifted by external concepts.

B. TEXTUAL SUMMARIZATION TECHNIQUES

In general, textual summarization techniques can be grouped into three categories of language models. The categories are as described below:

1) Linguistic Based Techniques

Basically, there are four approaches in linguistic based techniques, such as lexical based approaches, syntactic based approaches, semantic based approaches and discourse based approaches [27, 28]. Lexical analysis, which can sometimes be referred to as part of tokenization task, is the process of converting a stream of text into individual tokens. A set of predefined rules are established, such that a token should not consist of punctuation and whitespaces, each token is determined based on a contiguous string or alphabetic characters, and tokens are separated by white space, line break or punctuation characters. Syntactic based approaches are related to techniques that are used to capture the grammatical structural information within text. For example, it is common to build syntactic trees as a way to model the grammatical structure (e.g. collocation of words) from a set of predefined rules [29, 30]. In semantic based approaches, the linguistic analysis is focused on how to capture and measure the meaning from text. One of the commonly used technique in capturing the meaning from text is through Latent Semantic Analysis (LSA) [31, 32], under assumption that, words that are close in meaning, will also appear in similar pieces of text. Finally, in discourse based approaches, the relation that represents connections between sentences and parts in text is properly modelled [33, 34]. Then, this discourse model is used to measure the coherence and cohesion of any similar pieces of text. This is especially crucial in text summarization, as the coherence and cohesion level of the summary might be degraded during the summarization task, even though the original text semantic is preserved.

2) Machine Learning Based Techniques

As new machine learning algorithms have been developed and implemented over time, it has also affected research in text summarization systems that increasingly have been employing machine learning based techniques. Besides using new kinds of features, new machine learning algorithms were capable of solving various performance issues, such as overfitting and accuracy of standard machine learning algorithms. For example, in [35, 36], several machine learning algorithms were combined into a single meta-learning algorithm to improve the summarization results to classify and rank sentences in order of importance. The most important sentences were used as a basis in generating the summary.

One of the main reasons why text modelling through machine learning was difficult is because of data sparsity issues. Vectorizing texts into fixed length of vector resulted in sparse vectors, affecting the model performance in a negative way. Some strategies to workaround this issue have been proposed, such as in [37, 38], a document summarization through sparse optimization in yielding better summarization results. These techniques solve the sparsity issues by first representing the data in a compact form prior to the vectorization task. By adding sparsity constraints on the number of output vectors, condensed information, which can be treated as word salience, is generated.

3) Context-Modelling Based Techniques

As mentioned before, data sparsity is one of fundamental problems in modelling any particular language through texts. However, despite improvement being done on standard machine learning settings in dealing with data sparsity and other fundamental problems that exist in textual data, it must be noted that texts are always viewed in a specific context. This context can be very specific, such as culture, region, time and many more. For example, word *gay* can be viewed in negative or positive way, depending on the specific discussion surrounding the word *gay* at that particular instance [39].

Hence, the problem at hand now is how to incorporate specific context into the text information and model those combined information to ensure the obtained results, while also accurate, are also sensible to human interpretation. In [40], a context model based on language modelling approach for bursty features is proposed. Here, bursty feature is a feature representing sudden surge of the frequency of a single term or phrase in a text stream. Based on this bursty feature context model, the text summarization will automatically identify bursty features, in which, these features are then grouped into k-topics. The centroids of these topics are computed to facilitate the text summarization generation task [41, 42].

However, as textual data exists in many kinds of formats and is highly dimensional, traditional feature extraction methodology is inefficient. Hence, in 2013, a neural network based method to transform words into vectors, called Word2Vec was proposed [43, 44]. Word2Vec represents word semantics in one $n-$dimensional vector. Word2Vec has been

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

used in many domains such as in social network analytics [45, 46], medical text summarization [47, 48], and many other related text analytic applications. Since then, various neural-network based textual context modelling approaches have been developed to improve the contextual and semantic interpretation such as Glove [49], Fasttext [50], and, especially nowadays, BERT [51].

C. TEMPORAL ANALYSIS OF SCHOLARLY DATA

A particular research niche that is currently gaining interest in the summarization task is how to incorporate time factor into any automatic text summarization model itself. This is because massive amounts of spatial and temporal data are being generated on a prodigious scale. Such examples of textual data can be seen in news [55, 56], social networks [57, 58], and scholarly articles [47, 59, 60].

When it comes to scholarly document datasets, most of temporal analysis on large corpora can be considered as synchronic type (static) analysis. Those works only visualize the pattern changes of the corpus over time, without emphasizing why such captured patterns (from the large corpora) occur in the first place. For example, in [61, 62], a BERT model based on MIMIC-III dataset [63] is built to improve the temporal extraction task on medical annotated data called THYMES [64]. Such works are also observed in [65] based on story corpus in ROCStories [66], and as temporal event extraction in financial related news [67]. From a high-level of view, it can be generally stated that pre-trained language models are usually being used as a method for text vectorization, in specific domain, and for specific application.

Table 1 describes a few works on temporal analysis of scholarly data that were done till now. One approach [54] performs the temporal analysis based on differences in frequency, meanings and the underlying temporal scopes of temporal expressions used in scientific writing from 1665 to 2007. Similar research has also been observed in [52, 25], using frequency analysis to identify emerging terms in scientific scholarly collections. Further extension to the frequency analysis, such as semantic analysis of specialized terminology, could help in detecting emerging trends [53] and in summarizing entire domain-specific collections [47]. For example, in [53], the authors propose detecting emerging trends in scholarly publication collections in the computer science and bio-informatics areas. Rather than employing citation analysis or straightforward frequency-based trend assessment, as it has been usually done, the authors use temporal word embedding to observe shifts in scientific language over several decades. A simple improvement of approach in [53] would be to use contextualized embedding models pre-trained on domain-specific collections such as SciBERT that were trained on scholarly corpora [68]. Another one is to consider specialized term extraction techniques such as ones based on recognizing meaning shifts between general and domain-specific language [69]. In our work we further extend change-based scholarly data summarization approaches based on these and further ideas. To the best of our knowledge, we are the first to propose semantic evolution based approach to change-oriented summarization of scholarly data collections such that major changing concepts can be identified to portray the evolution of entire scholarly collection.

III. METHOD

We start first by discussing the characteristics of the dataset we use and we then introduce step by step the components of our method. Fig. 1 shows the overview of our approach. In the first step, we utilize temporal data as an input dataset (ACL Anthology Corpus in our case) and divide it into multiple time frames. Then word embeddings for each time frame are generated using the semantic information representation module. Then these temporal embeddings are sent to the Clustering module and Sentence selection module which after ranking, generate results in the form of trending clusters and representative sentences.

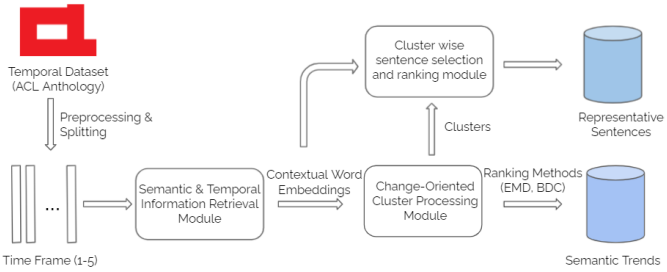

FIGURE 1: Overview of our approach.

A. DATA COLLECTION AND PREPROCESSING

1) ACL Dataset

The ACL Anthology Reference Corpus consists of conference and journal papers in natural language processing and computational linguistics research domains which are published within ACL Anthology. The corpus consists of 22,878 papers published during a span of 36 years from 1979 to 2015. The distribution of the number of research papers per year is non-uniform, with increasing numbers of papers in recent years as shown in Figure 2.

Every publication consists of the following:

1) Abstract: A summary of an article.
2) Body text: The main text of the article, typically composed of introduction, results, conclusion or discussion sections and including also abstract.
3) Metadata such as Author information, Publication date and others.

In this work, we use the body text and the year of the publication date (i.e., hence yearly granularity) to visualize how specific terminology drifted from 1979 to 2015.

2) Time Division & Vocabulary Creation

We first extracted the body text from each paper in the corpus. The dataset was then divided into 5 non-overlapping time-frames: 1979-1995, 1996-2000, 2001-2005, 2006-2010,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

TABLE 1: Summary of recent works in temporal summarization of scholarly article collections.

Scholarly Corpus	Dataset Periods	Techniques	Time Interval(s)	Analysis Unit	End Result
Computer Science [52]	1958 - 2015	Word2Vec + RNN	Years: 10, 15, 20, 25	Phrase	Predicting the next year's keyword trend
CORD-19 Dataset [47]	1955 - 2020	Word2Vec + k-means Clustering	5 Years	Word	Temporal semantic word association
Computer Science & Bioinformatics [53]	1987 - 2018	Word2Vec + Rank Ascent Identification	Incremental: 1 - 30 Years Window: 3 Years	Phrase	Identification of emerging keywords
Royal Society Corpus & SciTex [54]	1665 - 1869 1966 - 2007	Kullback-Leibler Divergence	50 Years	Part-of-Speech	Representing diachronic expressions through annotated time expressions

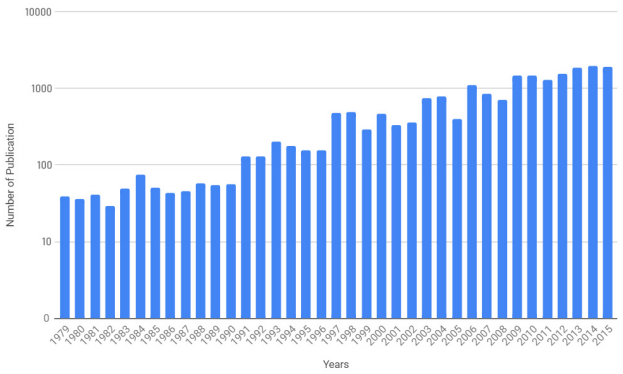

FIGURE 2: Articles distribution over years.

2011-2015. The skewness in data distribution was considered while making these divisions in order for the resulting time periods to contain more or less equal data portions. An initial vocabulary list, V, was created using the complete dataset after filtering it for stopwords and digits, and extracting nouns as well as noun phrases. This vocabulary was then filtered to remove words with frequency smaller than 100, since we focus on the changes of high significance in the NLP/CL scholarly field. Then the vocabulary was further filtered to remove words that occurred only in a single time frame. Furthermore, words with length less than 3 were also discarded to only consider the most meaningful words.

B. TEXT REPRESENTATION

Thanks to deep learning models which learn usage-dependent representations, neural contextualised word representations have gained widespread use in NLP. Contextualised representations have been shown to encode lexical meaning dynamically, reaching high accuracy on multiple NLP tasks. Different language models can be used to generate these embeddings like Embeddings from Language Models (ELMo), Bidirectional Encoder Representations from Transformers (BERT), GPT-2, etc. We used a BERT based model, called SciBERT [68], with 12 attention layers and hidden layers of size 768 which was pre-trained on a dataset of 1.14M papers taken from Semantic Scholar [2] constituting a corpus size of 3.17B tokens which consists of 18% papers from the computer science domain and 82% from the broad biomedical domain. BERT is a neural network model based on Transformer [70], which in its vanilla form includes two separate mechanisms, an Encoder that reads and converts text to vectors and numbers, and a Decoder that produces a prediction for the task. BERT relies on a transfer learning approach proposed by [71], where the first step is to pre-train the network as a language model on large corpora in order to learn general contextual word representations. It was pre-trained on two tasks:

1) Masked Language Modeling - From the input, a percentage of words are masked in advance, as shown in Fig. 3[3], and the objective is to predict these masked words from an unmasked context. This allows BERT to leverage both left and right context, meaning that a word w_t in a sequence is not determined just from its left sequence $w_{1:t-1} = \{w_1, \ldots, w_{t-1}\}$ as is the case in the traditional language modelling task - but also from its right word sequence $w_{t+1:n} = \{w_{t+1}, ..., w_{t+n}\}$. If the w_i^{th} token is chosen, it is replaced with (i) the $[MASK]$ token 80% of the time, (ii) a random token 10% of the time (iii) and the unchanged w_i^{th} token 10% of the time. Then, w_i is been used to predict the original token with cross entropy loss.
2) Next Sentence Prediction - the model receives pairs of sentences as input and learns to predict if the second sentence in the pair is the subsequent sentence in the original document.

This is usually followed by a task-specific fine-tuning step which is a technique of applying a pre-trained LM. It changes the pre-trained parameters slightly via downstream task learning. In particular, BERT has been shown to achieve state-of-the-art results for 11 NLP tasks, including document classification, question answering, and dependency parsing. In our experiment, SciBERT was fine-tuned for domain adaptation on the entire ACL corpus to incorporate the different semantic meaning any word in the corpus will have within a time-frame or inter time-frame.

The temporal semantic change requires the analysis of word semantics between different time frames. We exploit SciBERT contextualised embedding for representing the semantic information of the words. The data used for the analysis was stored as Mean Embedding Map, which in one time frame stores all the words in the vocabulary as a map

[2]https://www.semanticscholar.org/

[3]Image taken from: https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

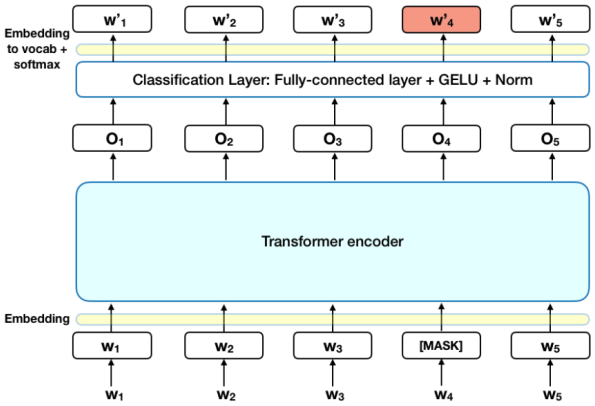

FIGURE 3: Masked Language Modeling

Terms	Least Similarity	Terms	Highest Similarity
seventh	0.6420	must	0.9975
fell	0.6728	example	0.9973
plan	0.7319	words	0.9972
mann	0.7648	corpus	0.9970
entail	0.7849	node	0.9970
west	0.7982	japanese	0.9969
copyright	0.7996	probability	0.9969
maxim	0.8000	terminal	0.9968
prop	0.8143	threshold	0.9968
jack	0.8190	complex	0.9966
stanford	0.8212	automaton	0.9966
volume	0.8385	frequency	0.9966
land	0.8487	rates	0.9965

TABLE 2: Terms with the smallest and highest cosine similarity between the last and the first time unit.

with key as the mean contextualised embedding for every occurrence of that word within that time frame. Thus this map stores the mean semantic information which will be used for further analysis.

C. REPRESENTING WORD CHANGE

Analysing term changes in scientific corpora as well as detecting emerging trends have been traditionally done by frequency analysis over time. A different approach, which we propose in this paper, is to compare semantic meanings of words, using the contextualized embeddings. Comparing embeddings of the same word in different time frames can be used to derive a mathematical representation of the change of a word.

In particular, the cosine similarity between the embedding of a target word in the first time frame (1979-1995) and the embedding of this word in the last time frame (2011-2015)[4] could represent the overall semantic change for the word such that the lower the similarity value, the higher the semantic change expected.

Cosine similarity between vector $\vec{u_1}$ and $\vec{u_2}$ is given by

$$sim(\vec{u_1}, \vec{u_2}) = cosine(\vec{u_1}, \vec{u_2}) = \frac{\vec{u_1} \times \vec{u_2}}{||\vec{u_1}|| \cdot ||\vec{u_2}||} \quad (1)$$

Table 2 shows that cosine similarity alone on the level of individual terms will not suffice for summarizing semantic change of words. It can be understood that constituent words from the high cosine similarity column of Table 2 have relatively high occurrence frequency, suggesting that words with high frequency have low semantic change. This observation is reinforced by the plot in Fig. 4 in which words with high frequency have near zero semantic change degree. While the results for high similarity values are rather intuitive, the results for the low values of the similarity deviate from our expectation. Theoretically these words should have the highest semantic movement, yet as we can observe they rather carry little significance for our task, neither offer any valid explanation. As reflected by Table 2,

[4]In the subsequent parts of the paper we will use the term "time frame 1" and "time frame 5" (or the first and the last time frame) for these, respectively

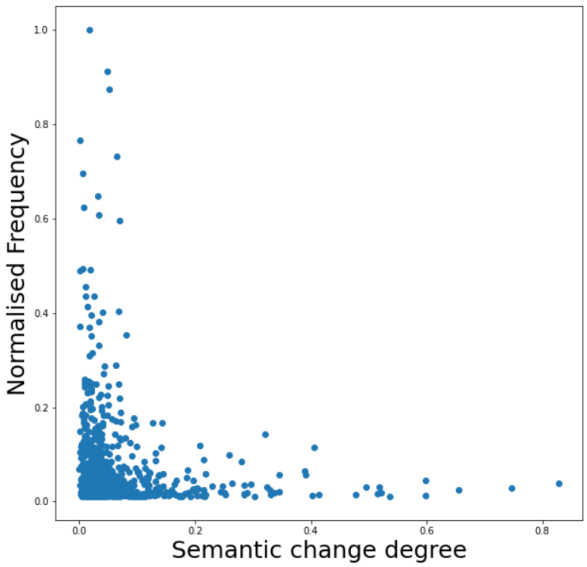

FIGURE 4: Word frequency vs. the semantic change degree.

named entities (e.g., `Xian`, `mann`, `stanford`), or numbers (e.g., `seventh`, `one`), or standard words like `copyright`, `mention`, `west`, `program` and `volume` seem to be coincidentally used in different contexts in different time frame. For better understanding, consider `sports program` and `education program` where the term program is used in different contexts thus it has low similarity value between its representation in time frame 1 and 5, although actually this case does not represent the actual temporal change. Likewise `copyright 1985` and `copyright 2015` are used in different contexts while the meaning of the term copyright remains the same.

Thus, to recognize and understand key semantic shifts on the level of the entire collection, we explore the possibility to group words with related meaning drifts. For each word $w_i \epsilon V$ we define a difference vector $\vec{d_{w_i}}$ representing the difference of its embedding in the first time frame $\vec{u_{w_i}^1}$ and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

the one in the last time frame $\vec{u_{w_i^2}}$.

$$\vec{d_{w_i}} = \vec{u_{w_i^1}} - \vec{u_{w_i^2}} \tag{2}$$

This difference vector captures the information representing the meaning change of a term from the first to the last time frame. To better understand this concept, let's take an example of the term `machine`. Its difference vector will point to words which the term machine moved away from, i.e. mechanical machine, factory, workshop and so on, towards the context uses such as algorithm, learning. The representation of this semantic change from 1979 to 2015 will be embedded in the difference vector.

To explore the drifts of target words (see Algorithm 1) we compute the cosine similarity values between the difference vector of w_i ($\vec{d_{w_i}}$) and other words in vocabulary $[w_0, ..., w_{i-1}, w_{i+1}, .., w_n]$ to find the concepts from where the target word w_i moved away (lowest similarity) and the words that w_i moved towards (highest similarity). In Table 3 we show the obtained top words with the lowest and highest similarity values for few selected target words w_i. For example, we can observe there that the word `merit` in 1980s meant (in most of the cases) the quality of being worthy or praised, yet it drifted towards the concept of evaluation in ML models in 2011-2015. Similarly the word `miner` in 1980s represents a person working in mines as its dominant use but it becomes related to data extraction and information retrieval in more recent times. Similarly, an interesting term `intercept` which was generally related to obstruct or kill in 1980s, is now mainly used in coordinate geometry and measurement.

Algorithm 1: Word movement pattern algorithm.

```
all_term_dict = {};
for w_i in V do
    cossim_values = {};
    d_{w_i} = u_{w_i^1} − u_{w_i^5};
    for w_j in V do
        cossim_values[w_j] = sim(u_{w_j}, d_{w_i});
    end for
    cossim_values ← sort(cossim_values.items());
    all_term_dict[w_i] = cossim_values;
end for
```

D. CHANGE-ORIENTED TERM GROUPING

For summarizing the semantic changes in a corpus, analysing the relative movement for all the words is important. When words are plotted in a hyperplane, the distribution of the words changes with time, and this change can be plotted for every word. As exposed above, some words have meaningful semantic drift while some are subject to random or no change. Grouping words by the patterns of their semantic drift will help to remove random and meaningless semantic changes (such as some examples in Table 2) and will allow us to find important changes. In other words, we look for interrelations between the movement patterns of different words. While many words may change their semantics in non-coordinated way, this relation can be sometimes positive,

TABLE 3: Examples of word movement patterns in our dataset.

Terms	Diverted From	Destination Word List
web	stein, fin, ray, ink, fell	document, advertising, words, text, citation, sentiment
treatment	conception, notion, exposition, formulation, treat	medication, drug, surgery, symptom, disease, health
intercept	attack, kill, shoot, combat, fight	beginning, logarithm, effects, latitude, centroid
miner	work, mining, father, husband	induction, information, extraction, statistics, retrieval
machine	workstation, coordinator, factory, constitution, graphic	bayes, learning, reinforcement, radial, regression
reinforcement	justification, proliferation, viability, uniformity	machine, descent, bayes, markov, radial
activation	elaboration, instantiation, initiation, manifestation	radial, loss, hinge, regularization, gradient
triplet	multitude, couple, variety, virtue, continuum	supervision, active, works, learning, imitation
merit	worthy, worth, warrant, wish, sake, permit	confidence, performance, improvement, accuracy, delta, slope

meaning that it characterizes a group of words that converge in terms of their semantics by moving towards the same point in the semantic vector space, even though at first the words may have had no or limited shared semantics. Alternatively, the relation can be also negative, for the words that divert apart from each other over time, despite having similar semantics initially.

To understand this relation better and to summarize semantic changes more efficiently, we cluster words which are "coming" from different meanings but which "converge" together. As the number of clusters is not available beforehand and can not be easily estimated beforehand, we use Affinity Propagation Algorithm [72]. It is an incremental graph-based algorithm that works by selecting exemplar of each data point by passing two types of messages - responsibility and availability between data points in an iterative manner. AP clustering algorithm does not need to have specified the initial cluster centers but automatically finds a subset of exemplar points which can best describe the data groups by exchanging messages. The number of exemplars is the same as the number of clusters.

To generate clusters of words converging together, we utilize the computed data which is shown in the *Destination Word List* column of Table 3 as it represents the concept towards which a given word is moving. Affinity Propagation algorithm takes input vectors for all words and clusters them based on the similarity of these input vectors. In our case the input vector is formed as the mean word embedding of the *Destination Word List* terms of each target term.

Since the clustering involves words' difference vectors, the words in a cluster could have different initial semantic meanings yet they all converge together or move towards

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

the same semantics. Example, words like `Architecture`, `Input`, `Parameter`, `Weight`, `Dimensions` are basically unrelated to each other and have different meaning in the initial time frames (1985-1995) but based on the subset of the scientific corpus spanning 2010-2015, these words represent or are related to deep learning models. Similarly words such as `web`, `document`, `link`, `article`, `summary` represent electronic media today, but terms like `web` and `link` had completely different meaning in 1985 as the Web was developed around 1989. Another similar example can be `software`, `library`, `script`, `package`, `toolbox`, `apache`, where in 1985, toolbox and package would mean mechanical instruments, a library would be a collection of resources and books, and script would represent written documents, but all these words semantically converge together. Thus we can deduce that the final destination for the movement pattern of words in these clusters is similar for all their constituent words. The entire process of change oriented clustering, starting from BERT-based word embedding to final clusters is summarized in Fig 5, where we start with calculating the difference vector for a particular target word (e.g., machine as is indicated in the figure) and then calculating the cosine similarity of that vector with each word in the corpus. Thus obtaining the "diverted from" word list (e.g., workstation, coordinator, factory) and the "destination word list" list (e.g., reinforcement, learning, bias). The mean embeddings of the constituting words of the "destination word list" list are averaged to form the change embeddings for each target word such as a machine. A similar process is conducted to obtain the change embeddings for all the words in the corpus. which are then passed as an input to the AP clustering resulting in the produced final clusters.

E. CLUSTER RANKING

Clusters produced in the previously-described change-oriented clustering represent the temporal movement of word semantics. The output of Affinity Propagation resulted however in 340 clusters, and these results consist of clusters whose constituent words may have significant and meaningful semantic drift and others with negligible change. We then need to determine the cluster quality to detect the most important and informative clusters. Thus, we propose ranking method for the clusters based on their constituent words.

1) Earth Mover's Distance

To rank clusters on the basis of semantic importance we exploit an observable property of the clusters that, as the words converge together to the similar semantics, they appear together more frequently in the same sentences than before. Thus for each cluster, we calculate the frequency counts of the occurrences of this cluster's members in the same sentences in each analyzed time frame. In other words, for each cluster we determine the number of sentences having high containment of the cluster's words in both the first and the last time frame.

To compute EMD distance, we create a discrete probability distribution of sentences based on the number of common words of that cluster, for every cluster. For example, Fig. 6 shows a sample visualization of the probability distributions of sentences containing certain cluster's words based on the numbers of these words in the start and end time frame. We consider the number of cluster members in the same sentence to be utmost 10. Let X be the set of records in x, where x is the set of all the sentences in a time frame and Y be the integer representing the number of words common in a sentence and cluster. Thus for a cluster i, the probability of sentences having n words in common with the cluster (or $P_i(X \cap y = n)$) will be the number of sentences with n common words divided by the total number of sentences in a time frame. Hence the discrete probability distribution for the frequency count of a cluster will be:

$$y_i = [P_i(X \cap y = 1), P_i(X \cap y = 2), ..., P_i(X \cap y = 10)] \quad (3)$$

In particular, Wasserstein distance (also called Earth Mover's Distance or EMD) is used to calculate the distance between these two distributions. Given two random distributions, EMD can be conceptually portrayed as the task of taking a mass of earth (one distribution) to spread it in space understood as a collection of holes (another distribution) in that same space. EMD measures the least amount of work needed to fill the holes with earth. Thus for a given cluster, the higher the work needed, the higher the semantic importance & novelty of that cluster. Note that EMD is good for our case as there are ordinal relations between units of the distribution. Sentences with 4 words of the same cluster are more important than sentences with 3 words, which in turn are more important than sentences with only 2 words. If for a cluster there are many sentences containing large numbers of its words (for example, sentences with 4 or more words from the cluster) it means this cluster is quite coherent. EMD in our case is used for measuring the difference in the cluster's coherence at the last and the first time frames. Suppose there is a cluster for which there were only a few sentences with high numbers of its member words in the first time unit. On the other hand, in the last time unit many sentences can be found that contain a large number of cluster words. In this case, EMD will be high and the cluster will be judged as important since it represents words that converged to each other over time.

EMD is defined to minimize the following equation:

$$WORK(U, V, F) = \sum_{i=1}^{m} \sum_{j=1}^{n} d_{i,j} f_{i,j} \quad (4)$$

where $d_{ij} = d(y_i, y_j)$ is the ground distance between y_i and y_j, and $F = [f_{ij}]$ such that f_{ij} is the flow that needs to be determined between y_i and y_j. In this work, variables in Eq. 4 are:

The top-ranked clusters by this metric are given in Table 4. As is observable from the table, most of the top clusters (cluster 1, 3, 5 and 6) consist of words which drifted towards a general context of data science and machine learning, as is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

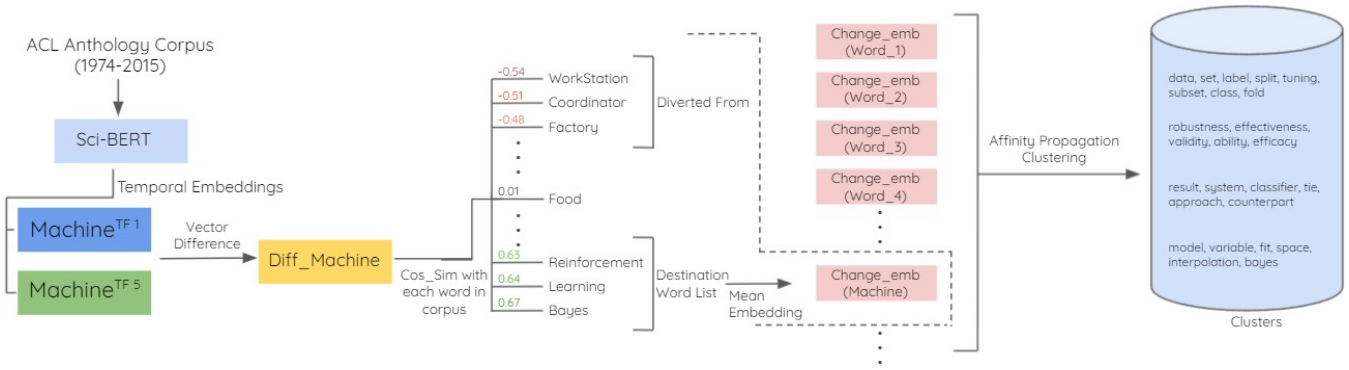

FIGURE 5: Workflow of the proposed change-oriented clustering approach focusing on the example word machine. Starting from the raw data in each time frame (shown on the left) the mean embedding vectors are obtained by averaging embeddings vectors of the "destination word list". The same process is done for all the words in our vocabulary. Finally, the clusters are obtained by applying AP clustering as shown on the right hand side.

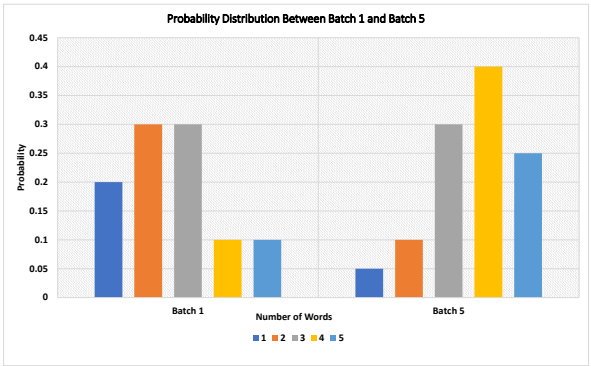

FIGURE 6: Example for a hypothetical cluster: Given a cluster we construct a probability distribution of how many sentences there are containing 1, 2, 3, 4 and 5 words in each batch.

y_i : discrete probability distribution for i_{th} cluster in 1^{st} time frame.
y_j : discrete probability distribution for j_{th} cluster in 5^{th} time frame.
U : a matrix of y_i for all i in 1^{st} time frame.
V : a matrix of y_j for all j in 5^{th} time frame.

generally expected from the computational linguistics related corpus.

2) Binary Difference Coherence

A coherent and informative cluster would contain words with similar movement patterns. To incorporate this idea we propose an additional metric that will take into account the similarity between the difference vectors for each pair of words in the cluster, i.e. quantifying the similarity in their movement patterns. The formula to calculate binary difference coherence for k^{th} cluster (B_k) is given below. It computes the average value of cosine similarity between difference vectors ($d_i = w_i^1 - w_i^5$) of each pair of member word for a cluster. The top 5 clusters sorted in a decreasing order of this metric are given in Table 5.

$$B_k = \frac{2}{n \times (n-1)} \times \sum_{i=1}^{n} \sum_{j=1, i \neg j}^{n} similarity(w_i^1 - w_i^5, w_j^1 - w_j^5) \quad (5)$$

TABLE 4: Top-ranked clusters using Earth Mover's Distance.

$Value$	Terms
2317.15	data, set, label, split, tuning, subset, class, fold, generalization, validation
2068.34	standard, test, real, deployment, trial
1737.55	model, variable, fit, space, interpolation, bayes, predictor, cue, cache, novelty
1611.03	integer, factor, permutation, discount, inflation, multiplicity
1376.89	submission, pipeline, evaluation, setting, official, configuration, setup, backup
1374.19	result, system, classifier, tie, approach, counterpart, detector

TABLE 5: Ranking of clusters using Binary Difference Coherence.

$Value$	Terms
0.3973	robustness, effectiveness, validity, ability, contribution, efficacy, viability
0.3972	seven, six, nine, twelve, thirteen
0.3971	graph, sentence, path, verb, redundancy, authority, occurrence, closure, equivalence, credibility
0.3845	retrieval, similarity, index, scheme, associate, judge, coherence, relatedness, relevance, utility, engagement
0.3797	small, reduction, reducing, narrow, lowering

F. REPRESENTING CHANGES BY SENTENCES

The clusters formed in the proposed change-oriented clustering represent the direction of semantic drift of terms from the first time-frame to the last. To better understand the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

meaning of these clusters, we propose the methodology of picking up sentences which represent the clusters from both the time frames. As considering the entire corpus for possible sentence candidates would have been computationally expensive, we restrict to a smaller set of sentences. We make a list of *N* sentences (=500) for each cluster based on the number of words that are common in a sentence and the cluster on a hierarchical basis. This means that if there exist *M (M<N)* sentences, with the maximum number of words in common with the cluster being *k*, then the rest *N-M* sentences will be selected with *(k-1)* words in common with cluster, and similar process is done down the hierarchy if the list is incomplete. For example, if the maximum number of words common in a sentence and cluster is 4, with 200 sentences following this property, then the rest of the 300 sentences will be taken with 3 words and then 2 words in common with the sentence.

For each cluster, this list is formed for both the first and the last time frame each containing *N* sentences. Now to select a sentence from each list, i.e. a pair to represent the cluster, we take the following factors into consideration:

1) Alignment of Cluster Words (Aln): We pick up a sentence pair which has the maximum words in common with cluster for better understanding and comparison of semantic change, thus the Alignment score of a pair of sentence is the number of words they have in common.
2) Variation in Sentence Meaning (Var): This factor will contribute for selecting a pair with distinct semantic properties. This is desirable for showing the change of sentences over time. It is represented as a cosine distance between the embedding of sentences. Sentence embeddings were obtained using above mentioned SciBERT model fine-tuned on the ACL Anthology dataset.

The score(s) assigned to a pair of sentences is given as:

$$s(a,b) = Aln(a,b)^i \times Var(a,b)_i^{1-i} \tag{6}$$

where,

a	: sentence from top N sentences of time-frame 1
b	: sentence from top N sentences of time-frame 5
$Aln(a,b)$	: Alignment of Cluster Words
$Var(a,b)$	: Variation in Sentence meaning
i	: Weight factor

IV. EXPERIMENTAL SETTINGS

We next analyze in this section the proposed approach using the ACL Anthology corpus. As to the best of our knowledge there are no similar baselines available for our task, we modify existing common semantic summarizing approaches to extract trending clusters from them. These methodologies include Latent Dirichlet Allocation (LDA), Word2Vec embedding based clustering and Latent Semantic Analysis (LSA) modelling. In this section we discuss the experiment design to extract trending clusters from these methodologies.

1) LDA Approach

Topic modeling approaches such as Latent Dirichlet Allocation (LDA) [73] have been frequently used in the past for capturing changes in topics. LDA is a generative probabilistic model for collections of discrete data (e.g., text corpora) with the goal to map all the documents to topics such that the words in each latent topic tend to co-occur with each other. In our experiment we utilize LDA[5] to monitor the change of topic importance from the time frame 1 to 5 and we compare the results to the proposed change-oriented clustering. LDA model was trained on the combined dataset of documents from the time frame 1 & 5, with fine-tuned hyper-parameters such as the number of topics equal to 43 with 20 passes through corpus along with filtering words that occur in less than 30 and more than 75% of the documents (other parameters had default values). The number of topics was decided on the basis of C_v coherence score [74] which attained the maximum value of 0.5517 with topics as 43. C_v coherence captures the degree of semantic similarity between high scoring words in the topic. It is based on a sliding window, one-set segmentation of the top terms and an indirect confirmation measure that uses normalized pointwise mutual information (NPMI) and the cosine similarity. LDA assumes documents are produced from a mixed set of topics. Words are then generated by these topics based on their probability distribution. Given a set of documents, LDA backtracks and tries to figure out what topics would create those documents in the first place. LDA outputs a document-topic matrix representing the importance of these topics per each document. To compute relevant topics for each time-frame, the topics were sorted in the increasing order of the difference between their importance scores of topics in time frame 1 and ones in time frame 5. The top 10 topics that gained most importance (i.e., become more dominant in the recent time) were then selected as the results.

2) LSA Approach

Similar to LDA, Latent Semantic Analysis (LSA) is a topic modelling algorithm which leverages the context around the words to capture the hidden topics in a text. LSA works based on a Bag of words (BOW) approach, where the dataset is converted into a term document matrix in which the order of the terms is neglected (only term frequency in document matters). The model was trained on a combined dataset of the first and last time frame. Topics were ranked using the document-term matrix which provides an importance score of a term for each document. For a document, this score is accumulated for all the terms in a topic (cluster), thus representing the significance of the cluster to document in mathematical terms. Clusters were then ranked according to the increasing value of their scores in time frame 5's documents compared to time frame 1's documents.

[5]We used Gensim implementation: https://radimrehurek.com/gensim/models/ldamodel.html

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

3) Word2Vec Approach

Word2Vec approach for semantic evolution was used in several works [47, 53] and it qualifies as a good baseline for evaluating the effectiveness of contextualised embedding based clustering. We train a skipgram model using the gensim library's Word2Vec module. Word embedding were randomly initialised at the beginning of the experiment, which were then trained on the preprocessed data from time frame 1. After extracting word embeddings for the first time frame, they were fine-tuned with the dataset from time frame 5. The difference-based clustering as described before was used to form the clusters which were then ranked and the cluster quality was evaluated manually. The results can be found in Table 6.

V. EXPERIMENTAL RESULTS

This section comprises result analysis and discussion for all the methodology described in the sections above. Subsection VA illustrates both qualitative (scored by reviewers) and intrinsic evaluation for clusters obtained by the proposed Bert-based methodology and the baselines. We then discuss the movement patterns of the obtained clusters in Subsection VB. We end this section by discussing the results of experimenting with sentence selection methods introduced above.

A. CLUSTER GENERATION RESULTS

Table 6 shows the comparison between the cluster scores for our proposed methods, which are Earth Mover's Distance based and & Binary Difference Coherence (BDC) based semantic change-oriented clustering vs. the LDA, LSA and Word2Vec methods which are used as the baselines. 3 expert reviewers, who have worked in NLP field for at least 5 years, were asked to score the top 10 clusters returned by each method on the basis of their quality using 1-5 Likert scale (1 is meaningless, 2 is rather meaningless, 3 is somewhat meaningful, 4 is rather meaningful, while 5 is meaningful result). The average score by judges was taken as the final score for each cluster. The results are shown in Table 6 for the top 1, 3, 5, 10, 15 and 20 clusters from both the rankings. We can see that EMD-based Change-oriented clustering outperforms all the baselines in all the cases with a maximum score improvement of 40% along with an average score improvement of 21.17%. Additionally, a common trend of decreasing scores when moving from the top-1 to top-20 signifies the effectiveness of the proposed EMD based ranking. Figure 7 visualizes the results from Tab. 6 for ease of comparison.

TABLE 6: Experimental results of the top ranked clusters.

Approach	Top 1	Top 3	Top 5	Top 10	Top 15	Top 20
LDA	2.66	2.33	2.80	2.96	3.07	3.05
W2V	3.00	2.22	2.73	2.43	2.38	2.42
LSA	1.66	1.66	2.26	2.20	2.09	2.03
BDC	2.33	2.22	2.53	2.00	2.24	2.18
EMD	**4.66**	**3.88**	**3.53**	**3.76**	**3.71**	**3.68**

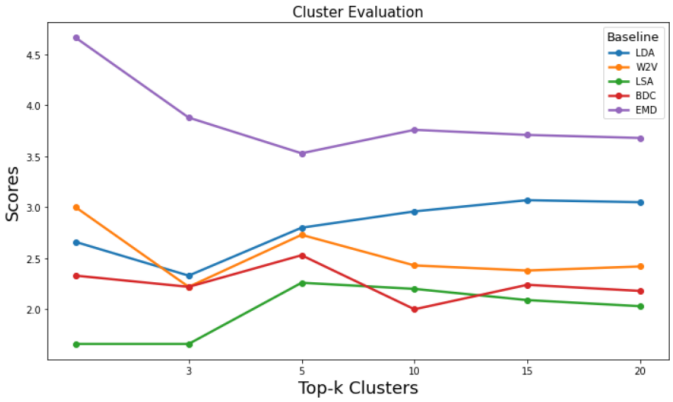

FIGURE 7: Visualization for cluster generation's experimental results.

Along with the above qualitative evaluation, we also perform quantitative cluster evaluation. As there are no ground truth clusters available, we utilize intrinsic measurements to evaluate cluster performance. Intrinsic methods evaluate the structure of obtained clusters and their relations to each other. We compute the results of Silhouette Coefficient and Calinski-Harabasz Index. Silhouette Coefficient uses pairwise distances between all the objects in a cluster, along with measuring the separation with the average distance of objects to alternative clusters. Calinski-Harabasz(CH) Index is the ratio of the sum of inter-clusters' dispersion and of intra-cluster dispersion for all clusters. Formula for CH index is given in Equation 7. Table 7 shows the results obtained for these metrics for the proposed Bert-based, Change-Oriented Term Grouping along with 3 baselines that were introduced in Section IV. Our methodology outperforms all the baselines for both the Silhouette Coefficient and Calinski-Harabasz Index. This proves the effectiveness of Change-Oriented Term Grouping as the evaluation is done on all the clusters for a particular methodology, unlike the human based evaluation shown in Table 6 which takes only the top ranked clusters into consideration.

$$CH = \frac{\sum_i d^2(c_i, g)/(N-1))}{\sum_i \sum_{x \epsilon C_i} d^2(c_i, g)/(n-N)} \tag{7}$$

where,

n	: number words
N	: number of clusters
c_i	: center of i^{th} cluster
g	: mean embedding for all words
$d(x, y)$	: distance between point x and y

B. CLUSTER MOVEMENT ANALYSIS

As defined in the above sections, each cluster is a group of words which comes from diverse meanings and converges to a similar meaning or expression. To intuitively understand the meaning change embedded in a cluster, we associate few words with each cluster which it moved to. These words represent the semantic meaning for a cluster's temporal change.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

TABLE 7: Experimental results of automatic cluster evaluation.

Approach	Silhouette Coef.	Calinski Harabasz
LDA	0.19	159.4
W2V	0.16	140.3
LSA	0.05	28.2
Change Oriented Clust.	**0.28**	**236.7**

To create such a set, we utilize Word movement pattern (Tab (3)) of each cluster word which shows the past and present meanings of a word. As the words constituting a cluster can have originally diverse meanings, the *moved from* column of table 3 has no significance here, thus we will only be creating the *Destination Word List* set as the words in a cluster have similar final meaning. The *Destination Word List* lists of each word in a cluster are grouped and the terms occurring with highest frequency in the grouped list are taken as *Destination Word List* list for the cluster as they represent the majority of changing concepts. The results are provided in Table 8. Table 8 should help in better understanding the cluster movement patterns by showing several examples.

TABLE 8: Cluster movement pattern examples.

Cluster	Destination Word List
show, mean, match, discrepancy, arithmetic, min, hypothetical, performance, efficiency	average, harmonic, marc, ro, cosine, sag, accuracy, kappa, elastic
model, variable, fit, space, interpolation, bayes, predictor, cache, novelty	train, descent, regression, classifier, learning, vector, machine, regularization, likelihood
words, representative, entail, capital, polarity, genre, passage, diversify	sentiment, entity, retrieval, document, similarity
score, agreement, correlation, slope, prevalence, coefficient	rate, alpha, performance, kappa, show, rho, rates, score, deviation
problem, inference, projection, modeling, framework, programming, penalty, learning, likelihood, augment	descent, bayes, elastic, boost, ridge, induction, method, loss, radial, plane
final, binary, feature, combine, anomaly, boundary, descriptor, cascade, discrimination	classifier, train, accuracy, performance, ensemble, imbalance, baseline, learning, validation
reference, public, development, external, gathering, primary, synthetic, creation, disposal	benchmark, gold, baseline, submission, setup, genre

C. ANALYSIS OF APPROACHES FOR SELECTING SENTENCES

To validate the usefulness of the proposed sentence scoring metrics, we perform ranking of all the sentence pairs for a cluster using our two proposed metrics. The results were then manually reviewed. Two criteria were kept into consideration while reviewing the sentences:

1) The representation of cluster meaning: Evaluating sentences on their ability to explain or elaborate cluster meaning.
2) Representation of Semantic Change: this is a quantitative representation of how well a sentence pair expresses the meaning change of cluster words from the time-frame 1 to 5.

Three reviewers assigned scores based on Likert scale (1-5) where 5 means highest rating. The average scores by the reviewers were used as the final score. The results are shown in Tab. 9 (representation of cluster meaning) and 10 (representation of semantic change).

TABLE 9: Scores for different sentence selection settings with regards to representing cluster meaning.

Score	Top 1	Top 3	Top 5	Top 10
Var (i=0)	3.67	3.44	3.26	2.97
Mixed (i=0.5)	3.67	3.55	3.13	3.03
Aln (i=1)	4.00	3.77	3.53	3.17

TABLE 10: Scores for different sentence selection settings with regards to representing semantic change.

Score	Top 1	Top 3	Top 5	Top 10
Var (i=0)	3.33	3.00	3.13	3.23
Mixed (i=0.5)	2.67	2.11	2.27	2.16
Aln (i=1)	2.33	2.11	1.93	2.03

The weight factor (i) in Eq. 6 is used to vary the contribution of the individual scoring metrics into ranking the sentence. When $i=0$, only the Variation in Sentence Meaning is considered while when $i=1$ only the Alignment of cluster words is considered. We also provide a reviewer score for $i=0.5$ for a better understanding of results. The results from Table 9 and 10 are in accordance with the theoretical results as the *Var(a,b)* score which selects dissimilar sentences performed the best at representing semantic change (Table 10). *Aln(a,b)* which selected similar sentences scored highest at representing cluster meaning (Table 9).

Table 11 showcases a few examples of the clusters and their sentence pairs. Several trends can be observed from the table such as the use of *web* before the development of WWW, the meaning shift of *mining* from extracting minerals to data and features. Similarly, the increase in the use of *gradient* as *gradient descent* can also be observed. This evidence, albeit perhaps less nuanced, can be easily interpreted and proves the effectiveness of our proposed cluster formulation and ranking approach as well as our sentence selection method.

VI. LIMITATIONS

In this work we aim at detecting and characterizing major changes on the level of the entire collection by using approaches from the field of semantic word evolution analysis. We note that the next stage of this work would be to detect particular time points of changes around which the concepts were diverging significantly.

We would like to note that using frequent terms as the main criteria to identify important terms might not be suitable in domain specific scenarios. A systematic methodological

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135051, IEEE Access

TABLE 11: Examples of extracted sentence pairs for cluster descriptions.

Cluster	Sentence from time-frame 1	Sentence from time-frame 5
machine, field, support, vector, active, classification, triplet, vision, deep, policy, reinforcement	One direct reason to put the two previously separate kinds of functionality into a single system was to **support** the knowledge-based machine aided translation environment which involves an interactive human editor who uses an interface to help the **machine** understand the source text.	For **classification**, we employed a performant **support vector machine**-based (SVM) technique that has been used in a range of studies.
web, predicate, document, links, authorship, title, term, article, taxonomy, summary, characteristic, patent	In a figurative sense, an english **article** is a **web** of concepts woven together according to the rules of english grammar.	When people read news **article**, **web** pages and other **documents** online they may encounter named entities which they are not familiar with and therefore would like to look them up in an encyclopedia.
mining, mapping, detection, scarcity, cur, integration, dynamics, checker, assurance, miner	The scientists has won the first round in an effort to save the three children in a **mining** town.	Implicit feature **detection**, also known as implicit feature identification, is an essential aspect of feature-specific opinion **mining** but previous works have often ignored it.
loss, activation, cost, oracle, conversion, machinery, classic, corruption	For example an engineer might explore requesting a dlc **activation** for a given site and **cost** or changing a fiber **activation** time.	We chose rectifier as the **activation** function and the logarithmic **loss** function for NNs.
software, branch, library, script, package, toolbox, viewer, apache	An **apache** myth and three decameron stories which were of comparable difficulty at the sentence level were presented.	For pre-processing the authors used **apache** opennlp2 **library** which is a machine learning based **toolkit**.
objective, formulation, batch, gradient, expectation, prior, derivative, update, bias, square, regular, stopping, reward	The telephone poles heights form a **gradient** that correlates with their locations on the road.	We use stochastic **gradient** descent which requires computing the **derivative** of the **objective** function with respect to each parameter for each training example.

framework should be developed to correctly identify domain-based specific terms. For example, a prior knowledge in the form of a dictionary of medicine terms can be used to guide the summarization task on the medicine related dataset.

In addition to that, as a pre-trained model is used in this research, some specific contexts might be missed during the modelling task. However, as SciBERT model that we used in this paper is derived from a corpus that has 18% computer science related contents, we assume that our analysis is based on a model related to current research in computer science.

BERT is a large model with around 110M parameters, thus the time and space complexity for BERT-based methodology are naturally high (>20X) compared to baseline methods like LDA, W2V and LSA. However, as this is a one-time process that can be done in offline settings, the quality of results can be prioritized over the time-complexity. Our proposal allows obtaining 21.17% better results than the other compared methods. With regards to using a pre-trained model as our basis model, training a new BERT model is a computationally expensive process (e.g., SciBERT according to its authors took one week to complete the training process). Hence, at this stage, building and executing the training stage on a new dataset is still complex within the context of this research.

VII. CONCLUSIONS

We proposed in this paper a method to identify groups of terms as well as representative sentences from a large scientific document corpora that are indicative for major semantic evolution of the corpora. Such results can help in better understanding the document collection and the changes that are latent in it over time.

The approach we propose is based on analyzing temporal changes in semantic values of terms and on extracting as well as grouping their drifting patterns. While the semantic evolution analysis has been done so far on the level of individual words, in this work we study the semantic changes in the entire document collection. We are the first to propose grouping terms by the semantic direction they drift towards over time and to use such proposed clustering for the purpose of temporal summarization of the underlying document collection. We also introduce a novel method to select indicative sentence pairs that through the comparison provide more specific indication of important changes occurring in the dataset over time. Finally, we evaluate our approach on the ACL anthology corpus and discuss the results and show sample clusters.

In the future, we plan to perform more experiments on datasets from other domains. We will also look for the possibility of applying supervised approaches to this task. However, due to the lack of available datasets unsupervised techniques such as the one introduced in this paper are the most reasonable at present.

REFERENCES

[1] Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth D Trippe, Juan B Gutierrez, and Krys Kochut. Text summarization techniques: a brief survey. arXiv preprint arXiv:1707.02268, 2017.

[2] Mahak Gambhir and Vishal Gupta. Recent automatic text summarization techniques: a survey. Artificial Intelligence Review, 47(1):1–66, 2017.

[3] Hans Peter Luhn. The automatic creation of literature

abstracts. IBM Journal of research and development, 2(2):159–165, 1958.

[4] Nina Tahmasebi, Lars Borin, and Adam Jatowt. Survey of computational approaches to lexical semantic change, 2019.

[5] Andrey Kutuzov, Lilja Øvrelid, Terrence Szymanski, and Erik Velldal. Diachronic word embeddings and semantic shifts: a survey. arXiv preprint arXiv:1806.03537, 2018.

[6] Isaac G Councill, C Lee Giles, and Min-Yen Kan. Parscit: an open-source crf reference string parsing package. In LREC, volume 8, pages 661–667, 2008.

[7] Steven Bird, Robert Dale, Bonnie J Dorr, Bryan Gibson, Mark Thomas Joseph, Min-Yen Kan, Dongwon Lee, Brett Powley, Dragomir R Radev, and Yee Fan Tan. The acl anthology reference corpus: A reference dataset for bibliographic research in computational linguistics. 2008.

[8] Dragomir R Radev, Eduard Hovy, and Kathleen McKeown. Introduction to the special issue on summarization. Computational linguistics, 28(4):399–408, 2002.

[9] Andrew Turpin, Yohannes Tsegay, David Hawking, and Hugh E Williams. Fast generation of result snippets in web search. In Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, pages 127–134, 2007.

[10] Rashmi Mishra, Jiantao Bian, Marcelo Fiszman, Charlene R Weir, Siddhartha Jonnalagadda, Javed Mostafa, and Guilherme Del Fiol. Text summarization in the biomedical domain: a systematic review of recent research. Journal of biomedical informatics, 52:457–467, 2014.

[11] Aqil M Azmi and Nouf I Altmami. An abstractive arabic text summarizer with user controlled granularity. Information Processing & Management, 54(6):903–921, 2018.

[12] Vishal Gupta and Gurpreet Singh Lehal. A survey of text summarization extractive techniques. Journal of emerging technologies in web intelligence, 2(3):258–268, 2010.

[13] Milad Moradi and Nasser Ghadiri. Different approaches for identifying important concepts in probabilistic biomedical text summarization. Artificial intelligence in medicine, 84:101–116, 2018.

[14] Yinfei Yang, Forrest Sheng Bao, and Ani Nenkova. Detecting (un) important content for single-document news summarization. arXiv preprint arXiv:1702.07998, 2017.

[15] Darshna Patel, Saurabh Shah, and Hitesh Chhinkaniwala. Fuzzy logic based multi document summarization with improved sentence scoring and redundancy removal technique. Expert Systems with Applications, 134:167–177, 2019.

[16] Francisco Afonso Raposo, David Martins de Matos, and Ricardo Ribeiro. An information-theoretic approach to machine-oriented music summarization. Pattern Recognition Letters, 123:75–81, 2019.

[17] K Girthana and S Swamynathan. Query-oriented patent document summarization system (qpss). In Soft Computing: Theories and Applications, pages 237–246. Springer, 2020.

[18] Keivan Kianmehr, Shang Gao, Jawad Attari, M Mushfiqur Rahman, Kofi Akomeah, Reda Alhajj, Jon Rokne, and Ken Barker. Text summarization techniques: Svm versus neural networks. In Proceedings of the 11th International Conference on Information Integration and Web-based Applications & Services, pages 487–491, 2009.

[19] Deepak Sahoo and Rakesh Chandra Balabantaray. Single-sentence compression using svm. In Soft Computing in Data Analytics, pages 483–492. Springer, 2019.

[20] Mahmood Yousefi-Azar and Len Hamey. Text summarization using unsupervised deep learning. Expert Systems with Applications, 68:93–105, 2017.

[21] Akanksha Joshi, E Fidalgo, E Alegre, and Laura Fernández-Robles. Summcoder: An unsupervised framework for extractive text summarization based on deep auto-encoders. Expert Systems with Applications, 129:200–215, 2019.

[22] Guy D Rosin and Kira Radinsky. Generating timelines by modeling semantic change. In Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL), pages 186–195, 2019.

[23] Goran Glavaš and Jan Šnajder. Event graphs for information retrieval and multi-document summarization. Expert systems with applications, 41(15):6904–6916, 2014.

[24] Yating Zhang, Adam Jatowt, and Katsumi Tanaka. Causal relationship detection in archival collections of product reviews for understanding technology evolution. ACM Transactions on Information Systems (TOIS), 35(1):1–41, 2016.

[25] Michael Färber and Adam Jatowt. Finding temporal trends of scientific concepts. In BIR@ ECIR, pages 132–139, 2019.

[26] Michael Färber, Chifumi Nishioka, and Adam Jatowt. Scholarsight: visualizing temporal trends of scientific concepts. In 2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL), pages 438–439. IEEE, 2019.

[27] Steven Bird, Ewan Klein, and Edward Loper. Natural language processing with Python: analyzing text with the natural language toolkit. " O'Reilly Media, Inc.", 2009.

[28] Nitin Indurkhya and Fred J Damerau. Handbook of natural language processing, volume 2. CRC Press, 2010.

[29] Lanting Fang, Luu Anh Tuan, Siu Cheung Hui, and Lenan Wu. Syntactic based approach for grammar question retrieval. Information Processing & Management, 54(2):184–202, 2018.

[30] Erwin Marsi and Emiel Krahmer. Automatic analysis of semantic similarity in comparable text through syntactic tree matching. In Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 752–760, 2010.

[31] Josef Steinberger and Karel Jezek. Using latent semantic analysis in text summarization and summary evaluation. Proc. ISIM, 4:93–100, 2004.

[32] Kaiz Merchant and Yash Pande. Nlp based latent semantic analysis for legal text summarization. In 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pages 1803–1807. IEEE, 2018.

[33] William C Mann and Sandra A Thompson. Rhetorical structure theory: Description and construction of text structures. In Natural language generation, pages 85–95. Springer, 1987.

[34] Johanna D Moore and Martha E Pollack. A problem for rst: The need for multi-level discourse analysis. Computational linguistics, 18(4):537–544, 1992.

[35] Mohamed Abdel Fattah. A hybrid machine learning model for multi-document summarization. Applied intelligence, 40(4):592–600, 2014.

[36] MS Patil, MS Bewoor, and SH Patil. A hybrid approach for extractive document summarization using machine learning and clustering technique. International Journal of Computer Science and Information Technologies, 5(2):1584–1586, 2014.

[37] Jin-ge Yao, Xiaojun Wan, and Jianguo Xiao. Compressive document summarization via sparse optimization. In Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015.

[38] Piji Li, Wai Lam, Lidong Bing, Weiwei Guo, and Hang Li. Cascaded attention based unsupervised information distillation for compressive summarization. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2081–2090, 2017.

[39] Adam Jatowt and Kevin Duh. A framework for analyzing semantic change of words across time. In IEEE/ACM Joint Conference on Digital Libraries, pages 229–238. IEEE, 2014.

[40] Wayne Xin Zhao, Jing Jiang, Jing He, Dongdong Shan, Hongfei Yan, and Xiaoming Li. Context modeling for ranking and tagging bursty features in text streams. In Proceedings of the 19th ACM International Conference on Information and Knowledge Management, CIKM '10, page 1769–1772, New York, NY, USA, 2010. Association for Computing Machinery.

[41] Xiongjun Liang, Wei Chen, and Jiajun Bu. Bursty feature based topic detction and summarization. In 2010 2nd International Conference on Computer Engineering and Technology, volume 6, pages V6–249. IEEE, 2010.

[42] Xiongjun Liang, W. Chen, and J. Bu. Bursty feature based topic detction and summarization. In 2010 2nd International Conference on Computer Engineering and Technology, volume 6, pages V6–249–V6–253, 2010.

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

[44] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

[45] Bai Xue, Chen Fu, and Zhan Shaobin. A study on sentiment computing and classification of sina weibo with word2vec. In 2014 IEEE International Congress on Big Data, pages 358–363. IEEE, 2014.

[46] M Ali Fauzi. Word2vec model for sentiment analysis of product reviews in indonesian language. International Journal of Electrical and Computer Engineering, 9(1):525, 2019.

[47] Muhammad Syafiq Mohd Pozi, Adam Jatowt, and Yukiko Kawai. Temporal summarization of scholarly paper collections by semantic change estimation: Case study of cord-19 dataset. In Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, pages 459–460, 2020.

[48] Milad Moradi, Maedeh Dashti, and Matthias Samwald. Summarization of biomedical articles using domain-specific word embeddings and graph ranking. Journal of Biomedical Informatics, page 103452, 2020.

[49] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pages 1532–1543, 2014.

[50] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. Fasttext. zip: Compressing text classification models. arXiv preprint arXiv:1612.03651, 2016.

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

[52] Shenhao Jiang, Animesh Prasad, Min-Yen Kan, and Kazunari Sugiyama. Identifying emergent research trends by key authors and phrases. In Proceedings of the 27th International Conference on Computational Linguistics, pages 259–269, 2018.

[53] A. Dridi, M. M. Gaber, R. Muhammad Atif Azad, and J. Bhogal. Leap2trend: A temporal word embedding approach for instant detection of emerging scientific trends. IEEE Access, 7:176414–176428, 2019.

[54] Stefania Degaetano-Ortlieb and Jannik Strötgen. Diachronic variation of temporal expressions in scientific writing through the lens of relative entropy. In International Conference of the German Society for Computational Linguistics and Language Technology, pages 259–275. Springer, 2017.

[55] Yijun Duan and Adam Jatowt. Across-time comparative summarization of news articles. In Proceedings of the

Twelfth ACM International Conference on Web Search and Data Mining, pages 735–743, 2019.

[56] Chengyu Wang, Xiaofeng He, and Aoying Zhou. Event phase oriented news summarization. World Wide Web, 21(4):1069–1092, 2018.

[57] Muhammad Syafiq Mohd Pozi, Yukiko Kawai, Adam Jatowt, and Toyokazu Akiyama. Sketching linguistic borders: Mobility analysis on multilingual microbloggers. In Proceedings of the 26th International Conference on World Wide Web Companion, pages 825–826, 2017.

[58] M Kishore Kumar and Sanampudi Suresh Kumar. Temporal summarization for online news articles. In 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), pages 1376–1379. IEEE, 2017.

[59] Mayank Singh, Pradeep Dogga, Sohan Patro, Dhiraj Barnwal, Ritam Dutt, Rajarshi Haldar, Pawan Goyal, and Animesh Mukherjee. Cl scholar: The acl anthology knowledge graph miner. arXiv preprint arXiv:1804.05514, 2018.

[60] Ruifang He, Yang Liu, Guangchuan Yu, Jiliang Tang, Qinghua Hu, and Jianwu Dang. Twitter summarization with social-temporal context. World Wide Web, 20(2):267–290, 2017.

[61] Chen Lin, Timothy Miller, Dmitriy Dligach, Steven Bethard, and Guergana Savova. A bert-based universal model for both within-and cross-sentence clinical temporal relation extraction. In Proceedings of the 2nd Clinical Natural Language Processing Workshop, pages 65–71, 2019.

[62] Chen Lin, Timothy Miller, Dmitriy Dligach, Farig Sadeque, Steven Bethard, and Guergana Savova. A bert-based one-pass multi-task model for clinical temporal relation extraction. In Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing, pages 70–75, 2020.

[63] Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-Wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. Scientific data, 3(1):1–9, 2016.

[64] William F Styler IV, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C De Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, et al. Temporal annotation in the clinical domain. Transactions of the Association for Computational Linguistics, 2:143–154, 2014.

[65] Rujun Han, Mengyue Liang, Bashar Alhafni, and Nanyun Peng. Contextualized word embeddings enhanced event temporal relation extraction for story understanding. arXiv preprint arXiv:1904.11942, 2019.

[66] Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. A corpus and evaluation framework for deeper understanding of commonsense stories. arXiv preprint arXiv:1604.01696, 2016.

[67] Xianchao Wu. Event-driven learning of systematic behaviours in stock markets. arXiv preprint arXiv:2010.15586, 2020.

[68] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. arXiv preprint arXiv:1903.10676, 2019.

[69] Anna Hätty, Dominik Schlechtweg, and Sabine Schulte im Walde. Surel: A gold standard for incorporating meaning shifts into term extraction. In Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (* SEM 2019), pages 1–8, 2019.

[70] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. Advances in neural information processing systems, 30:5998–6008, 2017.

[71] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. arXiv preprint arXiv:1801.06146, 2018.

[72] Brendan J Frey and Delbert Dueck. Clustering by passing messages between data points. science, 315(5814):972–976, 2007.

[73] D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation. the Journal of machine Learning research, 3:993–1022, 2003.

[74] Michael Röder, Andreas Both, and Alexander Hinneburg. Exploring the space of topic coherence measures. In Proceedings of the eighth ACM international conference on Web search and data mining, pages 399–408, 2015.

NAMAN PAHARIA is an undergraduate student from the Indian Institute of Technology, Kharagpur with a major in Electrical Engineering along with a minor in Computer Science and Engineering. His research interests include Reinforcement Learning, Natural Language Processing and Information Retrieval.

MUHAMMAD SYAFIQ MOHD POZI is an academician and a machine learning researcher in School of Computing, Universiti Utara Malaysia. He received the Bachelor of Computer Science with Honours from Infrastructure University of Kuala Lumpur, Malaysia in 2012. He received his Doctor of Philosophy in Computer Science from Universiti Putra Malaysia in 2016. Then, he was appointed as a postdoctoral researcher in Kyoto Sangyo University, Japan and then, in Universiti Tenaga Nasional, Malaysia. His research interest is on modelling uncertainty in machine learning, optimization, computer vision and natural language processing. His research collaboration spans across various research domains, such as network security, social science and lately, medicine.

ADAM JATOWT is a Professor at the Computer Science department of the University of Innsbruck. He is also affiliated with the Digital Science Center at the University of Innsbruck. He received his Ph.D. from the University of Tokyo, Japan in 2005 and has worked as an Assistant and Associate Professor at Kyoto University. His research interests include broad topics in natural language processing, information retrieval, digital humanities and digital libraries. Adam has published over 150 research papers in international conferences and journals. He is on the editorial board of IP&M, JASIST, IJDL, JIIS and IEEE JSC journals.