

## 问题分析：

如果对一个 string field 进行排序，结果往往不准确，因为分词后是多个单词，在排序就不是我们想要的结果了，通常的解决方案是，将一个 string field 建立两次索引，一个分词，用来进行搜索，一个不分词，用来进行排序

### 创建 mapping

```
1  PUT /website
2  {
3    "mappings": {
4      "article": {
5        "properties": {
6          "title": {
7            "type": "text",           // 使用分词
8            "fielddata": true,       // 只有设置为 true 才能 开始字
          字符串排序
9            "fields": {
10              "raw": {
11                "type": "keyword"    // type 是 keyword 不使用分词
12              }
13            },
14          },
15          "content": {
16            "type": "text"
17          },
18          "post_date": {
19            "type": "date"
20          },
21          "author": {
22            "type": "text"
23          }
24        }
25      }
26    }
27  }
28
```

### 排序搜索

## 使用分词字段搜索排序

```
1 GET /website/article/_search
2 {
3   "query": {
4     "match_all": {}
5   },
6   "sort": [
7     {
8       "title": {
9         "order": "asc"
10      }
11    }
12  ]
13 }
```

搜索结果 按照首字母升序排，并且搜索字段被分词

```
1 {
2   "took": 4,
3   "timed_out": false,
4   "_shards": {
5     "total": 5,
6     "successful": 5,
7     "skipped": 0,
8     "failed": 0
9   },
10  "hits": {
11    "total": 3,
12    "max_score": null,
13    "hits": [
14      {
15        "_index": "website",
16        "_type": "article",
17        "_id": "DXS1s2EBdXiUQYE3zPcm",
18        "_score": null,
19        "_source": {
20          "title": "first article",
21          "content": "this is my first article in elasticsearch",
22          "post_date": "2018-02-20",
23          "author": "daiwei"
24        },
25        "sort": [
26          "article"
```

```

27     ]
28 },
29 {
30     "_index": "website",
31     "_type": "article",
32     "_id": "DnS1s2EBdXiUQYE31PdG",
33     "_score": null,
34     "_source": {
35         "title": "second article",
36         "content": "this is my second article in elasticsearch",
37         "post_date": "2018-03-20",
38         "author": "zzq"
39     },
40     "sort": [
41         "article"
42     ]
43 },
44 {
45     "_index": "website",
46     "_type": "article",
47     "_id": "D3S1s2EBdXiUQYE33PfR",
48     "_score": null,
49     "_source": {
50         "title": "third article",
51         "content": "this is my third article in elasticsearch",
52         "post_date": "2018-04-20",
53         "author": "zzw"
54     },
55     "sort": [
56         "article"
57     ]
58 }
59 ]
60 }
61 }

```

## 使用 不分词字段进行搜索

```

1 GET /website/article/_search
2 {
3     "query": {
4         "match_all": {}
5     },
6     "sort": [

```

```

7   {
8     "title.raw": {
9       "order": "asc"
10    }
11  }
12 ]
13 }

```

搜索结果,按照搜索结构 首字母升序排列搜索字段未被分词

```

1  {
2    "took": 1,
3    "timed_out": false,
4    "_shards": {
5      "total": 5,
6      "successful": 5,
7      "skipped": 0,
8      "failed": 0
9    },
10   "hits": {
11     "total": 3,
12     "max_score": null,
13     "hits": [
14       {
15         "_index": "website",
16         "_type": "article",
17         "_id": "DXS1s2EBdXiUQYE3zPcm",
18         "_score": null,
19         "_source": {
20           "title": "first article",
21           "content": "this is my first article in elasticsearch",
22           "post_date": "2018-02-20",
23           "author": "daiwei"
24         },
25         "sort": [
26           "first article"
27         ]
28       },
29       {
30         "_index": "website",
31         "_type": "article",
32         "_id": "DnS1s2EBdXiUQYE31PdG",
33         "_score": null,
34         "_source": {
35           "title": "second article",

```

```
36     "content": "this is my second article in elasticsearch",
37     "post_date": "2018-03-20",
38     "author": "zzq"
39 },
40 "sort": [
41     "second article"
42 ]
43 },
44 {
45     "_index": "website",
46     "_type": "article",
47     "_id": "D3S1s2EBdXiUQYE33PfR",
48     "_score": null,
49     "_source": {
50         "title": "third article",
51         "content": "this is my third article in elasticsearch",
52         "post_date": "2018-04-20",
53         "author": "zzw"
54     },
55     "sort": [
56         "third article"
57     ]
58 }
59 ]
60 }
61 }
```