zyte

In-depth analysis and evaluation on the quality of article body extraction

A comparison study of data quality from commercial services and open source libraries

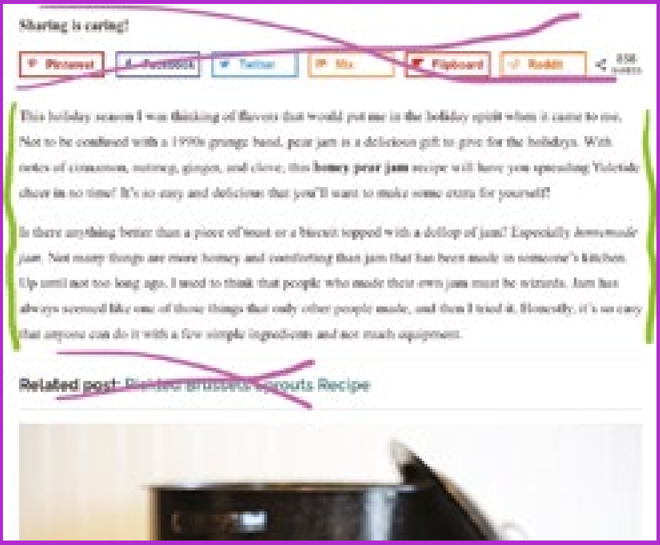
Written by
Konstantin Lopukhin
Data Science Team Lead

Introduction

In this report, we evaluated the quality of article body extraction for Zyte Automatic Extraction, Diffbot, and multiple open-source libraries such as Readability, Newspaper3k, Dragnet, Boilerpipe and html-text.

We define how the dataset was collected and describe the criteria used to evaluate a high-quality extraction. The results were analysed with conclusions and suggestions.

What is article extraction?

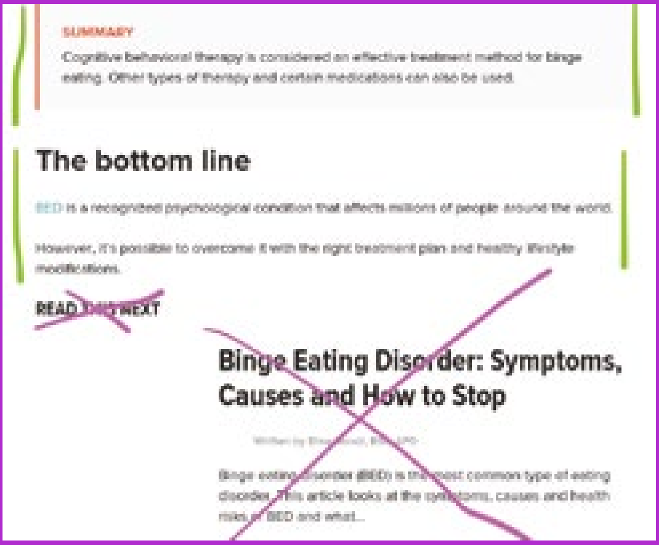

Sharing is caring!

Pinterest | Facebook | Twitter | Mix | Flipboard | Reddit | 858 SHARES

This holiday season I was thinking of flavors that would put me in the holiday spirit when it came to me. Not to be confused with a 1990s grunge band, pear jam is a delicious gift to give for the holidays. With notes of cinnamon, nutmeg, ginger, and clove, this **honey pear jam** recipe will have you spreading Yuletide cheer in no time! It's so easy and delicious that you'll want to make some extra for yourself!

Is there anything better than a piece of toast or a biscuit topped with a dollop of jam? Especially *homemade* jam. Not many things are more homey and comforting than jam that has been made in someone's kitchen. Up until not too long ago, I used to think that people who made their own jam must be wizards. Jam has always seemed like one of those things that only other people made, and then I tried it. Honestly, it's so easy that anyone can do it with a few simple ingredients and not much equipment.

Related post: Roasted Brussels Sprouts Recipe


SUMMARY

Cognitive behavioral therapy is considered an effective treatment method for binge eating. Other types of therapy and certain medications can also be used.

The bottom line

BED is a recognized psychological condition that affects millions of people around the world.

However, it's possible to overcome it with the right treatment plan and healthy lifestyle modifications.

READ THIS NEXT

Binge Eating Disorder: Symptoms, Causes and How to Stop

Written by [illegible]

Binge eating disorder (BED) is the most common type of eating disorder. This article looks at the symptoms, causes and health risks of BED and what...

Some examples of excluded content (crossed in cyan) and desired content (in green).

Article extraction is the task of extracting multiple fields from an article page given its URL. An „article" page could be a news article, blog post, press release, etc.

The main fields are the headline, article body (text of the article), cleaned HTML of the article, publication date, authors, main image and all other images. The article body is probably the most important field and the hardest to get right.

The task may look straightforward, but it is surprisingly tricky and nuanced.

Extracting the article body requires not only understanding where an article begins or ends, but also which parts to exclude from it such as ads, links to „proposed related" content, share buttons, author information, and other undesired elements.

Note

The article body may be a copyrighted work, so it is always best to obtain independent legal advice regarding the applicable copyright law when extracting such content. Some precautions include, do not republish the article body, engage in fair use, and provide attribution and links to the source if you publish any content from the article externally.

Our methodology

Systems included

We only considered generic extraction systems, which can work on any page or any domain without additional tuning. Writing a good rule-based article extraction system for just one domain is already hard, and maintaining it for thousands of domains is extremely labour intensive.

That's why all of the systems we evaluated are based on either machine learning or a set of generic heuristics, or both.

Datasets included

We collected a diverse set of URLs: news articles from popular and less popular sites, including blogs and non-news articles. The size of the dataset was 181 pages.

Both commercial services (Zyte Automatic Extraction and Diffbot) get a page URL, render it in a headless browser and extract various features, including visual features, HTML structure and text. Open-source libraries in this comparison take only HTML as input, which was obtained from the headless browser.

Metrics included

During the evaluation, we compare extraction results from each system with „ground truth" results, that is, desired output.

For a fair comparison, we made sure that all systems follow the same rules which we used for ground truth annotation. The metrics we decided to use to evaluate the quality of the extraction were precision, recall and F1, which are common choices for such evaluations.

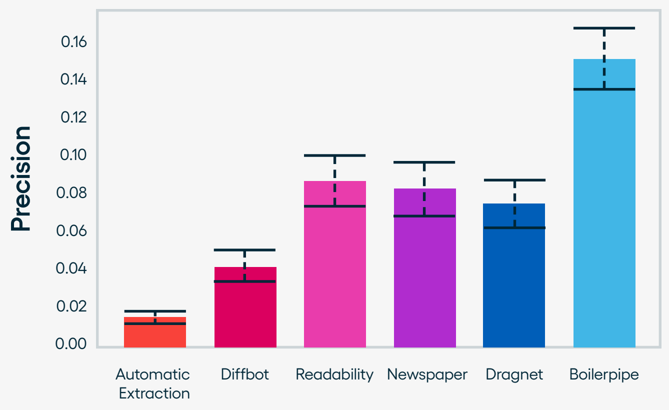
Precision here measures how well the systems are at excluding undesired parts of the article body

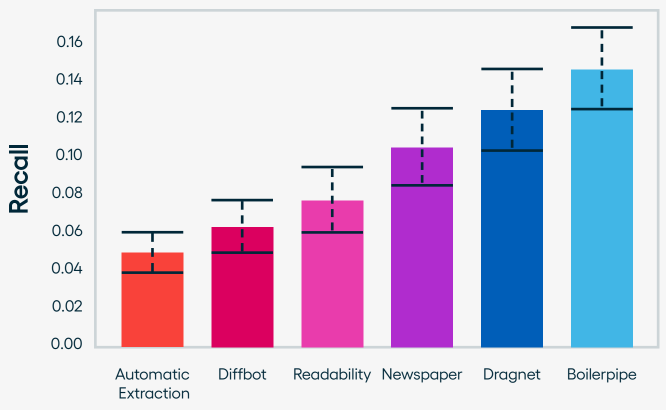
Accuracy which measures the ratio of "perfect" articleBody predictions, having exactly the same content as ground truth after normalization. This metric is easy to understand but quite noisy, as a single small difference makes the whole page a failure.

F1 is a summary of precision and recall measuring overall quality (a harmonic mean)

Recall measures how well the systems are at capturing desired parts of the article body

Let's check the results





	F1	Precision	Recall	single
Automatic Extraction	0.970 ± 0.005	0.984 ± 0.003	0.956 ± 0.010	0.470 ±0.038
Diffbot	0.951 ± 0.010	0.958 ± 0.009	0.944 ± 0.013	0.348 ± 0.036
Readability	0.922 ± 0.013	0.913 ± 0.014	0.931 ± 0.015	0.315 ± 0.035
Newspaper3k	0.912 ± 0.014	0.917 ± 0.014	0.906 ± 0.018	0.260 ± 0.033
Dragnet	0.907 ±0.014	0.925 ± 0.012	0.889 ± 0.019	0.221 ±0.031
Boilerpipe	0.860 ± 0.016	0.850 ± 0.016	0.870 ± 0.019	0.006 ± 0.006
html-text	0.665 ± 0.015	0.500 ± 0.017	0.994 ± 0.001	0.000 ±0.000

For the purpose of this study, we took F1 as the main single metric, but we think it's important to look at precision and recall as well as they can be more informative. For example, html-text which has the worst F1, has the best recall, because this is a baseline system which extracts all text from an article (can you guess why recall is not perfect?).

When we compared Zyte Automatic Extraction and Diffbot, we see that Automatic Extraction is better on all metrics, but only the precision difference is significant (in the table and charts above we show standard deviation estimated with bootstrap).

Automatic Extractions precision score of 0.984 means that the article body text extracted would only have 1.6% of unneeded content on an average webpage, compared to 4.2% for Diffbot.

While looking at recall values, we see that Automatic Extraction would miss 4.4% of desired content, compared to 5.6% for Diffbot.

All open-source libraries we studied had significantly worse metrics, but it's interesting that some modern libraries (such as Readability, Newspaper, Dragnet) are significantly better than an older Boilerpipe.

In terms of accuracy, the performance of the different systems is the same as for F1 results, but the difference is bigger. Notably, even the best score is only 0.47, so about half of the pages still have some differences with ground truth, however minor.

Conclusion & suggestions

Top system for quality

Automatic Extraction scored the highest across all the metrics we defined for quality on article body, with a significant difference for precision. These findings are backed by the feedback we have received from our customers who have extracted millions of articles. Still, there can be biases in our evaluation which we missed, so external evaluation is always welcome.

Quality is worse with Open Source libraries

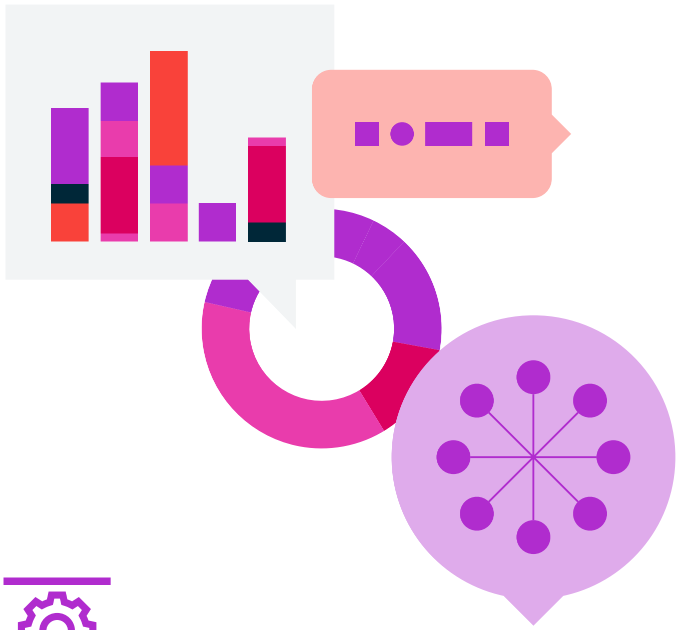
Quality of Open Source libraries is significantly worse than the quality of commercial services (e.g. even the most precise OSS library provides 4.6x more unwanted content in the results while missing 2.5x more content than Automatic Extraction). That said, they do work, and quality is not that bad; it can be adequate for some use cases, where unclean data is acceptable.

Avoid Boilerpipe Library

Boilerpipe library should be avoided if possible, and we would recommend using better-performing Python libraries as an option, as it provides significantly worse quality, especially in terms of precision.

Top system for content coverage

If the goal is to make sure no content is missing, use html-text or a similar "dumb" library because it gives the best recall score. Keep in mind that even basic html to text conversion is surpris-ingly tricky and easy to get wrong: simple approaches, like XPath string() functions, can produce noisy results: bad whitespaces, missing or extra text. Because of this reason, we suggest using a library, where these issues are solved.



Extra functionality from commercial services

Besides better quality, commercial services provide many other features. For example, Automatic Extraction can get you cleaned and normalized HTML of the article, article author, headline, date posted, images and many other attributes. Commercial services also handle downloading, which is a whole different aspect, significantly affecting the final result.

The experiment details

Rules used to define what should be included in article body

For a fair evaluation, we need to define what should be included in the article body to constitute it to being of high quality and make sure that all systems followed the same goals.

The underlying principle is that the article body should be “clean text”, without other fields such as author, navigation elements, ads, etc.
At the same time, we want to include all sub-headings, block-quotes and other elements which constitute article content.

Here is a list of items we defined should not be included as an article body:

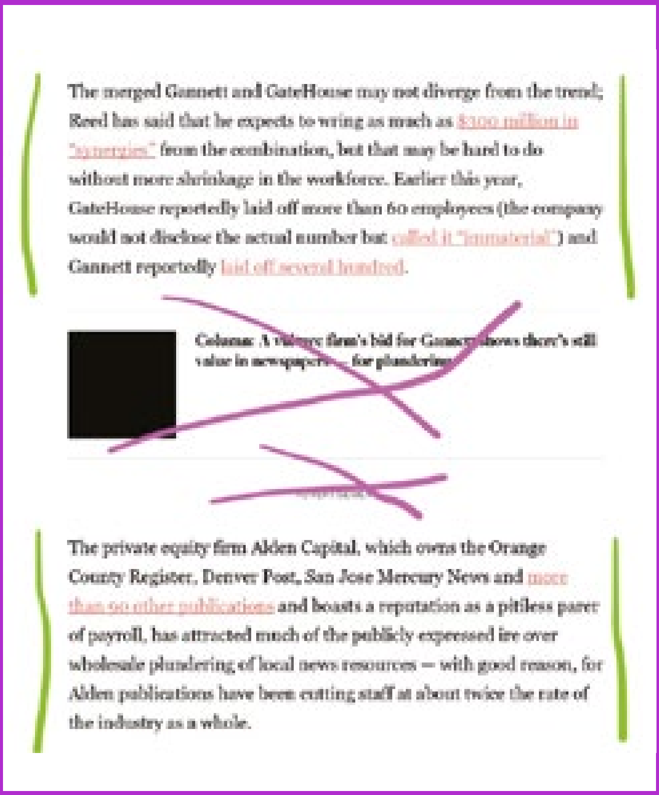
- Fields such as authors, publication date, keywords, image captions and markup around them
- Share buttons and suggestions to share the article
- Comments and UI related to comments (e.g. “Comments” header, forms, links to comment)
- Any links which are not part of the article and whose goal is to keep the reader on the platform such as “related” articles, „read next” links, „recommended to you”, etc.
- Newsletter signup links, generic calls to action, various forms
- Advertisments
- Author information, usually at the end of the article
- Control elements around images which produce unneeded text e.g. the number of images in a gallery, buttons overlaid over an image/video, etc.
- Blocks denoting the reading time of the article
- Other items unrelated to the article content

Here is a list of items that should be included as the article body:

- Article body proper
- Links to read in more detail, to the source, or to other content directly related to this article: these might require a deeper understanding of the article content to annotate correctly
- Embedded posts related to the article, such as tweets / Instagram / Facebook posts
- Embedded posts related to the article, such as tweets / Instagram / Facebook posts
- Copyright notices at the end

Both Zyte Automatic Extraction and Diffbot systems follow the above guidelines with both occasionally making understandable errors.

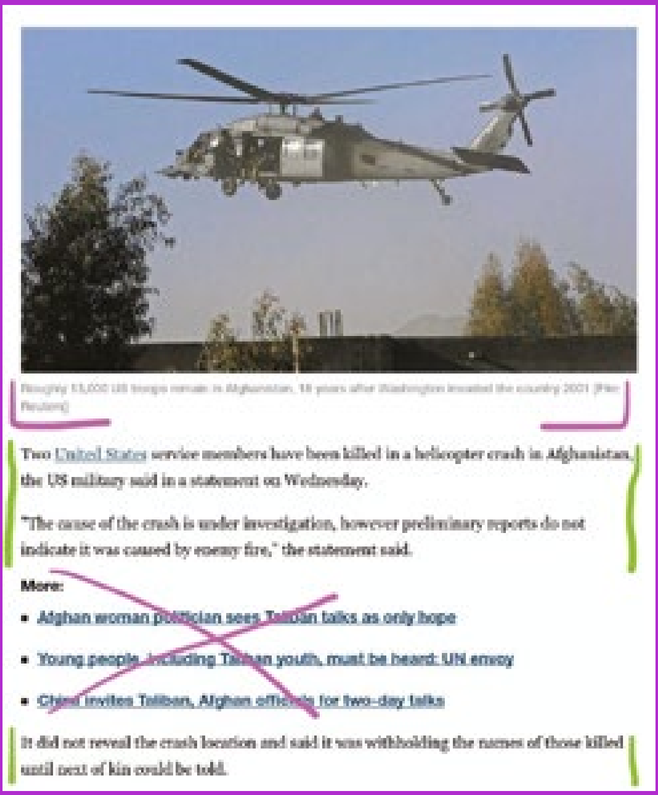
Even though these rules are a good default for most use cases, and reasonable for this evaluation study other use cases may require something different.For example, if the article body is passed as an input to an ML-based classifier (e.g. to categorize articles), it could be beneficial to keep image captions in the article body itself. With Zyte Automatic Extraction it is possible to customize the output using the articleBodyHtml field - see this blog post.



The merged Gannett and GateHouse may not diverge from the trend; Reed has said that he expects to wring as much as $300 million in "synergies" from the combination, but that may be hard to do without more shrinkage in the workforce. Earlier this year, GateHouse reportedly laid off more than 60 employees (the company would not disclose the actual number but called it "immaterial") and Gannett reportedly laid off several hundred.

~~Column: A vulture firm's bid for Gannett shows there's still value in newspapers — for plundering~~

The private equity firm Alden Capital, which owns the Orange County Register, Denver Post, San Jose Mercury News and more than 90 other publications and boasts a reputation as a pitiless parer of payroll, has attracted much of the publicly expressed ire over wholesale plundering of local news resources — with good reason, for Alden publications have been cutting staff at about twice the rate of the industry as a whole.



Roughly 13,000 US troops remain in Afghanistan, 18 years after Washington invaded the country 2001 [File: Reuters]

Two United States service members have been killed in a helicopter crash in Afghanistan, the US military said in a statement on Wednesday.

"The cause of the crash is under investigation, however preliminary reports do not indicate it was caused by enemy fire," the statement said.

~~**More:**~~

- ~~Afghan woman politician sees Taliban talks as only hope~~
- ~~Young people, including Taliban youth, must be heard: UN envoy~~
- ~~China invites Taliban, Afghan officials for two-day talks~~

It did not reveal the crash location and said it was withholding the names of those killed until next of kin could be told.

Here you can find a gallery with some examples, where desired content is highlighted in green, and undesired content is crossed-out or highlighted in cyan.

How was the dataset collected?

We wanted the dataset to be diverse, representative and unbiased.

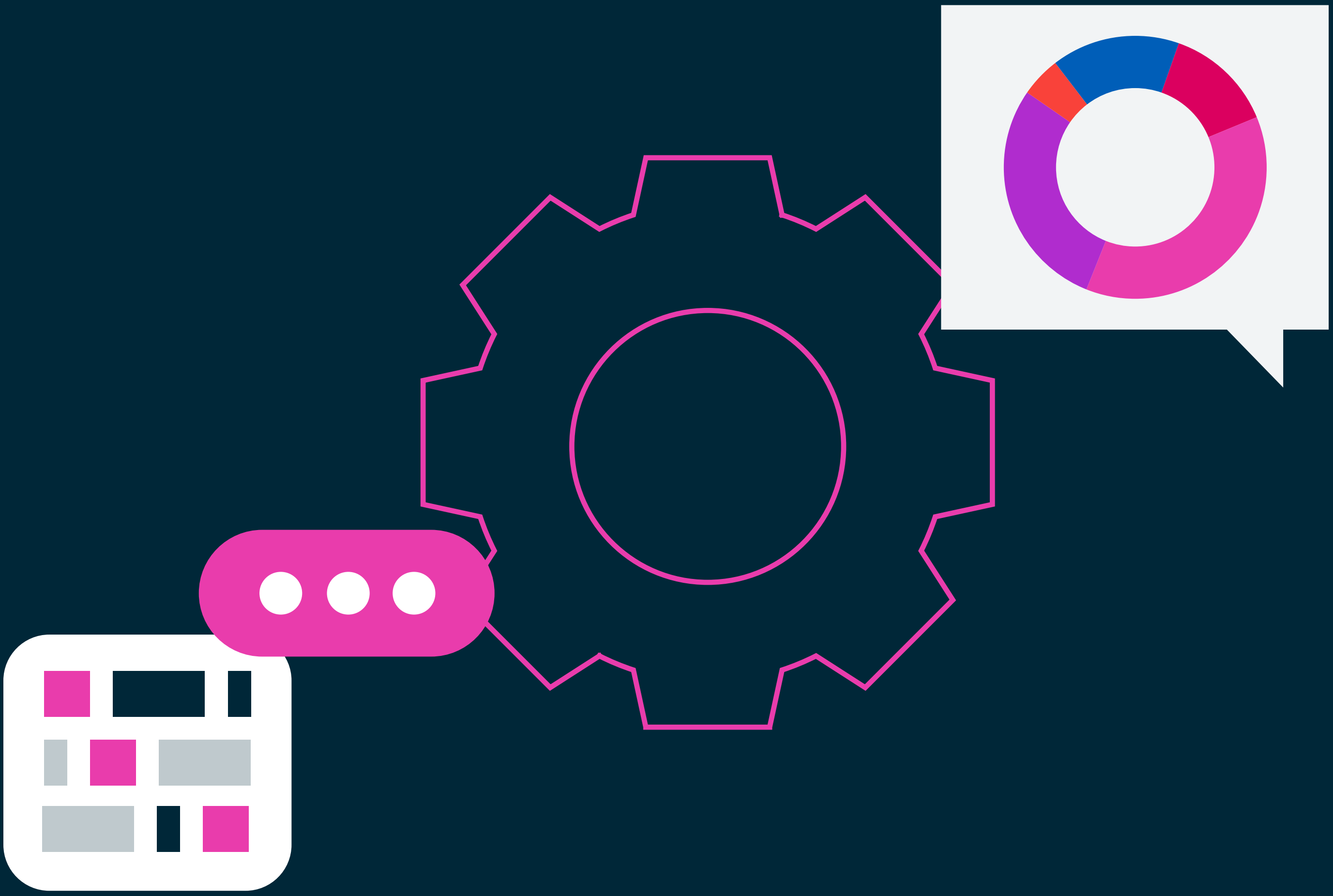
By diversity, we mean that it should contain different kinds of pages in different languages, not just news articles in English, but also a variety of non-news articles.

A representative dataset should contain pages which are similar to what people want to achieve by article extraction, including modern complex news articles.

We wanted the dataset to be unbiased in a sense that it should be different from our internal datasets in terms of how the initial URLs were collected, to avoid biasing results in our favor.

URL collection

We did URL collection in two stages, both being significantly different from the approach we use for internal training or testing datasets.



Stage 01:
Obtaining long-tail URLs

We took a random sample of 1,000 domains from the top million most popular sites according to Alexa.

For each site, we checked if it contained any articles, and if it did, we selected two random articles from it. This resulted in a very diverse set of pages in different languages, including not only news articles, but also blog posts, non-news articles, press releases, etc.

We obtained around 120 URLs from around 60 domains this way.

Stage 02:
News articles from modern popular sites

News articles from modern popular sites. We went to https://news.google.com/ in incognito mode, set Language and region to English (United States) and collected all links to articles, getting around 190 URLs from 90 domains. Next, we went to a news aggregator http://www.gigablast.com/news and obtained all article links from the first 3 pages, getting around 60 URLs from 40 domains.

We combined all URLs from Stage 1 and Stage 2, excluding youtube links, obtaining 356 URLs from 189 domains as many domains had more than 10 pages.

Article extraction

We extracted articles from all the seed URLs via Zyte Automatic Extraction and Diffbot at the same time, minimizing the chances of the article being updated. We also made sure that the pages didn't change by analyzing extraction results. If one of the systems was not able to download a page, or a downloaded page had unexpected content, we excluded it from the evaluation entirely, to focus on extraction quality only.

Open Source libraries were not supposed to download anything, and we didn't want to penalize Diffbot for having fewer pages downloaded, as it could be a temporary issue. Analysis of download success/failure rates is out of scope for this comparison. HTML files for open-source libraries were obtained from the headless browser output.

Article annotation

A maximum of two pages per domain was annotated. After performing an initial evaluation, we had to exclude a few pages which we could not evaluate properly where clearly different content was downloaded by Automatic Extraction and Diffbot (e.g. Diffbot didn't handle some non-ascii URLs well).

We also excluded 6 pages in Japanese and Korean where slightly different whitespace handling made it hard to compare results using. In the end, we were left with 181 pages with a non-empty article body. This dataset is released, see below for more details on where you can access it.

How were the metrics defined?

Measuring precision, recall and F1 are the common and sensible choice for this evaluation. For this particular task, precision measures how "clean" the output article body was - in other words, how effectively unneeded content was excluded.

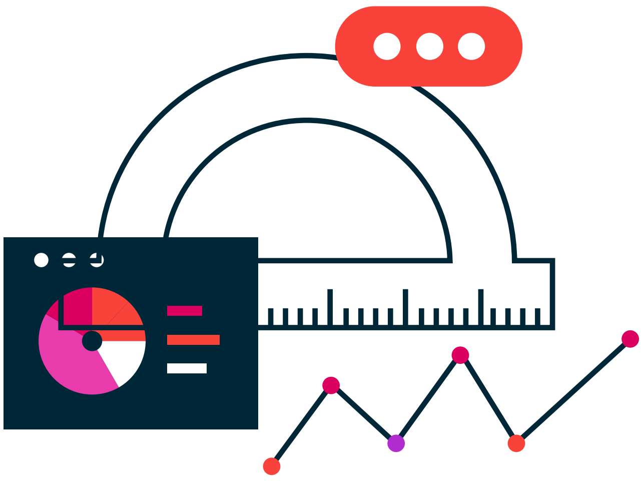
Recall measures how the system preserves the desired parts of the article body. And F1 is a common way to combine precision and recall.

Precision and recall

But how do we actually compute precision and recall? A possible choice is to treat the extraction result as a set of words (also known as “bag” of words) and then calculate precision and recall based on such sets. Our approach was different in two aspects:

First, we use n-grams instead of words: this makes sense because single words can be seen both in the desired text of the article and in undesired part of it, while for n-grams (we use 4-grams) it’s significantly less likely.

Secondly, we considered counts of n-grams as a set of words does not penalize unneeded repeated text. You can check the full metric implementation in the evaluation script on Github.



This explains why html-text, which extracted all text, didn’t get a perfect recall score: because if we need to exclude some text in the middle, e.g. “bad” from “good good bad very good”, then ground truth output would contain 4-gram “good good very good”, while html-text output would lack it.

Tokenization

The final aspect of the metric is how to obtain n-grams (in other words, Tokenization): for that we use a very simple approach of extracting all alpha-numeric sequences, discarding extra whitespace and punctuation. For example “- This is a 2020/01 article!Next” input would be normalized to “This is a 2020 01 article Next”. This works well in most cases, as usually text to be excluded lies in separate blocks separated by newlines or spaces.

But there are some interesting cases here: first, different systems might insert whitespace in different places, e.g. if we have HTML like “aa<span>bb</span>” then it’s not immediately clear if a space should be added between “aa” and “bb”. In practice, we have seen some weird markup like “<span>A da</span>y in …”, where it’s clear that “A day in …” is correct.

Another case when this tokenization approach can be problematic is languages like Japanese, where words commonly don’t have spaces separating them, and a small difference in space handling can lead to big changes in the final metric. For this comparison, we opted to exclude 6 pages in Japanese and Korean where we were not confident in our Ground Truth and didn’t have a smart enough Tokenizer for evaluation.

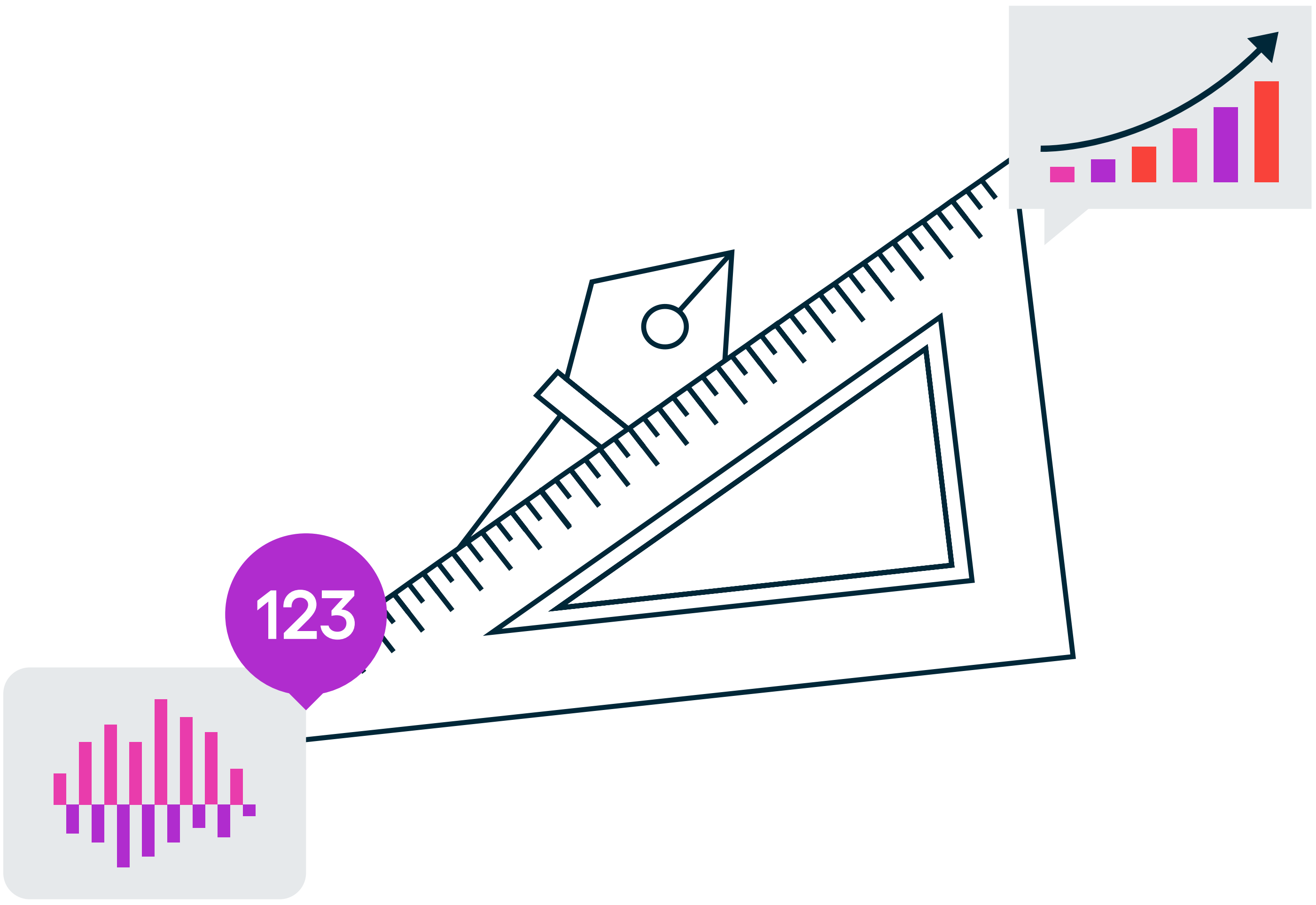
Excluding them doesn’t mean the tools we’re comparing were not working with these languages, it is just that evaluation was a bit more complicated in these cases, and we decided those few pages were not important enough.

Accuracy

Additionally to measuring precision, recall and F1, we added accuracy, which measures the ratio of “perfect” article bodies, which have exactly the same content as ground truth after normalization. This metric is easy to understand but quite noisy, as a single small difference renders the whole page as a failure. Normalization here was performed by joining all tokens with whitespace, using the same tokenization as above.

Uncertainty

Since the dataset is quite small (only 181 pages), there is a great deal of uncertainty in results. Bootstrap allows us to quantify this uncertainty: we took 1000 samples from the dataset with replacement and calculated mean and standard deviation. This allows us to see that difference in precision is much more significant compared to recall differences, and that accuracy is indeed quite noisy, as you can see in the results table.



How were errors analysed?

We did an extensive comparison of ground truth and extraction results for Zyte Automatic Extraction and Diffbot and noticed that a lot of errors were similar but there were some differences.

- A common error is when some extra text is present at the end of the article, e.g. author information – such cases are often hard to judge during annotation as well.

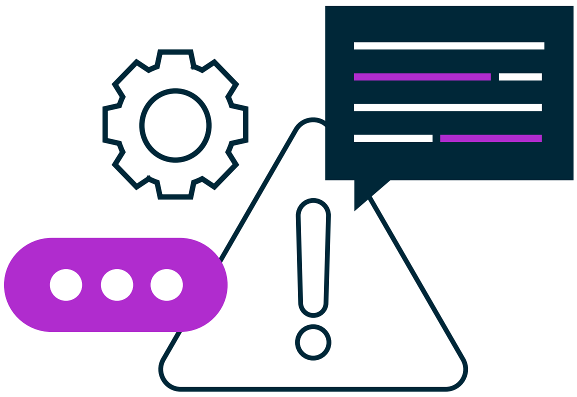
- In some cases both systems exclude links to the original article or to the source towards the end, confusing them with links to unrelated content which also often come at the end in a similar style. Correct annotation often requires understanding the article in question, so it's not surpris-ing automated systems struggle here.

- Figure caption present in the output is a more common error for Diffbot, especially when there are multiple elements around the image, e.g. one for copyright and one which describes what is in the image, although Automatic Extraction also has some amount of errors here.

- Subheadings in non-default style (not black) are sometimes erroneously excluded by Diffbot, perhaps they are confused with headings for unrelated links.

- In some cases, Automatic Extraction inserts unneeded spaces around inline tags, e.g. "Macbook s" instead of "Mac-books"

- Not excluding all "sharing" sugges-tions happens with both systems, although it's quite rare.



- Diffbot is more likely to exclude tables or more often parts of tables, although it's a rare occurrence. In a few cases, Diffbot output stopped too early, with large portions of the article body missing – e.g. if an article was broken up with a big advertising block.

- Failure to exclude links to "related" (not really) content seemed to happen more for Diffbot than to Automatic Extraction.

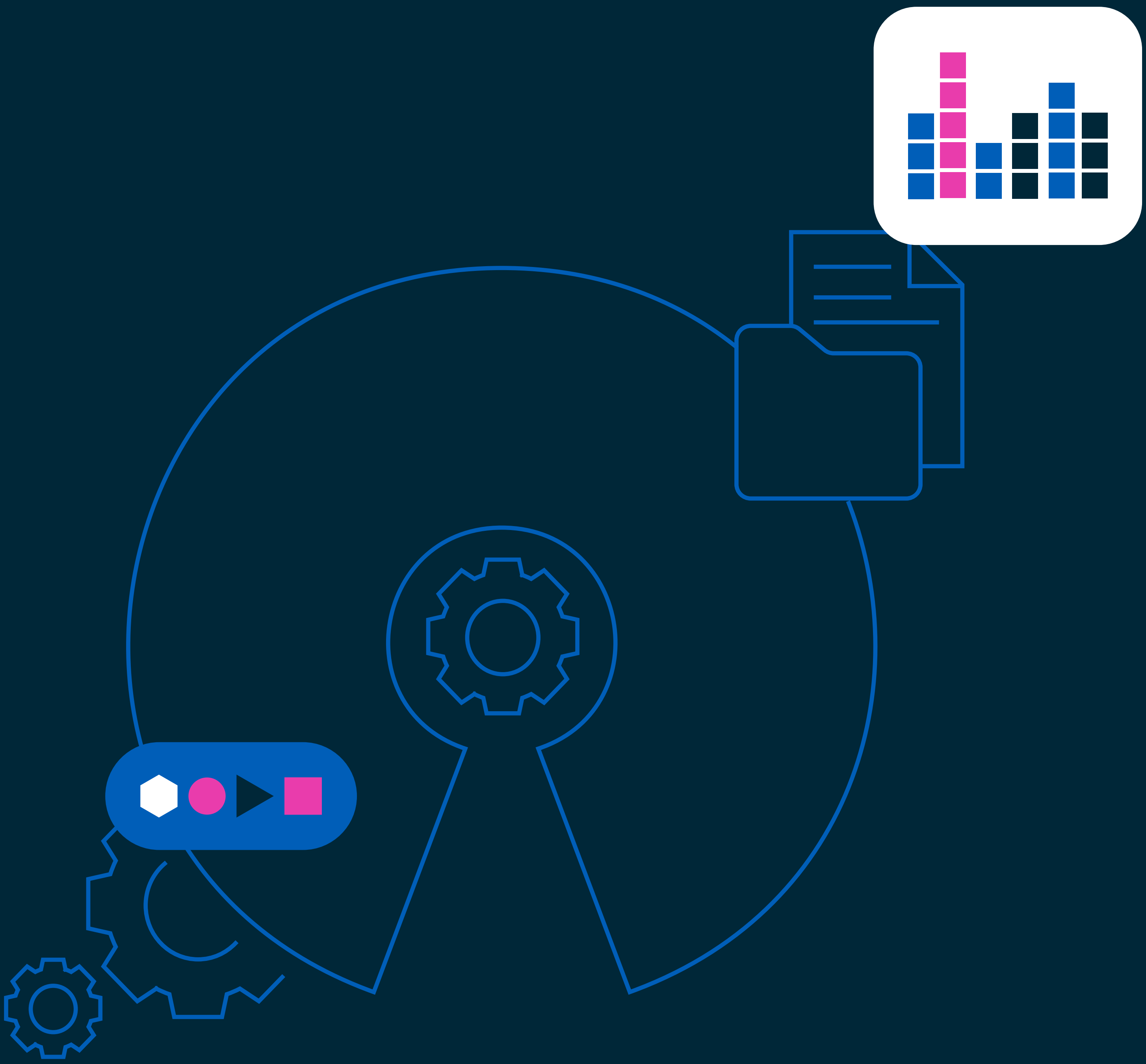
But overall, the errors both systems were making were quite similar and expected, given how complicated and nuanced article extraction can be. There is always room for improvement though!

We actually fixed some of the Automatic Extraction issues found during this evaluation; of course, these improvements were not included in the metrics as we wanted an unbiased evaluation of both systems.

What open-source libraries were used?

We selected some popular and well-supported libraries such as Readability, we also added Dragnet as it was featured in a moz.com benchmark, and also an older but well-known boilerpipe library.

No retraining or tuning was performed, except for choosing Boilerpipe extractor which provided the best score.



Newspaper3k:
news, full-text, and article metadata extraction in Python 3.

Readability-lxml:
fast python port of arc90's readability tool, updated to match latest readability.js.

Dragnet:
an ML web page content extraction library.

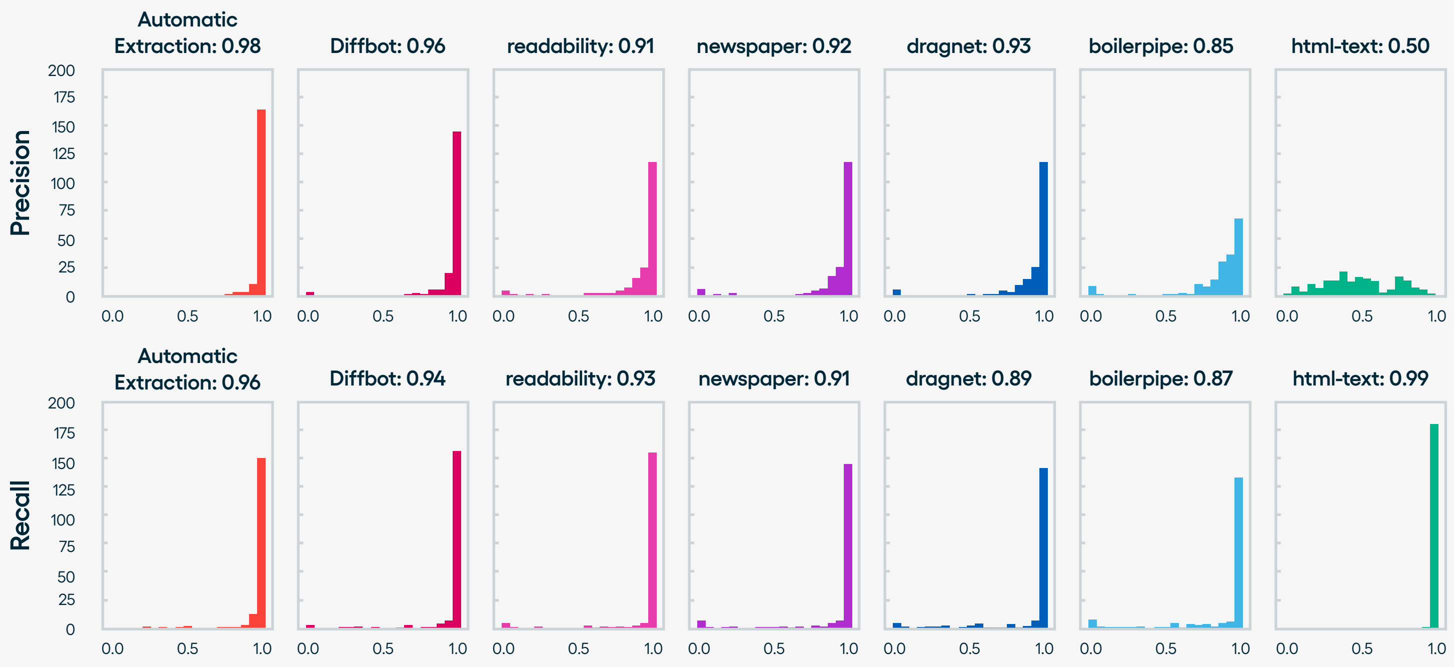
Boilerpipe:
a python interface to Java boilerpipe library.

Html-text:
a baseline which extracts the full text of an HTML page.

When checking predictions qualitatively, we noticed that open-source libraries make two kinds of errors:

- Not excluding unrelated content: This is expected given our high requirements for what should be excluded. Also, not all libraries might follow the same article body criteria; Eg. Boilerpipe tends to include a lot more content both at the start and at the end of the article which we consider to be unrelated.
- Occasionally making quite significant errors when determining the article body, with large chunks excluded or included incorrectly.

These charts show how precision and recall scores are distributed across different pages for all tested systems:



While the first kind of error happens with Automatic Extraction and Diffbot as well (although less often), the second kind is more serious and is extremely rare for both commercial systems.

We realise that in our comparison open-source libraries work on bare HTML, while commercial services have access to the rendered page, but we are not aware of any open-source libraries that use a similar approach.

Dataset and evaluation released on Github

If you want more information around these evaluations we have released the dataset and scripts on Github. There you will find:

Evaluation scripts

Contain full metric implementation, including bootstrap estimation code, and allow for results to be reproduced. Python 3.6 or later, no dependencies required.

Datasets

Ground truth in JSON, with “articleBody” and “url” fields. HTML sources rendered in a headless browser, utf-8 encoded and gzipped, plus screenshots.

zyte

Try Automatic Extraction news API for high-quality article extraction at scale!

If you’re in need of news or article data try our api for free, and save time & effort through automatic web data extraction at scale. Sign up here!

www.zyte.com
Copyright 2021 © Zyte

Cuil Greine House, Ballincollig Commercial Park Link Road, Ballincollig / Co. Cork, Ireland