Making Jekyll related posts really related!

📅 Jul 21, 2016   👁 2346 Views   🏷 Jekyll   ⏱ 9 Mins read

Why should we use Related Posts?

If you have a lot of great content on your website and you want it easier for your users to see it then you should use **related posts**.

A viewer landing on one of your posts may be hungry for a similar type of information. You should have an easy way for them to browse around. This is where related posts come into the picture. Related posts are those articles which are related to the current article. They are usually at the bottom of the post.

> Related posts help you decrease the bounce rate(percentage of visitors to a particular website who navigate away from the site after viewing only one page).

Contents

What's wrong with Jekyll related_posts ?

As I have discussed in my earlier post `site.related_posts` should be called `site.recent_posts` because it just shows 10 recent posts. Your recent posts can be on a different category but, this variable doesn't recognize that. I was looking for a better way to solve this issue.

A possible solution was to look for posts with matching category or tag. This will make sure the related posts are actually related. To make it even better we are comparing more categories or tags. If they match then those posts will show up.

Pre-requisites

Assigning categories or tags is the first step in using this method of extracting related posts. You should categorize all your posts into few topics. If you are not sure how to use tags or categories then please refer to How to use Jekyll categories. Proceed to other next only after this.

How to implement a better Jekyll Related Posts?

Use the below liquid syntax code to get 4 related posts. You can change many parameters here. I picked it up from a post on stackoverflow. This works if you are using **tags**.

</> HTML

```
<div class="relatedPosts">

<h4>You May Also Enjoy</h4>

{% assign maxRelated = 4 %}
{% assign minCommonTags =  2 %}
{% assign maxRelatedCounter = 0 %}

  {% for post in site.posts %}

    {% assign sameTagCount = 0 %}
    {% assign commonTags = '' %}

    {% for tag in post.tags %}
      {% if post.url != page.url %}
        {% if page.tags contains tag %}
          {% assign sameTagCount = sameTagCount | plus: 1 %}
          {% capture tagmarkup %} <span class="label label-default">{{ tag }}
</span> {% endcapture %}
          {% assign commonTags = commonTags | append: tagmarkup %}
        {% endif %}
      {% endif %}
    {% endfor %}

    {% if sameTagCount >= minCommonTags %}
      <div>
      <h5><a href="{{ site.baseurl }}{{ post.url }}">{{ post.title }}{{ commo
nTags }}</a></h5>
      </div>
      {% assign maxRelatedCounter = maxRelatedCounter | plus: 1 %}
      {% if maxRelatedCounter >= maxRelated %}
        {% break %}
      {% endif %}
    {% endif %}

  {% endfor %}

</div>
```

If you are using **categories** then use the code below.

```html
</> HTML

<div class="relatedPosts">

<h4>You May Also Enjoy</h4>

{% assign maxRelated = 4 %}
{% assign minCommonTags =  2 %}
{% assign maxRelatedCounter = 0 %}

{% for post in site.posts %}

    {% assign sameTagCount = 0 %}
    {% assign commonTags = '' %}

    {% for category in post.categories %}
      {% if post.url != page.url %}
        {% if page.categories contains category %}
          {% assign sameTagCount = sameTagCount | plus: 1 %}
          {% capture tagmarkup %} <span class="label label-default">{{ catego
ry }}</span> {% endcapture %}
          {% assign commonTags = commonTags | append: tagmarkup %}
        {% endif %}
      {% endif %}
    {% endfor %}

    {% if sameTagCount >= minCommonTags %}
      <div>
      <h5><a href="{{ site.baseurl }}{{ post.url }}">{{ post.title }}{{ commo
nTags }}</a></h5>
      </div>
      {% assign maxRelatedCounter = maxRelatedCounter | plus: 1 %}
      {% if maxRelatedCounter >= maxRelated %}
        {% break %}
      {% endif %}
    {% endif %}

  {% endfor %}

</div>
```

Brief explaination of the code

`{% assign maxRelated = 4 %}` - Maximum number of related posts that will be shown in the output.

`{% assign minCommonTags = 2 %}` - Minimum number of tags or categories that should be equal to be considered as related. In this case, at least 2 tags or categories should match with the current post to be considered as related.

`{% if post.url != page.url %}` - Current post will not be shown in the related posts output

The code basically compares the current posts' tags or categories with other posts. If the `minCommonTags` is satisfied then it will show such posts in the output.

If our post has tags `Jekyll` `Github` and `Web-Design` then the code looks for all the posts with at least two common tags. Which means posts with tags `Jekyll` and `Github` are candidates,

ones with `Github` and `Web-Design` are candidates, ones with `Jekyll` and `Web-Design` are candidate and ones with all three tags are also candidates.

If there are more than `maxRelated` posts are available then only the recent ones are considered.

How to increase the accuracy of Jekyll Related Posts?

`minCommonTags` is directly proportional to the relation between posts. If posts have more common tags then they must be related isn't it? This code uses the same logic. You can increase this number to get highly targeted related posts but sometimes you may get no output.

Use this variable carefully so that for every post you get some output. I have given it a value `1` because at this point I don't have too much content on my website and I use less number of tags.

Jekyll Related Posts for collections

If you are new to collections then please refer to 3 Easy steps to implement Jekyll Collections. In my blog, I have a theme collection. I wanted related posts(themes) on that as well.

Here is the code I have used. This may change according to your collection.

```html
</> HTML

<h4>You May Also Enjoy</h4>
{% if page.url contains 'jekyll-themes' %}
<div class="relatedPosts">

    {% assign maxRelated = 4 %}
    {% assign minCommonTags =  1 %}
    {% assign maxRelatedCounter = 0 %}

    {% for jekyll-themes in site.jekyll-themes %}

    {% assign sameTagCount = 0 %}
    {% assign commonTags = '' %}

    {% for tag in jekyll-themes.tags %}
      {% if jekyll-themes.url != page.url %}
        {% if page.tags contains tag %}
          {% assign sameTagCount = sameTagCount | plus: 1 %}
          {% capture tagmarkup %} <span class="label label-default">{{ tag }}
</span> {% endcapture %}
          {% assign commonTags = commonTags | append: tagmarkup %}
        {% endif %}
      {% endif %}
    {% endfor %}

    {% if sameTagCount >= minCommonTags %}
      <div>
      <h5><a href="{{ site.baseurl }}{{ jekyll-themes.url }}">{{ jekyll-theme
s.title }}{{ commonTags }}</a></h5>
      </div>
      {% assign maxRelatedCounter = maxRelatedCounter | plus: 1 %}
      {% if maxRelatedCounter >= maxRelated %}
        {% break %}
      {% endif %}
    {% endif %}

 {% endfor %}

</div>

{% else %}

//Related posts code for usual posts
{% endif %}
```

How I have implemented Related Posts?

I have implemented the same method but with the value of variable `minCommonTags` is 1. I'm showing 4 posts at the bottom of every post along with thumbnails. Here is the verbatim code I'm using.

</> HTML

```
<div class="rect">

  {% assign maxRelated = 4 %}
  {% assign minCommonTags =  1 %}
  {% assign maxRelatedCounter = 0 %}


  {% for post in site.posts %}


    {% assign sameTagCount = 0 %}
    {% assign commonTags = '' %}

    {% for tag in post.tags %}
      {% if post.url != page.url %}
        {% if page.tags contains tag %}
          {% assign sameTagCount = sameTagCount | plus: 1 %}
          {% capture tagmarkup %} <span class="label label-default">{{ tag }}
</span> {% endcapture %}
          {% assign commonTags = commonTags | append: tagmarkup %}
        {% endif %}
      {% endif %}
    {% endfor %}

    {% if sameTagCount >= minCommonTags %}

        <a href="{{ site.baseurl }}{{ post.url }}">
            <div class="rel">
                <h5>{{ post.title }}</h5>
                <img width="100%" src="/thumbs/{{post.image}}">
            </div>
        </a>

      {% assign maxRelatedCounter = maxRelatedCounter | plus: 1 %}
      {% if maxRelatedCounter >= maxRelated %}
        {% break %}
      {% endif %}
    {% endif %}
  {% endfor %}
</div>
```

I'm using some CSS to give it a nice look.

</> CSS

```
.rect {
    border-top: 3px solid #333;
    border-bottom: 3px solid #333;
    padding-top: 15px;
    margin:0 auto;
}

.rel {
    width: 200px;
    display: inline-block;
    padding: 5px;
    opacity: 0.8;
}

.rel hover {
    opacity: 1;
}
```

Here is a screenshot of the same

Conclusion

Even though Jekyll did not provide a simple variable to fetch related content, the community has found some ways to overcome that. Do implement this instead of simple `related_posts` because the links should be related to the content of the current post.

Let me know if you are facing issues while implementing this. Leave any suggestions in the comment section.

Thanks for reading!

Also read

JEKYLL DATA - HOW IT...

4 STEPS TO MIGRATE F...

JEKYLL SITE SCORING ...

JEKYLL SNIPPETS - JU...

JEKYLL REDIRECT 301

JEKYLL PAGINATION

ALSO ON WEBJEDA

14 Ways to Customize your Company Website

4 years ago • 1 comment

Every company purchases a domain and creates a website with the hope of …

The easiest search option for Jekyll

4 years ago • 2 comments

Why do we need a search option? Having a search bar in your Jekyll blog helps …

How to Create a Beautiful Jekyll Blog?

3 years ago • 16 comments

I created this beautiful looking Jekyll blog using default Jekyll template. …

19 Comments   WebJeda   🔒 Disqus' Privacy Policy

♡ Recommend 3   Tweet   f Share   Sort

Join the discussion…

My Writing Corner • a year ago

I know this post is old, but thanks for sharing! I just found it and implemented it already
like a charm!

2 ∧ | ∨ • Reply • Share ›

Oliver Ullman • 3 years ago

Extremely helpful, Thank you!

2 ∧ | ∨ • Reply • Share ›

Michele Nasti • 2 years ago • edited

Hello there! I have a dumb question. on my blog I have only two articles with the same
When I read one of these two articles, I see in the "related post" the other. When I click
second, I see the first.

Is there any way to actually avoid this behaviour?

Since I have an infinite scrolling set up on my blog, readers would be stuck in these tw
arrticles...

1 ∧ | ∨ • Reply • Share ›

Giovanni K Bonetti • 8 months ago

Thanks for the post! We ended up switching this custom logic by this machine learning
that is very easy to use - https://github.com/jekyll/c...

∧ | ∨ • Reply • Share ›

Nicolas HOIZEY • 3 years ago

I tried to use the https://github.com/alfanick... plugin, but it slows down site generation

I think you could swap these two lines in your first code:

{% for tag in post.tags %}

{% if post.url != page.url %}

∧ | ∨ • Reply • Share ›

sherwin • 4 years ago

what if there is no related post? will the "You May Also Enjoy" be displayed regardless

∧ | ∨ • Reply • Share ›

> **sharathdt** webjeda → sherwin • 4 years ago
>
> Then it will show latest posts.
>
> ∧ | ∨ • Reply • Share ›

>> **sherwin** → sharathdt • 4 years ago
>>
>> sorry, i don't seem to follow. i don't see the conditional statement that ha
>> minCommonTags is initialized as 1. in order to increment sameTagCoun
>> check against minCommonTags you have to go through the for loop. the
>> of the text "You May Also Enjoy" is handled outside the for loop. i've bee
>> implement your example but if there's no related post, the "You May Also
>> text is displayed but nothing below. was hoping there's a way to hide it if
>> no related post. thanks for the help
>>
>> ∧ | ∨ • Reply • Share ›

>>> **sharathdt** webjeda → sherwin • 4 years ago • edited
>>>
>>> Oh I see the problem here. You can actually customize it I believe
>>> text "You May Also Enjoy" - if there are no related post - seems h
>>> since it is outside of the for loop. But you can have another cond
>>> inside after
>>>
>>> ```
>>> {% if sameTagCount >= minCommonTags %}
>>> ```
>>>
>>> as
>>>
>>> ```
>>> {% if sameTagCount == 0 %}
>>>     show latest posts
>>> ```
>>>
>>> Here is the full code
>>>
>>> ```
>>>     {% assign commonTags = '' %}
>>>     {% for tag in jekyll-themes.tags %}
>>> ```

```
{% if jekyll-themes.url != page.url %}
  {% if page.tags contains tag %}
    {% assign sameTagCount = sameTagCount | plus: 1 %}
    {% capture tagmarkup %} {{ tag }} {% endcapture %}
    {% assign commonTags = commonTags | append: tagmarkup %}
  {% endif %}
```

see more

⌃ | ⌄ • Reply • Share ›

sherwin ➜ sharathdt • 4 years ago

that didn't work the way i wanted. but i think i found a way. i adde
variable called isHeaderDisplayed and set it to true. during the di
of post.title i displayed the header text i wanted then quickly set i
now it only displays during the first pass of the loop.

```
{% if sameTagCount >= minCommonTags %}
{% if isHeaderDisplayed %}
<h4>Related Posts</h4>
<div class="list-group">
{% endif %}
{{ post.title }}
{% assign isHeaderDisplayed = false %}
{% assign maxRelatedCounter = maxRelatedCounter | plus: 1 %}
{% if maxRelatedCounter >= maxRelated %}
{% break %}
</div>
{% endif %}
{% endif %}
```

⌃ | ⌄ • Reply • Share ›

sharathdt webjeda ➜ sherwin • 4 years ago

:) if it works as expected then no need to worry.

⌃ | ⌄ • Reply • Share ›

sherwin ➜ sharathdt • 4 years ago

thanks, i will try it

⌃ | ⌄ • Reply • Share ›

Duc Nguyen • 4 years ago

thanks you, used it for my blog

⌃ | ⌄ • Reply • Share ›

Duc Nguyen ➜ Duc Nguyen • 4 years ago

But it always shows latest post of the category. can it show random posts which
same category?

⌃ | ⌄ • Reply • Share ›

sharathdt webjeda ➜ Duc Nguyen • 4 years ago

@disqus_5eJEnjCPfN thanks for the comment. Unfortunately, Jekyll Liq
no random variable or tag so you have to generate one by yourself base
current time.

```
{% assign random = site.time | date: "%s%N" | modulo: site.posts.size %}
<h1>{{ site.posts[random].title }}</h1>
```

Here is the link:
http://stackoverflow.com/qu...

⌃ | ⌄ • Reply • Share ›

Duc Nguyen ➜ sharathdt • 4 years ago • edited

thanks for your answer. I will try it next time. I applied your tips an
result https://trucnt.com

⌃ | ⌄ • Reply • Share ›

sharathdt webjeda ➜ Duc Nguyen • 4 years ago

That looks wonderful :)

⌃ | ⌄ • Reply • Share ›

Xab • 4 years ago

awesome! thank you!

⌃ | ⌄ • Reply • Share ›

sharathdt webjeda ➜ Xab • 4 years ago

@Xabaras89 welcome!

1 ^ | ⌄ • Reply • Share ›

✉ **Subscribe**   ⒟ **Add Disqus to your site**Add Disqus**Add**   ⚠ **Do Not Sell My Data**

WEBJEDA

WebJeda is a company oriented towards building modern websites and apps.

▶ YouTube   f   𝕏   github   ♥

Quick Links

About

Contact

Sitemap

Themes

Contact us

✉ Get email address

🌐 Visit our website

🖼 Image credits - Freepik

© 2020 | All rights reserved

Get free updates.

Enter your email

JOIN US