KPTimes: A Large-Scale Dataset for Keyphrase Generation on News Documents

Abstract

Keyphrase generation is the task of predicting a set of lexical units that conveys the main content of a source text. Existing datasets for keyphrase generation are only readily available for the scholarly domain and include non-expert annotations. In this paper we present KPTimes, a large-scale dataset of news texts paired with editor-curated keyphrases. Exploring the dataset, we show how editors tag documents , and how their annotations differ from those found in existing datasets. We also train and evaluate state-of-the-art neural keyphrase generation models on KPTimes to gain insights on how well they perform on the news domain. The dataset is available online at https:// github.com/ygorg/KPTimes.

Introduction

Keyphrases are single or multi-word lexical units that best summarise a document BIBREF0. As such, they are of great importance for indexing, categorising and browsing digital libraries BIBREF1. Yet, very few documents have keyphrases assigned, thus raising the need for automatic keyphrase generation systems. This task falls under the task of automatic keyphrase extraction which can also be the subtask of finding keyphrases that only appear in the input document. Generating keyphrases can be seen as a particular instantiation of text summarization, where the goal is not to produce a well-formed piece of text, but a coherent set of phrases that convey the most salient information. Those phrases may or may not appear in the document, the latter requiring some form of abstraction to be generated. State-of-the-art systems for this task rely on recurrent neural networks BIBREF2, BIBREF3, BIBREF4, and hence require large amounts of annotated training data to achieve good performance. As gold annotated data is expensive and difficult to obtain BIBREF5, previous works focused on readily available scientific abstracts and used author-assigned keyphrases as a proxy for expert annotations. However, this poses two major

issues: 1) neural models for keyphrase generation do not generalize well across domains, thus limiting their use in practice; 2) author-assigned keyphrases exhibit strong consistency issues that negatively impacts the model's performance. There is therefore a great need for annotated data from different sources, that is both sufficiently large to support the training of neural-based models and that comprises gold-standard labels provided by experts. In this study, we address this need by providing KPTimes, a dataset made of 279 923 news articles that comes with editor-assigned keyphrases.

Online news are particularly relevant to keyphrase generation since they are a natural fit for faceted navigation BIBREF6 or topic detection and tracking BIBREF7. Also, and not less importantly, they are available in large quantities and are sometimes accompanied by metadata containing human-assigned keyphrases initially intended for search engines. Here, we divert these annotations from their primary purpose, and use them as gold-standard labels to automatically build our dataset. More precisely, we collect data by crawling selected news websites and use heuristics to draw texts paired with gold keyphrases. We then explore the resulting dataset to better understand how editors tag documents, and how these expert annotations differ from author-assigned keyphrases found in scholarly documents. Finally, we analyse the performance of state-of-the-art keyphrase generation models and investigate their transferability to the news domain and the impact of domain shift.

Existing datasets

Frequently used datasets for keyphrase generation have a common characteristic that they are, by and large, made from scholarly documents (abstracts or full texts) paired with non-expert (mostly from authors) annotations. Notable examples of such datasets are SemEval-2010 BIBREF8 and KP20k BIBREF2, which respectively comprises scientific articles and paper abstracts, both about computer science and information technology. Detailed statistics are listed in Table . Only two publicly available datasets, that we are aware of, contain news documents: DUC-2001 BIBREF9 and KPCrowd BIBREF10.

Originally created for the DUC evaluation campaign on text summarization BIBREF11, the former is composed of 308 news annotated by graduate students. The latter includes 500 news annotated by crowdsourcing. Both datasets are very small and contain newswire articles from various online sources labelled by non-expert annotators, in this case readers, which is not without issues.

Thus, unlike author annotations, those produced by readers exhibit significantly lower missing keyphrases, that is, gold keyphrases that do not occur in the content of the document. In the DUC-2001 dataset for example, more than 96% of the gold keyphrases actually appear in the documents. This confirms previous observations that readers tend to assign keyphrases in an extractive fashion BIBREF12, which makes these datasets less suitable for the task at hand (keyphrase generation) but rather relevant for a purely extractive task (keyphrase extraction). Yet, author-assigned keyphrases commonly found in scientific paper datasets are not perfect either, as they are less constrained BIBREF13 and include seldom-used variants or misspellings that negatively impact performance. One can see there is an apparent lack of sizeable expert-annotated data that enables the development of neural keyphrase generation models in a domain other than scholarly texts. Here, we fill this gap and propose a large-scale dataset that includes news texts paired with manually curated gold standard annotations.

Building the KPTimes dataset

To create the KPTimes dataset, we collected over half a million newswire articles by crawling selected online news websites. We applied heuristics to identify the content (title, headline and body) of each article and regarded the keyphrases provided in the HTML metadata as the gold standard. A cherry-picked sample document is showcased in Figure , it allows to show present and absent keyphrases, as well as keyphrase variants (in this example News media and journalism).

We use the New York Times as our primary source of data, since the content tagging policy that it applies is rigorous and well-documented. The news articles are annotated in a semi-automatic way, first the editors revise a set of tags proposed by an algorithm. They then provide additional tags which will be used by a taxonomy team to improve the algorithm.

We first retrieved the URLs of the free-to-read articles from 2006 to 2017, and collected the corresponding archived HTML pages using the Internet Archive. Doing so allows the distribution of our dataset using a thin, URL-only list. We then extracted the HTML body content using beautifulsoup and devised heuristics to extract the main content and title of each article while excluding extraneous HTML markup and inline ads. Gold standard keyphrases are obtained from the metadata (field types news_keywords and keywords) available in the HTML page of each article. Surface form variants of gold keyphrases (e.g. “AIDS; HIV”, “Driverless Cars; Self-Driving Cars” or “Fatalities; Casualties”), which are sometimes present in the metadata, are kept to be used for evaluation purposes.

We further cleansed and filtered the dataset by removing duplicates, articles without content and those with too few (less than 2) or too many (more than 10) keyphrases. This process resulted in a set of 279 923 article-keyphrase pairs. We randomly divided this dataset into training (92.8%), development (3.6%) and test (3.6%) splits.

Restricting ourselves to one source of data ensures the uniformity and consistency of annotation that is missing in the other datasets, but it may also make the trained model source-dependent and harm generalization. To monitor the model's ability to generalize, we gather a secondary source of data. We collected HTML pages from the Japan Times and processed them the same way as described above. 10K more news articles were gathered as the JPTimes dataset.

Although in this study we concentrate only on the textual content of the news articles, it is worth noting

that the HTML pages also provide additional information that can be helpful in generating keyphrases such as text style properties (e.g. bold, italic), links to related articles, or news categorization (e.g. politics, science, technology).

Data analysis

We explored the KPTimes dataset to better understand how it stands out from the existing ones. First, we looked at how editors tag news articles. Figure illustrates the difference between the annotation behaviour of readers, authors and editors through the number of times that each unique keyphrase is used in the gold standard. We see that non-expert annotators use a larger, less controlled indexing vocabulary, in part because they lack the higher level of domain expertise that editors have. For example, we observe that frequent keyphrases in KPTimes are close to topic descriptors (e.g. “Baseball“, “Politics and Government“) while those appearing only once are very precise (e.g. “Marley's Cafe“, “Catherine E. Connelly“). Annotations in KPTimes are arguably more uniform and consistent, through the use of tag suggestions, which, as we will soon discuss in §SECREF12, makes it easier for supervised approaches to learn a good model.

Next, we further looked at the characteristics of the gold keyphrases in KPTimes. Table shows that the number of gold keyphrases per document is similar to the one observed for KP20k while the number of missing keyphrases is higher. This indicates that editors are more likely to generalize and assign keyphrases that do not occur in the document ($\approx 55\%$). It is therefore this ability to generalize that models should mimic in order to perform well on KPTimes. We also note that keyphrases are on average shorter in news datasets (1.5 words) than those in scientific paper datasets (2.4 words). This may be due to the abundant use of longer, more specific phrases in scholarly documents BIBREF14.

Variants of keyphrases recovered from the metadata occur in 8% of the documents and represent 810

sets of variants in the KPTimes test split. These variants often refer to the same concept (e.g. “Marijuana; Pot; Weed“), but can sometimes be simply semantically related (e.g. “Bridges; Tunnels“). Thereafter, keyphrase variants will be used during model evaluation for reducing the number of mismatches associated with commonly used lexical overlap metrics.

Performance of existing models

We train and evaluate several keyphrase generation models to understand the challenges of KPTimes and its usefulness for training models.

Performance of existing models ::: Evaluation metrics

We follow the common practice and evaluate the performance of each model in terms of f-measure (F$_1$) at the top $N=10$ keyphrases, and apply stemming to reduce the number of mismatches. We also report the Mean Average Precision (MAP) scores of the ranked lists of keyphrases.

Performance of existing models ::: Models ::: Baseline: FirstPhrase

Position is a strong feature for keyphrase extraction, simply because texts are usually written so that the most important ideas go first BIBREF15. In news summarization for example, the lead baseline –that is, the first sentences from the document–, while incredibly simple, is still a competitive baseline BIBREF16. Similar to the lead baseline, we compute the FirstPhrases baseline that extracts the first N keyphrase candidates from a document.

Performance of existing models ::: Models ::: Baseline, unsupervised: MultipartiteRank

The second baseline we consider, MultipartiteRank BIBREF17, represents the state-of-the-art in unsupervised graph-based keyphrase extraction. It relies on a multipartite graph representation to enforce topical diversity while ranking keyphrase candidates. Just as FirstPhrases, this model is bound to the content of the document and cannot generate missing keyphrases. We use the implementation of MultipartiteRank available in pke BIBREF18.

Performance of existing models ::: Models ::: State-of-the-art, supervised: CopyRNN

The generative neural model we include in this study is CopyRNN BIBREF2, an encoder-decoder model that incorporates a copying mechanism BIBREF19 in order to be able to generate phrases that rarely occur. When properly trained, this model was shown to be very effective in extracting keyphrases from scientific abstracts. CopyRNN has been further extended by BIBREF3 to include correlation constraints among keyphrases which we do not include here as it yields comparable results.

Two models were trained to bring evidence on the necessity to have datasets from multiple domains. CopySci was trained using scientific abstracts (KP20k) and CopyNews using newspaper articles (KPTimes), the two models use the same architecture.

Performance of existing models ::: Results

Model performances for each dataset are reported in Table . Extractive baselines show the best results for KPCrowd and DUC-2001 which is not surprising given that these datasets exhibit the lowest ratio of absent keyphrases. Neural-based models obtain the greatest performance, but only for the dataset on which they were trained. We therefore see that these models do not generalize well across domains, confirming previous preliminary findings BIBREF2 and exacerbating the need for further research on this topic. Interestingly, CopyNews outperforms the other models on JPTimes and achieves very low scores

for KPCrowd and DUC-2001, although all these datasets are from the same domain. This emphasizes the differences that exist between the reader- and editor-assigned gold standard. The score difference may be explained by the ratio of absent keyphrases that differs greatly between the reader-annotated datasets and JPTimes (see Table), and thus question the use of these rather extractive datasets for evaluating keyphrase generation.

Finally, we note that the performance of CopyNews on KPTimes is significantly higher than that of CopySci on KP20k, proving that a more uniform and consistent annotation makes it easier to learn a good model.

Conclusion

In this paper we presented KPTimes, a large-scale dataset of newswire articles to train and test deep learning models for keyphrase generation. The dataset and the code are available at https://github.com/ygorg/KPTimes. Large datasets have driven rapid improvement in other natural language generation tasks, such as machine translation or summarization. We hope that KPTimes will play this role and help the community in devising more robust and generalizable neural keyphrase generation models.