DYNAMICTOC: Persona-based Table of Contents for Consumption of Long Documents

Anonymous ACL submission

Abstract

Long documents like contracts, financial documents, etc., are often tedious to read through. Linearly consuming (via scrolling or navigation through default table of content) these documents is time-consuming and challenging. These documents are also authored to be consumed by varied entities (referred to as persona in the paper) interested in only certain parts of the document. In this work, we describe DYNAMICTOC, a dynamic table of content-based navigator, to aid in the task of non-linear, persona-based document consumption. DYNAMICTOC highlights sections of interest in the document as per the aspects relevant to different personas. DYNAMICTOC is augmented with short questions to assist the users in understanding underlying content. This uses a novel deep-reinforcement learning technique to generate questions on these persona-clustered paragraphs. Human and automatic evaluations suggest the efficacy of both end-to-end pipeline and different components of DYNAMICTOC.

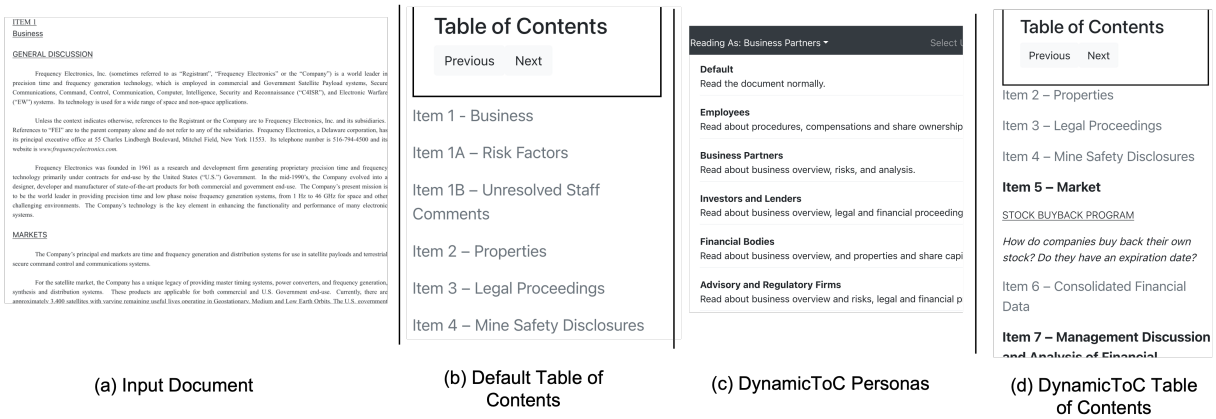
1 Introduction

Documents such as financial statements, reports and contracts are often long and comprehensive, replete with domain-specific description and information. They are meant to be consumed by several entities or personas, *e.g.* legal department of companies, customers or financial organizations such as banks. As these documents contain vital information about the business, the business personas are often required to read through and analyze the documents in details. These personas are often interested in different sections of the document, based on the business requirements. For example, employees might be interested in the stock programs of the company, whereas the lenders and investors would like to read through profit statements. The traditional technology to navigate long documents is through a Table of Contents (ToC) populated with the heading of each section and chapters. However, the Table of Contents does not show the information present in the underlying paragraphs of a section, and there is no way to highlight information relevant to different personas.

To this effect, we propose DYNAMICTOC, an intelligent table of contents-based navigator. DYNAMICTOC provides user the flexibility to choose the persona and read the document from its lens. For the current work, we focus on the finance and legal domain, and hence, personas are taken as commonplace entities like investors, lenders, financial bodies, etc. DYNAMICTOC highlights the relevant sections of the document as per the persona. For this, the input finance or contract document is segmented at the paragraph level and a cluster of "aspects or topics" is inferred for each para. These are then mapped to the interest topics of the personas. Further, DYNAMICTOC has a novel question-based guided experience, to enhance the visibility of underlying information. Studies have shown that questions are more intuitive and informative than headings and hence can provide a better understanding of what the paragraph talks about. The overall interface is shown in Figure 1.

2 Related Work

Document understanding is a critical and challenging task in information processing. There have been many related research works in this direction. Keyword detection (Liu et al., 2009; Tixier et al., 2016) & topic modeling (Blei et al., 2003) works aim is to describe the document by a few important words or topics for concise representation. The first step is to acquire a list of keyword candidates (e.g., n-grams or chunks) with heuristic methods (Hulth, 2003; Shang et al., 2018), then rank them in accordance with their importance to the document (Wu et al., 2005; Gollapalli and Caragea, 2014; Bougouin et al., 2013). Another task is compact and informative headline generation from a document (David and Zajic, 2003; Lopyrev, 2015). Text



(a) Input Document (b) Default Table of Contents (c) DynamicToC Personas (d) DynamicToC Table of Contents

Figure 1: A comparison of the DynamicToC-based experience with the default Table of Content. DynamicToC enables the user to choose a reading persona and enhances navigation through highlighting relevant sections. The relevant sections are supplemented with automatically generated questions to guide the user.

summarization is the process of generating natural language summaries from an input document retaining the most important information (Rush et al., 2015; See et al., 2017). Recently, an Outline Generation task was introduced by (Zhang et al., 2019) as a hierarchical structured prediction problem. Given a document, their aim is to first predict a sequence of section boundaries and then a sequence of section headings accordingly to come up with a Table of Contents for the same.

A related direction of work to ours is of aspect detection, which has been explored in the literature largely using user reviews for products. Early works focused on rule-based approaches using lexicons and dependency relations, and utilize manually defined rules to identify patterns and extract aspects (Qiu et al., 2011; Liu et al., 2016), which require domain-specific knowledge and human expertise. Supervised approaches formulate aspect extraction as a sequence labelling problem that can be solved by hidden Markov models (HMM) (Jin et al., 2009), conditional random fields (CRF) (Li et al., 2010; Mitchell et al., 2013; Yang and Cardie, 2012), and recurrent neural networks (RNN) (Wang et al., 2016; Liu et al., 2015). These approaches have shown better performance compared to the rule-based ones, but require large amounts of labelled data for training. Early unsupervised systems are dominated by Latent Dirichlet Allocation (LDA)-based topic models (García-Pablos et al., 2018; Shi et al., 2018; Alvarez-López et al., 2016). Recently, deep learning based topic models (Srivastava and Sutton, 2017; Luo et al., 2019; He et al., 2017; Shi et al., 2021) have shown strong performance in extracting coherent aspects in an unsupervised manner. None of the prior works on aspect detection have worked with contracts or financial documents that are quite long (50-100 pages) in comparison to user reviews. Even if we break the document at paragraph level, it can still go over tens of lines. Hence, the importance of word frequency is much more in our case. We bridge the gap between directly using the unsupervised aspect detection frameworks for the financial documents by adding a TF-IDF based weighing parameter while training. Moreover, there are no gold standards for aspect detection for contract or finance domain, hence, we use unsupervised clustering based metrics for validating the output detection.

Further, it has been shown that question-answers play a critical role in scientific inquiry, information-seeking dialogue, and knowledge acquisition (Hintikka and Saarinen, 1979; Stede and Schlangen, 2004). In a dialogue system, question generation is used to obtain specific information from the user or make the conversation more pleasant (Shukla et al., 2019; Saeidi et al., 2018). Hence, we hypothesize that augmenting the default ToC with Questions that give a high level overview of the paragraphs can enhance the reading experience of the users. Question generation can also be seen as a summarization or seq2seq task. Various pre-trained language models like BART (Lewis et al., 2020), PEGASUS (Zhang et al., 2020), etc., have shown excellent results for these tasks. Researchers have worked on top of these language models & proposed various rewards for QGen to optimize these models. (Kumar et al., 2019) has employed BLEU-based rewards, (Zhang and Bansal, 2019) have used

answerability rewards, whereas (Xie et al., 2020) has used a combination of fluency, relevance, and answerability rewards.

Previous question generation literature has focused on generating questions based on an entity, phrase, or sentence. In this work, we explore long-form question generation, i.e., question-based long text. We explore deep reinforcement learning techniques for the same. as they have shown competitive results in various natural language generation tasks such as summarization (Pasunuru and Bansal, 2018), style transfer (Liu et al., 2021; Goyal et al., 2021), question generation (Hosking and Riedel, 2019; Xie et al., 2020) etc. Motivated by this we use BART as our base model. As there is a lack of labeled question datasets in financial domain, to overcome the domain shift problem, we train the model with additional rewards in a reinforcement learning setup to make more suitable for a general domain.

There are several commercial products for reading documents across devices, but all of them have a fixed document navigation, based on chapters and headings. To the best of our knowledge, there is no prior art looking into providing an end-to-end persona-based navigation. The mentioned technologies address only part of the required solutions. Following are the key contributions of our work:

1. We propose a novel DYNAMICTOC technology to enable persona-based non-linear navigation for efficient consumption of long documents.

2. We extend the unsupervised aspect detection to long domain-specific documents by combining TF-IDF with aspect detection process to make it more robust and show the improvements experimentally.

3. We propose a method to generate questions based on the content of the paragraph maximizing the information coverage, entity correctness & answerability of the question.

4. We showcase the viability of our pipeline and evaluate it using metric-based and a human survey-based evaluation.

3 Datasets

SEC Filing: The SEC filing is a financial statement document submitted to the U.S. Securities and Exchange Commission. Public companies, certain insiders, and broker-dealers are required to make regular SEC filings. There are many types of documents available on the EDGAR website (Eg. 10-K, 10-Q, Form 4, etc.). For our work, we focused on the SEC 10-K documents available on the EDGAR website[1]. For a given company, the 10-k documents are available in HTML, XBRL and XML format. The complete submission text file for a 10-k document (XML) was used for parsing. We split the document content into different items ranging from item 1 to item 16. These 10-K documents range from 50 to 120 pages and contain multiple tables along with text paragraphs.

ELI5: We use the ELI5 dataset (Fan et al., 2019) to train the question generation model. ELI5 or Explain Like I'm 5, is a question-answer dataset scraped from the subreddit r/explainlikeimfive/. The subreddit rules encourage people to ask a question about any topic and get an answer for it. To maintain the dataset's quality, we only select those question-answer pairs with more than two upvotes. Note that no dataset for such question generation task exists for contractual and financial documents. Hence, we resort to use the ELI5 dataset for supervised training and use that model for inferencing on documents from a different domain.

DATASET	TRAIN SIZE	TEST SIZE
ELI5	100,000	10,000
r/AskLegal	-	10,067
r/AskEconomics	-	98

Table 1: Train/Test Statistics of question-answer pairs for the datasets used.

To test the model's performance on the domain-specific dataset, we also scrape question-answer pairs from two different subreddits - r/AskLegal and r/AskEconomics. As the name suggests, they contain questions (and answers) from the legal and economics domain respectively. Table 1 includes the statistics of the three datasets.

4 Methodology

In this section, different components of the DYNAMICTOC are described in details.

4.1 Aspect Detection

Aspect detection has been popularly used with analysing user reviews to understand their preferences. We leverage an unsupervised technique

[1] https://www.sec.gov/edgar.shtml

for aspect detection and extend it to a new use case - for the modelling of user profiles from a given document and using this info for segregating the document text based on the determined aspects.

Data Pre-processing: For the input SEC filings, text corresponding to each paragraph is obtained and considered as a separate data point. Extra information such as headings, sub-headings, blank lines, signature fields etc. are discarded. Along with this, any paragraph with less than 10 words are discarded. The text is pre-processed before training the model for aspect detection. We require three formatting styles for each paragraph which is consolidated in a single dictionary. (1) Tokenized words converted to lowercase characters. (2) The word stems of the text which has been lower-cased and tokenised. (3) The content words (meaningful words, like nouns, verbs, adjectives and adverbs) of the paragraph.

Proposed Asp-SSCL Method: Aspect detection aims at extracting interpretable aspects from the textual documents without human supervision. We propose an approach, *Asp-SSCL* based on self-supervised contrastive learning framework by (Shi et al., 2021) for aspect detection. We use the following steps for aspect detection from the contractual and financial documents:

1. Vocabulary formation and IDF indexing: First, we obtain a vocabulary for the whole corpus. This is sorted alphabetically and each word is given an index, so that corresponding IDF/word vectors can be easily referenced. 128-dimensional word vectors are generated on the corpus by a skip-gram model with an n-gram size of 5.

2. Weak Mapping: Prior aspect detection methods require a gold set labels for validating aspect model training, either through human supervision or rules-based mapping to gold set keywords. However, as no such gold aspect labels exist for our case, we first use text embedding via sentence transformers[2]. These are then clustered using K-means to obtain 20 clusters comprising of 10 keywords each, which are used for aspect mapping.

3. Contrastive Learning: The mapping and generated word vectors are then used for training the self-supervised contrastive learning method, (Shi et al., 2021), which outputs the final aspect clusters and keywords.

4. TF-IDF Weighing: Further, since these documents are text-heavy, we introduce a modification, **Asp-SSCL-TFIDF** which includes TF-IDF weighing term in the original implementation, to ensure rare but relevant words are considered as important as opposed to more frequently occurring words. Each word representation is modified by multiplying it with the TF-IDF score so that the algorithm can adapt to the financial corpus better.

[2]https://www.sbert.net/

4.2 Persona Mapping

The aspects generated on the corpus are used as dimensions that define the document. Each persona is expected to be interested in one or more of these dimensions. We call the mapping between multiple personas and multiple aspects as the persona space.

We consulted a domain expert (financial domain; specifically for SEC 10-K filings) to get an understanding of the parties or personas who read such documents and what kind of information are they generally interested in. We constructed a matrix listing out the various stakeholders of a general 10-K filing against the different sections of the document each stakeholder is interested in. The stakeholders were grouped together to form the personas used in DYNAMICTOC, *viz.* employees, business partners, investors and lendors, financial bodies and advisory regulatory firms. Similarly, the columns (headings) were grouped together according to similarity to create a mapping of topics of interest for each persona.

The aspects we got from our unsupervised technique were compared against the simplified column values from the constructed matrix. The columns with the greatest similarity (above a threshold) were associated with each persona. For getting the personas interested in each paragraph, the paragraphs were first tagged for aspect. From the resultant vector (which represents the confidence score of the text for each aspect), the combined score for each persona was calculated using the scores of its constituent aspects. This was used to segregate the paragraphs for enhanced document consumption.

Figure 2 shows the relevance of different content (columns) for the various stakeholders (the rows) for a 10K filing document. Groups of stakeholders are made that form the personas interested in the different parts of the document. The different columns are also grouped together as to indicate what kind of information those sections will contain. Thus, we can map these columns to the aspects we get from the Aspect Detection Module and determine if a particular persona is interested

Key stakeholders who would be interested in various aspects of Form 10-K	Business overview	Risk factors	Management Discussion and Analysis	Financial statements and supplementary data	Legal proceedings	Changes and disagreements with accountants on financial disclosure	Related transactions	Controls and procedures	Director and executive compensation	ESOPs/ Share ownership of management	Properties	Share capital	Quantitative and Qualitative disclosure on market risks	Principal accountant fees and services	Persona
Employees	NO				NO			YES			NO		NO	NO	1
Customers Suppliers and other partners Competitors	YES				NO			NO			NO		YES	NO	2
Potential investors Potential lenders Government departments and agencies	YES				YES			NO			YES		NO	NO	3
Inland Revenue Service (IRS) Banks and financial institutions Insurance companies	YES				NO			NO			YES		NO	NO	4
Securities and Exchange Commission (SEC) Stock exchanges (NASDAQ, NYSE, etc.) Present investors / shareholders Proxy advisory firms Business and investment / stock analysts Accounting regulatory bodies	YES				YES			YES			YES		YES	NO	5
Aspect			A			B			C			D		E	F

Figure 2: Table showing different stakeholders who would be interested in consuming SEC filings and corresponding sections of their interests. The rows are grouped together to form the five personas used in the work. Similar column topics have also been grouped for aspect mapping.

in that paragraph or not.

The thing to note here is, for financial documents, we were able to get some domain knowledge and leveraged it to obtain the persona space. But the technique we are proposing is generalizable to other domains as well. In the absence of domain specific knowledge, each aspect is a sufficiently distinct topic and can be treated as a proxy to personas. Hence, modelling of interests can be done directly on the basis of aspects in such cases.

4.3 Intelligent Navigation via Question Generation

It has been shown that question-answers play a critical role in scientific inquiry, information-seeking dialogue, and knowledge acquisition (Hintikka and Saarinen, 1979; Stede and Schlangen, 2004). Additionally, unstructured lists of "Frequently Asked Questions (FAQs)" are regularly deployed at scale to present information. On top, questions can provide a meaningful understanding of the document at a paragraph or section level which cannot be directly captured by a heading (or sub-heading). Therefore, to aid in document consumption, we generate long-form questions (i.e., questions based on paragraphs instead of entities) to enhance the navigation experience.

Model Architecture: We use an encoder-decoder architecture for the task of generating questions given the paragraph as context which essentially is a sequence-to-sequence task. Large pre-trained language models like BART (Lewis et al., 2020), PEGASUS (Zhang et al., 2020), etc., have shown excellent results in summarization tasks. Motivated by this, we employ BART as our underlying language model. The task is to generate questions covering the entire paragraph and summarize it capturing the most-salient information in form of a question. The BART model has shown promising results in abstractive summarization tasks, making it a natural choice.

We use ELI5 (Fan et al., 2019) dataset for training the model. The answer is provided as the input to the encoder-decoder model which is trained to generate the corresponding question and minimize the cross-entropy loss with respect to the ground truth. Although such supervised training is straight-forward, due to the domain shift from ELI5 to financial language, qualitative evaluations showed that the model produced some irrelevant questions, some entities were artificially induced (that it might have seen during the training time) and sometimes, it could not cover the entire paragraph. Hence, we augmented the vanilla BART with three additional rewards targeting the qualities we seek in the final generated questions. We call the resulting model Variant BART. Figure 3 shows the proposed pipeline. The following sections explain these rewards in detail:

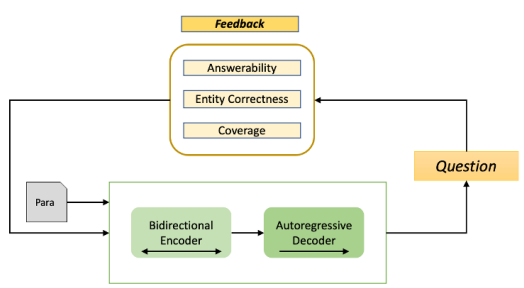


Figure 3: Training Variant BART for Question Generation with feedback rewards

- **Answerability Classifier Reward**: Qualitative evaluations showed some of the questions generated were not answerable by the para-

graph, making them unsuitable for the task of understanding the paragraph easily. To address this, we trained a classifier to judge the answerability of the question given the paragraph. Basically, the paragraph and the generated question would be fed as input to the classifier and the classifier predicts "1" if the question is answerable by the para, otherwise "0". The classifier is a fine-tuned Roberta model. We create data with both positive and negative samples to fine tune the Roberta model. We use the following strategy to create the training data for the binary classifier: (a) We select 10,000 random question-answer pairs from the ELI5 dataset. Note that in ELI5 data, some questions have multiple answer paras too. All these forms our positive samples. To create negative samples for a question Q', we take its corresponding answer para, A', and a random set of 100 question-answer pairs. We compare the similarity of the answer para (A') with all the 100 answers. Top 3 most similar answers are taken as the negative samples for the question Q'. (b) Given that style of ELI5 answers and Wikipedia paragraphs are very similar[3], we create negative samples in another way as well to introduce diversity. Wikipedia articles are chosen based on their similarity with ELI5 data and for each question, we compute the similarity score with each para in the wiki articles. The topmost similar paras are of interest, and we sample 10 most similar paras out of top 20 and call them our negative samples. Table 2 mentions the data statistics. The reward is computed using the equation: $R_{answerability} = Prob_{classifier}(1|P, Q)$

DATASET	TRAIN	TEST
#Unique Questions	10,000	1,000
Total Samples	208,871	21,045
Positive Samples	30,564	2,974
Negative Samples	178,307	18,071
Neg to Pos Ratio	5.83	6.04

Table 2: Dataset stats to train answerability classifier

- **Entity Correctness Reward**: The vanilla BART model can generated questions with hallucinated entities and names like Microsoft, Apple etc. even when there was no mention of them in the corresponding paragraph. To tackle this, we identify the named entities present in the generated question. If those entities appear in the passage, we give a reward of 1 else 0. If there is no entity in the generated question, a reward of 0.5 is given to the model. Mathematically, reward is given by:

$$R_{entity} = \begin{cases} 1, & \text{if } e(Q) \neq \phi, e(Q) \subseteq e(P) \\ 0, & \text{if } e(Q) \neq \phi, e(Q) \nsubseteq e(P) \\ 0.5, & \text{if } e(Q) = \phi \end{cases}$$

where, $e(.)$ denotes the entities in the question or paragraph.

- **Coverage Reward**: We observe that the output question did not cover the entire information present in the paragraph and instead focused on certain segments of it. We introduce this reward to improve information coverage. The idea is similar to the entity correctness reward. We first identify keywords from the paragraph using YAKE algorithm (Campos et al., 2018). Then we calculate the similarity of the generated question with these keywords. We use the Extended String Subsequence Kernel (ESSK) introduced in (Hirao et al., 2003) to calculate this similarity score. The idea is that YAKE would generate keywords from different parts of the paragraph. When we calculate the similarity of this keyword list with the generated question, we are encouraging the model to cover the entire paragraph. Thus, given a passage P and the generated question Q, the reward R is defined as follows:

$$R_{coverage} = ESSK(YAKE(P), Q)$$

(Lai et al., 2021) shows how to use rewards on top of language models for policy learning. We adopt the same setup. The policy gradient, $\nabla_\phi J(\phi)$ is given by:

$$\nabla_\phi J(\phi) = E[R \cdot \nabla_\phi log(P(y^s|x, \phi))] \quad (1)$$

where, R denotes reward value, ϕ represents the model parameters, x is the input paragraph and y^s is obtained by greedily maximizing the distribution of BART outputs at each timestep. Hence, the overall loss term for training the proposed Variant BART model becomes:

$$L_{total} = \lambda_{CE} \cdot L_{CE} + \lambda_{reward} \cdot L_{reward} \quad (2)$$

[3]https://yjernite.github.io/lfqa.html

5 Results & Discussion

In order to evaluate the different parts of the pipeline, we employ metric based evaluation schemes. Since there is no off-the-shelf criterion to evaluate all the stages of the pipeline together, we have provided independent evaluations for each of the sub-modules. However, the intended goal of our pipeline is to facilitate the readers in consuming long documents through the lens they deem most suitable for them. To facilitate an end-to-end evaluation and to understand whether the persona-based document segmentation with enhanced Table of Content is informative or not, we have conducted a small scale human evaluation. Both metric-based and human-based evaluations are discussed in the following subsections.

5.1 Metric-Based Evaluation

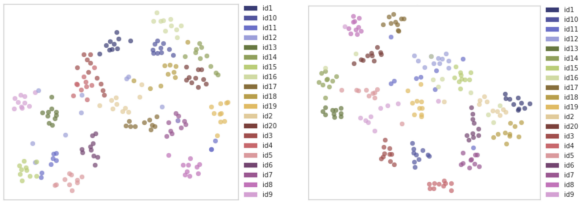
Aspect Detection: Figure 4 shows the t-SNE[4] clustering of the outputs of Asp-SSCL, Asp-SSCL-TFIDF, that are determined from the SEC-10K filing corpus. For a baseline comparison, we also plot the output for LDA (10 clusters) for the corpus. Essentially, each cluster is a bag of words indicating some vital theme that is mentioned in the corpus. We would want the clusters to be as independent from each other as possible as that would mean different kinds of information is captured by different clusters with minimal overlap. We observe that adding TF-IDF scores to the aspect detection module helps as the clusters' separation gets better as shown in the Figure 4. Further, for the baseline using LDA, the separation is not clear. Some of the examples of cluster keywords are shown in Figure 5.

Question Generation: Since no ground truth questions are available for the SEC-10K Filing dataset, we report the following evaluations for the question generation module. First, we report the "type" of questions that are generated using the Vanilla BART model and the Variant BART model that is trained with a combination of the rewards we added on top of it (Table 3) .

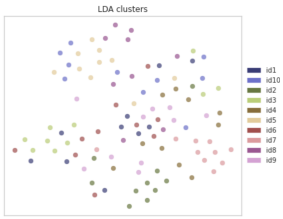
On analysing this table, we see that "What" questions are heavily generated on the 10K filing data. This suggests that the nature of 10K filing is such that the question asked about them is "What" type. We also observe that biases of training data are not creeping in the model, as the percentage of the "What" questions in the training dataset is four times less than the model's output. Similarly, the percentage of other questions is significant in the ELI5 dataset but is very small in our model's output. Thus, we can safely say that the model learns to generate questions and not mimic the ELI5 dataset.

Although we don't have the gold corpus for SEC filing dataset, we evaluate the performance of question generation model on the AskLegal and AskEconomics subreddits since we have the ground truth questions for them. We report the BLEU and ROUGE scores that are standard metrics in Natural Language Processing literature and are a measure of overlap or common n-grams between the gen-

[4] https://lvdmaaten.github.io/tsne/



(a) Asp-SSCL (b) Asp-SSCL with TF-IDF



(c) LDA

Figure 4: Aspect clusters using (a) default Asp-SSCL and (b) Asp-SSCL with TF-IDF and (c) LDA

LDA cluster words:
1. cost, fiscal year, revenue increase, expense, sale, million, primarily due
2. financial statement, report, control, intern, consolidate, account, audit, inform
3. rate, interest, value, invest, market, currency, hedge, risk, fair instrument
4. stock, share, common award, grant, option, incentive, per companies, date

Asp-SSCL cluster words:
1. vest, vesting, shares, vested, awards, bonus, equal, grant, month
2. regulations, laws, federal, act, rules, regulation, legislation, requirements, privacy, law
3. opportunities, strategy, initiatives, strategies, efficiencies, efforts focus opportunity, objectives
4. deferred, tax, unrecognized differences, recorded, depreciation, taxes, income, accumulated expense

Asp-SSCL-TFIDF cluster words:
1. damage, breaches, injury, disasters, breach, fines, damages, disruptions, failures, sanctions
2. assumptions, forward looking, estimates, inputs, hierarchy, qualitative, judgment, analysis
3. shares, awards, common, options, vesting, stock, voting, vested, award, vest
4. allowance, tax allowances, losses, adjustments, charges, deferred, differences, taxable, carry forwards

Figure 5: Examples of cluster words generated using LDA, Asp-SSCL and Asp-SSCL with TF-IDF.

Q Type	ELI5 Train Set (%)	Vanilla BART	Variant BART
What	12.93	54.28	44.75
Where	0.55	0.0	0.56
Why	36.23	11.47	11.93
How	21.88	30.18	37.44
Who	0.28	3.2	3.9
When	1.62	0.33	0.84
Other	26.51	0.54	0.56

Table 3: Type of questions generated by Vanilla BART & Variant BART on an SEC filing.

erated text and the ground truth. We also report the answerability score by feeding the generated question and input para to the classifier we trained (as mentioned in Reward 1 – Question Generation Section). Table 4 shows the corresponding results.

MODEL	BLEU	ROUGE-L	ANSWERABILITY
Vanilla (r/AskLegal)	0.274	0.278	95.92
Variant (r/AskLegal)	0.264	0.240	98.98
Vanilla (r/AskEconomics)	0.289	0.298	95.16
Variant (r/AskEconomics)	0.291	0.306	96.17

Table 4: Performance comparison of Vanilla and Variant BART on r/AskLegal r/AskEconomics subreddits

QUESTION	AVG SCORE (1-5)
How often do you come across a long document?	2.83
How often do you use a document reader?	4.33
How satisfied are you with the 'Default' reading experience ?	3.00
How would you rate the option to choose persona as an aid in reading the document?	**4.50**
How would you rate jumping to relevant parts of the document?	**4.16**
How would you rate the utility of presented questions?	**4.40**

Table 5: Results from the human experiment on using the Default Reading experience with DYNAMICTOC.

On closely analysing the paragraph, reference question, and the generated question, we observe the following three things: (i) There is more than one way to ask the same question and there could be multiple questions around the same topic. (ii) Input paragraphs may have more than one prominent topics. The generated question might be focused on one such topic, and the reference question is focused on another. (iii) Some answers/passages are unrelated to the question or require some background, and thus, the generated questions are very different from the actual question.

The above reasons explain the fluctuation in scores for Vanilla and Variant BART, and thus the answerability of the generated question becomes an important metric. The variant model trained with additional rewards has the highest answerability score across all the datasets. This suggests that including a coverage-based loss not only helps cover the information of the entire paragraph but also helps increase the generated question's answerability as different themes of the passages are covered.

5.2 Human Evaluation

We conducted a small human evaluation involving 8 participants (age - 27.8 ± 6.7, 2 females). The participants were technology workers internal to our organization. They were asked to play around with a web demo to experience the DYNAMICTOC, for different SEC filings. They were first shown the default section heading-based reading experience and then they choose the type of persona as whom they wish to consume the document. After this, they filled a questionnaire about their experience. The results of the survey are summarized in Table 5.

Some relevant comments from the survey are as follows - (i) How do we ensure that all the relevant information will be covered by the sections highlighted as important for a particular "persona"? (ii) Although questions generated are relevant, some of the why questions are not answered by the paragraphs they point to. (iii) Interesting experiment with possibly multiple use-cases. The first comment is actually true for all summarization tasks, hence, DYNAMICTOC does not disrupt the linear flow. The second feedback indicates scope for further research in the question generation space. The overall response is immensely positive and the scores of 4.50, 4.16 and 4.40 in Table 5 reflect the same.

6 Conclusion

In this work, we have proposed a novel DYNAMICTOC framework for consumption of long documents. Financial documents and contracts are high value documents for business entities, and are often long and complex. The default ToC-based reading experience is quite limited and there are immense opportunities to enhance the document consumption using intelligent technologies. We believe that DYNAMICTOC is one of the first works to pursue this exciting research direction, and would enable further exploration in the area.

For the future direction, DYNAMICTOC would benefit from a better supervised in-domain learning of aspect keywords and questions. We would also like to work on evaluation of the paragraph segmentation and mapping of personas to the aspects in future. A better understanding of personas or entities interested in consuming the document would help to generalize the work to different domains.

References

Tamara Alvarez-López, Jonathan Juncal-Martínez, Milagros Fernández-Gavilanes, Enrique Costa-Montenegro, and Francisco Javier González-Castano. 2016. Gti at semeval-2016 task 5: Svm and crf for

aspect detection and unsupervised aspect-based sentiment analysis. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 306–311.

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

Adrien Bougouin, Florian Boudin, and Béatrice Daille. 2013. Topicrank: Graph-based topic ranking for keyphrase extraction. In *International joint conference on natural language processing (IJCNLP)*, pages 543–551.

Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Mário Jorge, Célia Nunes, and Adam Jatowt. 2018. Yake! collection-independent automatic keyword extractor. In *European Conference on Information Retrieval*, pages 806–810. Springer.

Bonnie Dorr David and David Zajic. 2003. Hedge trimmer: A parse-and-trim approach to headline generation. In *in Proceedings of Workshop on Automatic Summarization*. Citeseer.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. Eli5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567.

Aitor García-Pablos, Montse Cuadros, and German Rigau. 2018. W2vlda: almost unsupervised system for aspect based sentiment analysis. *Expert Systems with Applications*, 91:127–137.

Sujatha Das Gollapalli and Cornelia Caragea. 2014. Extracting keyphrases from research papers using citation networks. In *Twenty-eighth AAAI conference on artificial intelligence*.

Navita Goyal, Balaji Vasan Srinivasan, N Anandhavelu, and Abhilasha Sancheti. 2021. Multi-style transfer with discriminative feedback on disjoint corpus. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3500–3510.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2017. An unsupervised neural attention model for aspect extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 388–397.

Jaakko Hintikka and Esa Saarinen. 1979. Information-seeking dialogues: Some of their logical properties. *Studia Logica*, 38(4):355–363.

Tsutomu Hirao, Jun Suzuki, Hideki Isozaki, and Eisaku Maeda. 2003. Ntt's multiple document summarization system for duc2003. In *Proc. DUC*.

Tom Hosking and Sebastian Riedel. 2019. Evaluating rewards for question generation models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2278–2283.

Anette Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 216–223.

Wei Jin, Hung Hay Ho, and Rohini K Srihari. 2009. Opinionminer: a novel machine learning system for web opinion mining and extraction. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1195–1204.

Vishwajeet Kumar, Ganesh Ramakrishnan, and Yuan-Fang Li. 2019. Putting the horse before the cart: A generator-evaluator framework for question generation from text. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 812–821.

Huiyuan Lai, Antonio Toral Ruiz, and Malvina Nissim. 2021. Thank you bart! rewarding pre-trained models improves formality style transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*. Association for Computational Linguistics, ACL Anthology.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Fangtao Li, Chao Han, Minlie Huang, Xiaoyan Zhu, Yingju Xia, Shu Zhang, and Hao Yu. 2010. Structure-aware review mining and summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 653–661.

Feifan Liu, Deana Pennell, Fei Liu, and Yang Liu. 2009. Unsupervised approaches for automatic keyword extraction using meeting transcripts. In *Proceedings of human language technologies: The 2009 annual conference of the North American chapter of the association for computational linguistics*, pages 620–628.

Pengfei Liu, Shafiq Joty, and Helen Meng. 2015. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1433–1443.

Qian Liu, Bing Liu, Yuanlin Zhang, Doo Soon Kim, and Zhiqiang Gao. 2016. Improving opinion aspect extraction using semantic similarity and aspect associations. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Yixin Liu, Graham Neubig, and John Wieting. 2021. On learning text style transfer with direct rewards. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4262–4273.

Konstantin Lopyrev. 2015. Generating news headlines with recurrent neural networks. *arXiv preprint arXiv:1512.01712*.

Ling Luo, Xiang Ao, Yan Song, Jinyao Li, Xiaopeng Yang, Qing He, and Dong Yu. 2019. Unsupervised neural aspect extraction with sememes. In *IJCAI*, pages 5123–5129.

Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1643–1654.

Ramakanth Pasunuru and Mohit Bansal. 2018. Multi-reward reinforced summarization with saliency and entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 646–653.

Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2011. Opinion word expansion and target extraction through double propagation. *Computational linguistics*, 37(1):9–27.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389.

Marzieh Saeidi, Max Bartolo, Patrick Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. Interpretation of natural language rules in conversational machine reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2087–2097.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *ACL (1)*.

Jingbo Shang, Jialu Liu, Meng Jiang, Xiang Ren, Clare R Voss, and Jiawei Han. 2018. Automated phrase mining from massive text corpora. *IEEE Transactions on Knowledge and Data Engineering*, 30(10):1825–1837.

Tian Shi, Kyeongpil Kang, Jaegul Choo, and Chandan K Reddy. 2018. Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In *Proceedings of the 2018 World Wide Web Conference*, pages 1105–1114.

Tian Shi, Liuqing Li, Ping Wang, and Chandan K Reddy. 2021. A simple and effective self-supervised contrastive learning framework for aspect detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13815–13824.

Pushkar Shukla, Carlos Elmadjian, Richika Sharan, Vivek Kulkarni, Matthew Turk, and William Yang Wang. 2019. What should i ask? using conversationally informative rewards for goal-oriented visual dialog. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6442–6451.

Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. In *5th International Conference on Learning Representations*.

Manfred Stede and David Schlangen. 2004. Information-seeking chat: Dialogues driven by topic-structure. In *Proceedings of Catalog (the 8th workshop on the semantics and pragmatics of dialogue; SemDial04)*. Citeseer.

Antoine Tixier, Fragkiskos Malliaros, and Michalis Vazirgiannis. 2016. A graph degeneracy-based approach to keyword extraction. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1860–1870.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016. Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 616–626.

Yi-fang Brook Wu, Quanzhi Li, Razvan Stefan Bot, and Xin Chen. 2005. Domain-specific keyphrase extraction. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 283–284.

Yuxi Xie, Liangming Pan, Dongzhe Wang, Min-Yen Kan, and Yansong Feng. 2020. Exploring question-specific rewards for generating deep questions. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2534–2546.

Bishan Yang and Claire Cardie. 2012. Extracting opinion expressions with semi-markov conditional random fields. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1335–1345.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted

gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, and Xueqi Cheng. 2019. Outline generation: Understanding the inherent content structure of documents. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 745–754.

Shiyue Zhang and Mohit Bansal. 2019. Addressing semantic drift in question generation for semi-supervised question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2495–2509.