Recognising Summary Articles

Mark Fisher[1,2], Dyaa Albakour[2], Udo Kruschwitz[1], and Miguel Martinez[2]

[1] School of Computer Science and Electronic Engineering, University of Essex, UK
[2] Signal, 145 City Road, London, EC1V 1AZ, UK
fishbpm@googlemail.com, research@signal-ai.com, udo@essex.ac.uk

Abstract. Online content providers process massive streams of texts to supply topics and entities of interest to their customers. In this process, they face several information overload problems. Apart from identifying topically relevant articles, this includes identifying duplicates as well as filtering *summary* articles that comprise of disparate topical sections. Such summary articles would be treated as noise from a media monitoring perspective, an end user might however be interested in just those articles. In this paper, we introduce the recognition of summary articles as a novel task and present theoretical and experimental work towards addressing the problem. Rather than treating this as a single-step binary classification task, we propose a framework to tackle it as a two-step approach of *boundary detection* followed by classification. Boundary detection is achieved with a bi-directional LSTM sequence learner. Structural features are then extracted using the *boundaries* and *clusters* devised with the output of this LSTM. A range of classifiers are applied for ensuing summary recognition including a convolutional neural network (CNN) where we treat articles as 1-dimensional structural 'images'. A corpus of natural summary articles is collected for evaluation using the Signal 1M news dataset. To assess the *generalisation* properties of our framework, we also investigate its performance on synthetic summaries. We show that our structural features sustain their performance on generalisation in comparison to baseline bag-of-words and word2vec classifiers.

1 Introduction

As the news domain becomes increasingly digitalised, individuals and companies are now ever more reliant on monitoring tools to filter streams of online news articles. In particular, individuals seek to find news relevant to their interests, while companies proactively monitor online content to manage their own brand image, and position themselves to react quickly to changes in the industry and market [1].

Figure 1 depicts a typical processing pipeline for such monitoring tools. The key steps of this pipeline are the topic classification of articles, and the identification of relevant entities within each article. This pipeline must cope with massive streams of documents from various online sources, which can be noisy. Hence, the pre-processing step plays an important role in removing undesirable content (noise) that may affect the output of the latter steps.

One distinct example of such noise is what we define as 'summary' articles. Our definition is as follows: a summary article aggregates several otherwise disparate topical sections. If one writes a summary of another topical article (an article discussing one topic), the resulting article is still clearly topical. By contrast, in our definition, a

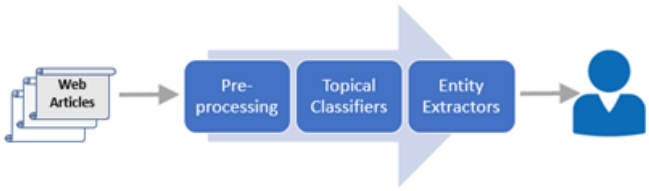

Fig. 1. A typical media monitoring pipeline.

summary article encompasses a collection of topics that do not bear any manifest relation. Such articles are often created by web aggregators[3], but are also published by other more traditional news sources, for example when reporting on today's current affairs. An example of a summary article is provided in Figure 2 (right). If these articles are passed over to the topical classifier of the pipeline in Figure 1, they might become classified under any of their constituent topics, rather than being discarded. Therefore, it is important to automatically identify these articles within a media monitoring context. In this paper, we introduce the new task of 'summary article recognition', which involves the binary classification of news articles into summaries or not (i.e. topical).

One may assume that summary articles will be presented in a characteristic form making them easy to recognise. Taking the two examples in Figure 2, the first (left) article is topical, because each paragraph discusses some aspect of one topic "Poland's financial markets". The article on the right consists of paragraphs which do not share any connective topic, and the article is hence a summary. In terms of their visual form, however, the two articles cannot be distinguished. Therefore, it is the underlying flow of topics and the entities, 'the linear structure' of an article, that is the principal determinant of its class, irrespective of its apparent visual form. Although the first article in Figure 2 also comprises of linearly segmented topics, these topics are each connected under the article's principal theme, thus rendering it topical.

Poland - Factors to Watch Sept 9

Following are news stories, press reports and events to watch that may affect Poland's financial markets on Wednesday. ALL TIMES GMT (Poland: GMT + 2 hours):

SWISS FRANC MORTGAGES
Poland's junior coalition partner the Polish Peasant Party (PSL), still wants banks to cover the bulk of costs of converting Swiss-franc mortgages into zlotys, a PSL parliamentarian said on Tuesday.

MIGRANTS
Polish Prime Minister Ewa Kopacz said the European Council which is to deal with Europe's migrant crisis is likely to take place earlier than planned.
Poland could accept more migrants than the 2,000 it declared earlier, but under certain conditions, Kopacz also said.

KGHM
The chief executive of Poland's KGHM, Europe's No.2 copper producer, said on Tuesday he expected copper prices to stabilise at $5,000 dollars per tonne.

10 Things to Know for Thursday

Your daily look at late-breaking news, upcoming events and the stories that will be talked about Thursday:

GOP OPPONENTS TARGET TRUMP
Rand Paul, calling The Donald too brash to lead, is among candidates ganging up on the front-runner at the second Republican debate.

CHAOS ERUPTS ON SERBIA-HUNGARY BORDER
Baton-wielding Hungarian police unleash tear gas and water cannons against hundreds of migrants trying to cross the border from Serbia.

WHY MUSLIM TEEN HAS BECOME SOCIAL MEDIA CAUSE
Sympathy spreads for 14-year-old Ahmed Mohamed after he was placed in handcuffs and suspended for taking a homemade clock to his Texas school that teachers thought resembled a bomb.

POWERFUL MAGNITUDE-8.3 EARTHQUAKE HITS CHILE
The quake causes buildings to sway in Santiago and other cities and sends people running into the streets.

Fig. 2. Examples of topical (left) and summary (right) articles.

There exists a range of established tools that can help in modelling the article's content in terms of topics and entities. These include generative methods with topic modelling as proposed by [2, 3], as well as entity-based approaches, the most effec-

[3] examples are `https://news360.com` and `https://www.bloomberg.com/series/top-headlines`

tive of which leverage a knowledge graph [4]. Although powerful, these methods have certain limitations. They either rely on corpus specific parameterisation that impairs generalisation, or require elaborate processing that limits extension to large datasets or document streams.

To this end, in this paper, we propose a framework for summary recognition that does not suffer from the aforementioned limitations. Our framework consists of two steps: structure extraction followed by classification. For structure extraction, we employ *boundary detection* to characterise the 'linear structure' of an article. In particular, boundary detection quantifies whether there is a topic shift at the end of every sentence [4] in the article. The output of boundary detection is then used to devise 'structural' features used for the classification step in a supervised manner. To illustrate the intuition behind our framework, in Figure 3 we visualise the output of boundary detection for each sentence in both topical and summary articles. One can visually discriminate between summary and topical articles, as summary articles exhibit blunt topic shifts in the text. It is therefore reasonable to rely solely on structural features devised from the boundary detection step to classify summary articles.

For boundary detection, we use neural networks and word embeddings for a supervised strategy. Building on the work of Koshorek *et al*. [5], we perform boundary detection with a pre-trained LSTM (Long Short-Term Memory) neural network. Unlike [5] who train on Wikipedia content headers for detection of 'narrative' boundaries, the distinctive aspect of our approach is a synthetic training set of 1 Million summary articles which we tailor to detect 'topical' boundaries.

As summary recognition has not, as far as we are aware, been previously attempted, we trial a variety of structural features. This enables us to comparatively assess their generalisation performance and identify potential candidates for future research. Our proposed features are as follows: (i) *boundary* features directly derived from the output of the boundary detector (example in Figure 3), (ii) *cluster* features derived by applying a linear clustering step on top of the boundary detection output. Finally, for the *classification* component of our framework, we evaluate a number of binary classification models, but foremost we propose boundaries be treated as 'structural images'; this enables us to capture the overall aggregate structure of an article regardless of its length, and to leverage the power of a convolutional neural network (CNN).

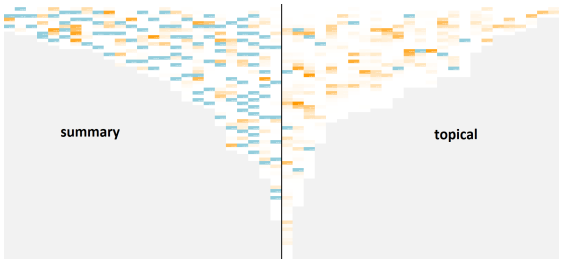

Fig. 3. Each column represents an article and rows the output of boundary detection applied subsequently on each sentence. Darker colours represents higher probability of boundaries.

[4] Paragraph delimiters are not consistently available, especially in the realm of digital web content. For robustness we thus perform all boundary detection at the sentence level.

To evaluate our framework, we build a *natural* dataset of summary and topical articles by annotating a sample of the Signal 1M news dataset [6]. To further gauge generalisation performance, we also construct an additional *composite* training set comprising of synthetic summaries. Our contributions can be summarised as follows:

- We introduce the new task of summary recognition and devise a dataset to foster further research on this task. This dataset is built on top of the public Signal 1M dataset and we make it publicly available [5].
- Using this dataset, we evaluate our framework with a number of structural features for summary recognition. The results show that it sustains its performance on generalisation in comparison to baseline bag-of-words and word2vec classifiers.

The remainder of the paper is structured as follows. We give a brief review of related work that underpins our functional components (Section 2), before presenting our framework thoroughly (Section 3). In Section 4, we present the three datasets we employ for experimentation. These experiments and their results are then detailed in Sections 5 and 6, followed by our conclusions (Section 7).

2 Related Work

Although our goal is recognising summary articles, the principal facet of our framework is boundary detection. Boundary detection is most prominently employed in the field of text segmentation; also known equivalently as 'linear clustering'. This involves the detection of boundaries within a text that together form an optimally cohesive sequence of contiguous segments.

2.1 Text Segmentation

Text segmentation is inherently an unsupervised problem as there are rarely true objective boundaries. Hence, supervised methods are usually domain specific relying on supplementary sources to mark out these boundaries. One area where such 'multi-source' approaches have proven effective is the transcript and newswire domains [7, 8], where content breaks are more explicit, aligned with natural prosodic features such as pauses, speaker change and cue phrases.

Unusually, Koshorek *et al*. [5] overcome this by leveraging Wikipedia headers to label sentence boundaries in Wikipedia articles, producing a somewhat 'narrative' segmentation. We take a similar approach, but instead synthesise our training data to produce a more blunt 'topical' segmentation suited to our summary recognition task.

Aside from these few supervised methods, the bulk of segmentation research is in the *un*supervised field where a wide variety of algorithms [9–11] have arisen. This includes statistical and hierarchical methods that involve dynamic programming [12] and more elaborate probabilistic modelling [13] to infer optimal clustering of a text. These generative methods all require parameterisation, for example in Misra *et al*.'s LDA method [3] the number of topics and Dirichlet priors must first be specified for the sample corpus. In this paper, we propose a framework that is generic and extensible, so that our model can be deployed dynamically to new document streams.

[5] `https://research.signal-ai.com/datasets/signal1m-summaries.html`

2.2 Neural Methods

The rise of neural networks has provided new mechanisms for representing words and sentences, which as demonstrated by [5] can also be employed for our boundary detection component. Although we opt to employ pre-trained word2vec embeddings, as first introduced by [14], more elaborate pre-trained embeddings are also available. This includes CPHRASE [15], which use syntactic relations to selectively determine which context words are used for training. Garten *et al*. [16] also show that combining multiple embeddings, via aggregation or maxima, further improves performance. There is thus plenty of scope for trialling different trained embeddings.

Sentence embeddings can also be trained using similar methods. Implementations such as FastSent [17] and Skip Thought [18] have adopted equivalent (Continuous Bag of Words) CBOW and Skip Gram training mechanisms, but employing contextual sentences rather than words. More recently, Sent2Vec [19] augmented the CBOW approach by including n-grams within the context window. Hill *et al.* [17] observed that simple aggregation of word and n-gram embeddings such as neural bag-of-words can still achieve equivalent performance to the aforementioned sentence embedding approaches on unsupervised tasks.

As shown by [5], LSTM neural networks can also be used in an equivalent unsupervised capacity to encode sentences. Here, aggregation is performed in alignment with the LSTM's bi-directional context window to capture a more sequential embedding. These recurrent LSTMs, as first proposed by Hochreiter and Schmidhuber [20], are well known for their performance in sequence learning such as machine translation (MT). Sutskever *et al*. [21] outperformed a statistical MT system, despite their LSTM being restricted to a more limited vocabulary. Moreover, Chenglin *et al*. [22] developed an elaborate bi-directional LSTM architecture incorporating Viterbi decoding to model prosodic and lexical features for sentence boundary detection in broadcast news. But as far as we know from our research, Koshorek *et al.*'s [5] is the first attempt at text segmentation using neural methods. We employ their model directly, but extract the softmax boundary layer for input into our feature-based approaches.

3 Framework for Structural Summary Recognition

We propose a framework comprising two generic functional components: the structure extractor, followed by a binary classifier (Figure 4). The structure extractor aims to characterise the linear structure of the article. It outputs structural features which are then used by a binary classifier for summary recognition. The output of the framework is a binary label for the article: positive (summary) or negative (topical) [6].

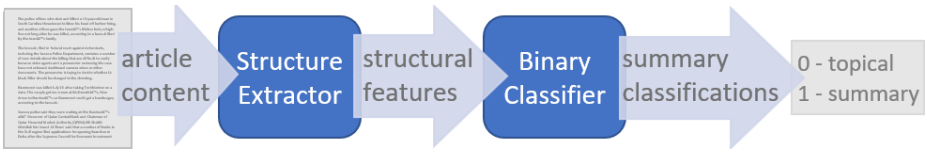

Fig. 4. Framework for Structural Summary Recognition.

[6] We refer to negative articles by our classification as 'topical', as the vast majority of non-summary articles are typically topical

For the **structure extractor** component of the framework, we propose to employ a boundary detection approach as shown in Figure 5. The boundary detector produces a probability for each sentence in the article, denoting the likelihood of a topic shift in the following sentence. These probabilities are used to engineer structural features. In our implementation of the framework, we propose two families of structural features; sentence *boundaries* and word *clusters* (Figure 5).

For the **binary classifier** component of the framework, we trial a number of models as suited to each class of structural features. As choice of classifier is intrinsically tied to the features, we present these classifier components alongside the respective features in Sections 3.2 and 3.3. Before this, we present our boundary detection approach in Section 3.1.

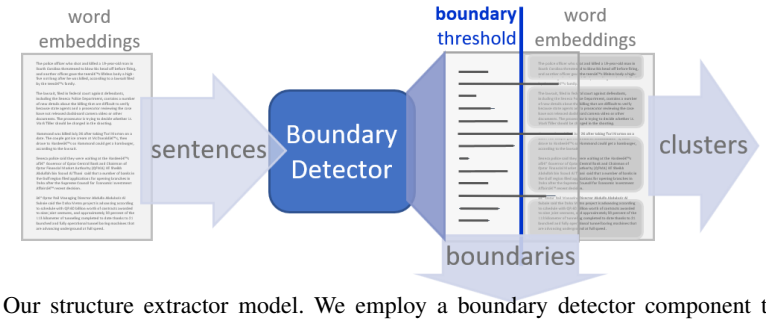

Fig. 5. Our structure extractor model. We employ a boundary detector component to extract *boundary* and *cluster* features. Sections 3.2 and 3.3 describe how these features are extracted.

3.1 Boundary Detector

For boundary detection, we employ the LSTM model proposed by Koshorek *et al*. [5]. This model has a dual architecture commencing with a first a *sentence encoder* followed by a *sequence labeller*. The sentence encoder is an *un*supervised LSTM network. It acts as an aggregator, encoding the set of word embeddings within each sentence. Rather than a flat aggregation of words, it aggregates the word sequences in the sentence, making it well suited to the boundary detection objective. The encoded sentences are then supplied to the sequence labeller, which is a supervised bi-directional LSTM trained to label a sequence of sentences as boundaries or not. The final softmax output layer of the network thus provides a boundary probability for each sentence in the article, that are used in constructing the boundary features in Section 3.2.

It should be noted that there are other options to implement the boundary detector component of our framework. This includes, for example, state-of-the-art unsupervised models such as GraphSeg [23]. We leave this for future work.

3.2 Boundary Features

Here, we use the boundary probabilities directly to characterise an article's linear structure. We extract these structural features in two forms.

In the first form, we apply the full sequence of boundary probabilities to feed a CNN classifier. We refer to this set as an '*image*', as it captures the complete sequential structure of the article. Just as a CNN convolves over 2-dimensional visual images, here it convolves over our 1-dimensional structural image. This would enable it to recognise any intrinsic elements or artefacts that might typify the style of summary articles, whilst

maintaining invariance to the specific position of these elements. We start with the foundation architecture optimised for image recognition by Lecun *et al*. [24], but make its convolutional layers 1-dimensional. The number of filters and dense layers are also adjusted through general experimentation to maximise performance. Drop out layers in particular were found to be benefical, as shown in Figure 6.

Fig. 6. Binary classification of summary articles using a CNN with image-based boundaries.

In the second form, we 'average' the boundary probabilities. Here, as an additional feature, we also include the quantity of detected boundaries by applying a threshold to the boundary probabilities as visualised in Figure 5. These two features provide a more generic representation of the article, which we hypothesise may improve its scope in summary classification. We trial the SVM model for binary classification of summary articles using these features.

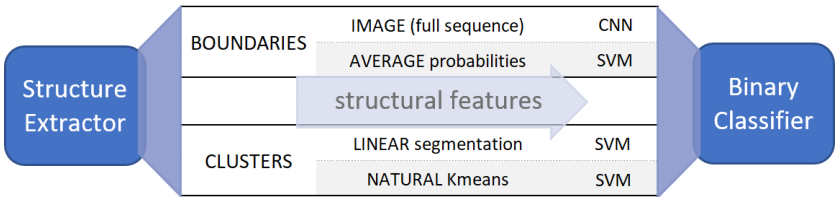

Fig. 7. Our two structural features. For cluster features we employ Silhouette [25] and Calinski Harabaz scores [26]. Apart from the image features, all features additionally include the normalised quantity of detected boundaries.

3.3 Clustering Features

Here, we aim to capture richer structural features of the articles. The hypothesis is that summary articles will consist of distant clusters, while topical ones will have close and cohesive clusters. We employ two forms of clustering; 'linear' and 'natural'.

Linear clusters are formed by segmenting the text in compliance with the boundary threshold as visualised in Figure 5. We also relax this linear constraint to perform *natural* K-Means clustering of the article's word embeddings. Here, to encourage a clustering that may still correlate with the article's potential topics, the *quantity* of detected boundaries is used to seed the K-Means algorithm. Therefore, these clusters are still indirectly dependent upon the boundary detection. While linear clusters are fully supervised by the boundary detector, natural clusters are semi-supervised by virtue of this dependency. Relaxing the linear constraint should allow K-Means to yield more cohesive clusters, but it is unclear whether these will still sufficiently correlate with the contiguous structure of the article, hence why we opt to trial both forms of clustering.

After clustering, we compute clustering metrics using Silhouette [25] and Calinski Harabaz scores [26], which characterise the cohesion and distribution of the resulting clusters. The clustering features comprise of both these metrics and the quantity of clusters (normalised by number of sentences). These features are then used for binary summary classification where, as before, we trial the SVM classifier.

4 Datasets

Three datasets are assembled for training and evaluation purposes. These are sampled from two distinct data sources as shown in Table 1 which outlines the composition of each dataset. We devise a *natural* dataset for evaluating summary classification performance. We make this dataset available for public use. [7] We also construct a much larger set of *synthetic* summaries for training of the LSTM boundary detector (described in Section 3.1). Additional summaries are then synthesised, alongside a randomised selection of topical articles to build a *composite* dataset to evaluate the generalisation performance of our summary classifiers. Next, we describe our datasets in more detail.

Table 1. The sizes and the sources of our three datasets. †synthesised from the 'Topical News Articles' source, as set out in Section 4.2

		Boundary training	*Summary classification*	
		Synthetic	**Natural**	**Composite**
Data Source	Size	1,000,000	892	892
Signal 1M	1,000,000		892	
Topical News Articles	31,000			446
†Aggregated (2-10 topics)	1,000,000	1,000,000		446

4.1 Natural Summaries Dataset

To evaluate summary article recognition, we collect summary articles using the Signal 1M news article dataset [6]. This dataset covers a typical stream of news articles processed for media monitoring purposes. It includes 1 million articles during September 2015 from 93K sources ranging from web aggregators to premium publications.

To label articles from Signal 1M, we obtained a biased sample of 2900 articles. To create this sample, we use a Lucene index and apply search terms such as 'month', 'week', 'review', 'report' and 'roundup', using vector-based tf-idf ranking to retrieve the highest scoring articles. In conducting this search, certain sources were found to produce a larger proportion of summary-style articles. As summaries are relatively uncommon, queries were also further tailored to promote articles of these sources. This biased sample (rather than a random one) somewhat limits the variance of the resulting dataset, but was necessary in order to obtain a sufficient quantity of summary articles.

From the biased sample (2900 articles), a subset of 400 articles was first labelled by 4 independent annotators, in order to gauge labelling accuracy, yielding a pair-wise agreement of 85%. This reflects the ambiguity inherent in the recognition of summary articles, as demonstrated by the examples in Figure 2. Due to resource limitations, the remaining 2500 articles were then labelled by one of our 4 annotators. Using the labels of this annotator, the biased sample had 446 summary articles and 2454 topical articles. To create a *balanced* natural dataset, we selected all the 446 summary articles and an equal quantity of topical articles drawn randomly, providing a total of 892 articles. Although surplus articles remain, balancing is very important for effective training of our binary classifiers.

[7] `https://research.signal-ai.com/datasets/signal1m-summaries.html`

4.2 Synthetic Summaries Dataset

Effective sequence training of an LSTM network requires a very large volume of labelled articles. As a large labelled dataset of summary articles is not readily available, we opt to instead *synthesise* our boundary training set, using topical articles.

To construct this training set, we first obtained a private set of 31,000 topical news articles, from similar sources of those covered by Signal 1M, but during a larger time frame (September 2015 till July 2018). Each article is manually labelled by independent commercial annotators with one of 50 different topical classes. To synthesise summary articles from these topical articles, we follow the protocols employed by Choi on his 'Choi dataset' [10], which has become recognised as the reference baseline for segmentation evaluation. In particular, these protocols are designed to mirror the variability of natural articles. With this protocol, to synthesise a summary article consisting of subsequent segments, a distinct topical article is selected (from the 31,000 topical articles) for each segment, ensuring no topic repetition. Then, a random position within this topical article is selected to extract the requisite quantity of contiguous sentences. Applying these protocols we synthesise 1 Million articles forming our synthetic summary dataset.

4.3 Composite Summaries Dataset

In order to assess the 'generalisation' capabilities of our framework, we assemble an additional *composite* dataset of summary articles. This set is for training purposes only, to evaluate its impact on natural summary classification.

We call this set 'composite' as its summary portion is synthetic (synthesised using the protocol described in Section 4.2), while its topical portion is natural (see Table 1). The basis for its use in generalisation is 2-fold, upon both content and structure:

(i) *Content*: The entire dataset is sourced from a much wider variety of articles, encompassing three years (Sept-2015 through July-2018), unlike our natural dataset which is sampled from the more restrictive 1 month (Sept-2015) Signal 1M dataset.

(ii) *Structure*: Having been algorithmically synthesised, its summary samples are each structurally much more uniform; They do not exhibit the same stochastic variation as natural occurring summaries.

To aid direct evaluation against our natural corpus, we also make this composite set the same size (892 articles), again balancing summary and topical. Topical articles are drawn *randomly* from the 31,000 dataset to maximise its variance.

5 Experiments

Our experiments seek to evaluate the main objectives we set out in Section 1. In particular, our experiments aim to answer the following research questions:

RQ1: How does our framework compare to existing content-feature and semantic-feature approaches, such as, bag-of-words or pairwise sentence similarity?

RQ2: Can our framework generalise effectively to new settings and content shift?

RQ3: Within our framework, which structural features are most effective for summary article recognition?

Next, we describe the baselines we use. Then, we detail our experimental setup and implementation details.

5.1 Baselines

As far as we are aware, the task of summary article recognition has not yet been attempted. Therefore, there is no obvious state-of-the-art baseline to compare our framework. We therefore experimented with three supervised baselines, encompassing both conventional and word embedding approaches. For all the baselines, we use standard binary classification applied on different sets of features. As an initial content-based baseline, we apply conventional **bag-of-word** features. We then leverage word embeddings for two further semantic baselines. First, we use the *relatedness score* [23] for every adjacent pair of sentences to calculate the **average pairwise sentence similarity** of each article. This exploits the same semantic relatedness properties as used to train our LSTM, but in an unsupervised setting, so is more directly comparable to our structural framework. Finally, we apply **aggregated embeddings** [27] where for each article, a single aggregated vector is produced by averaging the word2vec embeddings for each unique word in the article.

5.2 Experimental Setup

The experiments assess the performance of the baselines and our framework using all the combination of our structural features and summary classifier described in Section 3 and summarised in Figure 7.

We ran two distinct experiments. In the first experiment, *natural training*, we use our balanced natural dataset of 892 articles (Section 4.1). We employ 5-fold stratified cross validation (CV), reporting average independent performance on each fold. In the second experiment, *composite training*, we aim to evaluate the generalisation properties of our models. Here, we train separately on our identical size composite dataset (see Section 4.3). The identical 5-fold CV strategy is applied but we test on the corresponding natural dataset folds, as previously stratified, allowing direct comparison with natural training.

5.3 Implementation Details

Pre-Processing: As sentences are the base unit for boundary detection, pre-processing is aimed at yielding a coherent and contiguous set of candidate sentences, each of adequate length; Length is important to avoid data sparsity issues that might arise due to a lack of matched word2vec embeddings. First, employing a SpaCy syntactic parser, we clean all non-content words, numerics and punctuation. Thereafter, pre-processing involved discarding small paragraphs[2], which typically constitute headers (we opt for a 50-character limit on paragraph retention), then concatenating short sentences

Word Embeddigns: For all our approaches, features are built upon foundation word embeddings. As we source our articles from the Signal 1M collection [6], there is no particular relevant domain that would offer the potential to train tailored embeddings. We therefore employ word2vec pre-trained Google News embeddings [14], which are also well suited to the general news domain of our corpus. To maximise semantic interpretation, we allow Google News to enforce its own limit on stop words.

Neural Networks: Our LSTM two-layer network for boundary detection was trained using our synthetic datasets (See Section 4.2). It was trained in 40 hours using an Nvidia Tesla GPU. When performing CV, the CNN network weights are re-initialised on each fold to ensure a new model is fitted.

Feature Normalisation: As articles vary in size, structural features must also be normalised to enable effective use in classification. For most of our approaches this is achieved in straightforward fashion, normalising by the total quantity of clusters or boundaries as respective to the class of feature. For our *image* features, we perform normalisation using a bicubic image filter. Here, to best preserve the structural representation of the article we opt for a target size of 40 boundaries, which approximates the average size in our natural test dataset.

6 Results

The cross validation results for our framework (using boundaries and clustering features) and the baselines in both experiments are reported in Table 2.

In the *natural* training experiment, our bag-of-words and aggregated embeddings baselines show strongest classification performance in terms of both precision and recall (and thus F1 accuracy), exceeding our best performing linear segmentation features (0.8067 vs. 0.6834). This points to the degree of content consistency in the natural dataset, likely contributed by some publishers we have selected in our biased sampling of Signal 1M (see Section 4.1) having a consistent style.

In the *composite* training experiment, the performance of all of the baselines drop, up by 12 percentage points for aggregated embeddings, from 0.8067 to 0.6801. In other words, their performance is not resilient to content shift. By contrast, most variants of our framework maintain a similar classification performance when tested on a different setting. They exhibit a marginal drop in F1 when comparing their performance between natural training and composite training. With composite training, one variant of our framework 'linear segmentation/SVM' is significantly better than both the bag-of-words and pairwise sentence similarity baselines using McNemar Test ($p < 0.01$).

To summarise, as an answer to our research question RQ1, we can conclude that conventional content-features approaches may be adequate for summary recognition, and they outperform our framework. The aggregated embeddings in particular show strongest performance. This is only true, however, when re-training a model is feasible, and a budget is available to collect labelled data. For RQ2, we conclude our framework has a strong generalisation performance. It has shows to be resilient to content shift (the composite training). This suggests that it has the potential to sustain its performance if deployed on dynamic content streams that continuously change.

Table 2. Average CV classification performance for summary articles. All test results encompass the full 892 articles of our *natural* summaries dataset. For baselines, we report the best performing classifier (with composite training) from logistic regression, Naïve-Bayes (NB) and SVM. ∘ and † denote statistically significant differences of classification decisions when compared to bag-of-words and average pairwise sentence similarity respectively using McNemar Test ($p < 0.01$).

			Natural Training			*Composite Training*		
	Features	Class.	P	R	F1	P	R	F1
Boundaries	image	CNN	0.7382	0.6143	0.6703	0.6621	0.5492	0.6001
	average probabilities	SVM	0.6459	0.6728	0.6585	0.6107	0.6032	0.6416
Clusters	linear segmentation	SVM	0.6630	0.7062	0.6834	0.6581	0.7084	**0.6820**∘†
	natural K-Means	SVM	0.6628	0.6524	0.6573	**0.7210**	0.5828	0.6425
Baselines	aggregated embeddings	log. reg.	**0.7647**	0.8542	**0.8067**	0.6152	0.7622	0.6801
	avg. pairwise sent. sim.	SVM	0.5853	0.7534	0.6588	0.5839	0.7265	0.6469
	bag-of-words	NB	0.6429	**0.9685**	0.7728	0.5023	**0.9955**	0.6677

Next, we analyse the differences between our proposed combinations of *structural features* and binary classifiers within the framework. Table 2 shows that the 'image/CNN' achieves the strongest performance on *natural* training. On generalisation, however, this performance drops noticeably, unlike 'average probabilities' which sustains its performance. This suggests that the CNN is better equipped to learn the distinctive aspects of summaries in the training domain. From this viewpoint, the CNN's performance on natural training is perhaps still muted. We suggest this is due to the small size of our dataset; with more training samples, performance of this neural method can reasonably be expected to improve. The 'average probabilities/SVM' achieves a lower F1 than 'image/CNN', but appear to more resilient; with more balance between precision and recall. Also, it is more capable of sustaining performance on generalisation (dropping F1 only marginally 0.6585 to 0.6416), we thus suggest these averaging boundary probabilities provides a better foundation for improvement.

Our clustering feature variants 'linear segmentation/SVM' and 'natural K-Means/SVM' show the strongest overall performance on 'composite training' within our framework. In practical use, however, these clustering features may not be the optimal choice. As a gauge, our trained LSTM generates boundary predictions at the rate of 4 articles per second on an Nvidia Tesla GPU; thereafter, averaging these features is trivial. On the other hand, clustering costs additional CPU time for cluster assembly.

In summary, and as an answer to RQ3, we conclude that the clustering features have strong generalisation capabilities, and are more resilient to content changes than sophisticated CNN approaches trained on sequence of boundary probabilities. The caveat however is their computational complexity.

7 Conclusion

We present the new task of 'summary recognition', that is relevant in a media monitoring context in particular but is also applicable in many other scenarios. To address this task, we propose a structural framework for summary article prediction aimed principally at achieving generalised performance, which is resilient to variations and shifts in content. The salient component of the framework is a structure extractor that identifies the linear (or semantic) structure of the article to aid summary classification. Building on the work of boundary detection and text segmentation, we show that we can effectively devise structural features that are robust for summary recognition. In particular, we show that our structural features sustain their performance upon generalisation to new content distributions, compared to established aggregated embeddings and bag-of-words baselines that both markedly degrade in performance.

Central to our experiments in this paper, is the construction of new datasets (natural and synthetic) to evaluate the effectiveness of our framework and its generalisation behaviour. The natural is made public to foster further research in this area.

As we are the first to experiment with summary recognition, an important aspect of our work is to provide a foundation for further research. Based on performance of our boundary structural features, for future work, we suggest methods for tailoring the word embeddings to produce more purpose-specific boundaries that may enhance their performance. As entities are a principal topical indicator, we suggest incorporating knowledge graph concept vectors [28] or training embeddings on their entity usage contexts only. Following the findings of Garten *et al*. [16], such approaches may also be combined if beneficial to augment pre-trained generalised embeddings.

References

1. Martinez, M., Kruschwitz, U., Kazai, G., Hopfgartner, F., Corney, D., Campos, R., Albakour, D.: Report on the 1st International Workshop on Recent Trends in News Information Retrieval (NewsIR'16). SIGIR Forum **50**(1) (2016) 58–67
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3** (2003) 993–1022
3. Misra, H., Yvon, F., Jose, J.M., Cappe, O.: Text segmentation via topic modeling: An analytical study. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management. CIKM '09, New York, NY, USA, ACM (2009) 1553–1556
4. Schuhmacher, M., Ponzetto, S.P.: Knowledge-based graph document modeling. In: Proceedings of the 7th ACM International Conference on Web Search and Data Mining (WSDM). (2014) 543–552
5. Koshorek, O., Cohen, A., Mor, N., Rotman, M., Berant, J.: Text segmentation as a supervised learning task. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), Association for Computational Linguistics (2018) 469–473
6. Corney, D., Albakour, D., Martinez-Alvarez, M., Moussa, S.: What do a million news articles look like? In: Proceedings of the First International Workshop on Recent Trends in News Information Retrieval co-located with 38th European Conference on Information Retrieval (ECIR 2016), Padua, Italy, March 20, 2016. (2016) 42–47
7. Pillai, R.R., Idicula, S.M.: Linear text segmentation using classification techniques. In: Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India. A2CWiC '10, New York, NY, USA, ACM (2010) 58:1–58:4
8. Galley, M., McKeown, K., Fosler-Lussier, E., Jing, H.: Discourse segmentation of multi-party conversation. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1. ACL '03, Stroudsburg, PA, USA, Association for Computational Linguistics (2003) 562–569
9. Hearst, M.A.: TextTiling: Segmenting Text into Multi-paragraph Subtopic Passages. Computational Lingustics **23**(1) (1997)
10. Choi, F.Y.Y.: Advances in domain independent linear text segmentation. In: Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference. NAACL 2000, Stroudsburg, PA, USA, Association for Computational Linguistics (2000) 26–33
11. Dadachev, B., Balinsky, A., Balinsky, H.: On automatic text segmentation. In: Proceedings of the 2014 ACM Symposium on Document Engineering. DocEng '14, New York, NY, USA, ACM (2014) 73–80
12. Utiyama, M., Isahara, H.: A statistical model for domain-independent text segmentation. In: Proceedings of the 39th Annual Meeting on Association for Computational Linguistics. ACL '01, Stroudsburg, PA, USA, Association for Computational Linguistics (2001) 499–506
13. Riedl, M., Biemann, C.: TopicTiling: A Text Segmentation Algorithm based on LDA. In: Proceedings of ACL 2012 Student Research Workshop, Association for Computational Linguistics (2012) 37–42
14. Mikolov, T., Chen, K., Corrado, G.S., Dean, J.: Efficient estimation of word representations in vector space. CoRR **abs/1301.3781** (2013)
15. Pham, N.T., Kruszewski, G., Lazaridou, A., Baroni, M.: Jointly optimizing word representations for lexical and sentential tasks with the c-phrase model. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics (2015) 971–981

16. Garten, J., Sagae, K., Ustun, V., Dehghani, M.: Combining distributed vector representations for words. In: Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, Association for Computational Linguistics (2015) 95–101
17. Hill, F., Cho, K., Korhonen, A.: Learning distributed representations of sentences from unlabelled data. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics (2016) 1367–1377
18. Kiros, R., Zhu, Y., Salakhutdinov, R.R., Zemel, R., Urtasun, R., Torralba, A., Fidler, S.: Skip-thought vectors. In Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R., eds.: Advances in Neural Information Processing Systems 28. Curran Associates, Inc. (2015) 3294–3302
19. Pagliardini, M., Gupta, P., Jaggi, M.: Unsupervised learning of sentence embeddings using compositional n-gram features. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics (2018) 528–540
20. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8) (1997) 1735–1780
21. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q., eds.: Advances in Neural Information Processing Systems 27. Curran Associates, Inc. (2014) 3104–3112
22. Xu, C., Xie, L., Xiao, X.: A bidirectional lstm approach with word embeddings for sentence boundary detection. J. Signal Process. Syst. **90**(7) (2018) 1063–1075
23. Glavaš, G., Nanni, F., Ponzetto, S.P.: Unsupervised text segmentation using semantic relatedness graphs. In: Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics, Association for Computational Linguistics (2016) 125–130
24. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradientbased learning applied to document recognition. Proceedings of the IEEE **86**(11) (1998) 2278–2324
25. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. (In: Computational and Applied Mathematics 20: 53-65) 53–65
26. Calinksi, T., Harabasz, J.: A dendrite method for cluster analysis. (In: Communications in Statistics, Volume 3:1, 1974) 1–27
27. Balikas, G., Amini, M.R.: An empirical study on large scale text classification with skip-gram embeddings. arXiv preprint arXiv:1606.06623 (2016)
28. Schuhmacher, M., Ponzetto, S.P.: Exploiting dbpedia for web search results clustering. In: Proceedings of the 2013 workshop on Automated knowledge base construction, AKBC@CIKM 13, San Francisco, California, USA, October 27-28, 2013. (2013) 91–96