

Searching multiple fields



How the query works

Original query

```
GET /products/_search
{
  "query": {
    "multi_match": {
      "query": "vegetable",
      "fields": ["name", "description"]
    }
  }
}
```

translated into

```
GET /products/_search
{
  "query": {
    "match": {
      "name": "vegetable"
    }
  }
}
```

```
GET /products/_search
{
  "query": {
    "match": {
      "description": "vegetable"
    }
  }
}
```

Default relevance scoring

Original query

```
GET /products/_search
{
  "query": {
    "multi_match": {
      "query": "vegetable broth",
      "fields": ["name", "description"]
    }
  }
}
```

translated into

```
GET /products/_search
{
  "query": {
    "match": {
      "name": "vegetable broth"
    }
  }
}
```

```
GET /products/_search
{
  "query": {
    "match": {
      "description": "vegetable broth"
    }
  }
}
```

```
{
  "name": "Vegetable Broth", 12.698752
  "description": "Can be used to make vegetable soup." 8.510115
}
```

```
{
  "name": "Vegetable Soup (Spicy)", 10.623691
  "description": "Spicy stuff. Careful!" N/A
}
```

```
{
  "name": "Vegetable Spinach Lasagne", 9.9819723
  "description": "Spinach is very healthy." N/A
}
```



sorted by the highest relevance scores

Specifying a tie breaker

- By default, one field is used for calculating a document's relevance score
- We can “reward” documents where multiple fields match with the `tie_breaker` parameter
 - Each matching field affects the relevance score

Specifying a tie breaker

```
GET /products/_search
{
  "query": {
    "multi_match": {
      "query": "vegetable broth",
      "fields": ["name", "description"],
      "tie_breaker": 0.3
    }
  }
}
```



```
{
  "name": "Vegetable Broth", 12.698752
  "description": "Can be used to make vegetable soup." 8.510115
}
```

_score: 15.251787

Score calculation

```
12.698752
+ 2.5530345 (8.510115 * 0.3)
-----
= 15.251787
```

Specifying a tie breaker

```
GET /products/_search
{
  "query": {
    "multi_match": {
      "query": "vegetable broth",
      "fields": ["name", "description", "tags", "ingredients"],
      "tie_breaker": 0.3
    }
  }
}
```



```
{
  "_score": 18.7531707
  "name": "Vegetable Broth", 12.698752
  "description": "Can be used to make vegetable soup." 8.510115
  "tags": ["Broth", "Cooking"], 11.671279
  "ingredients": ["Water", "Carrot", "Celery", "Onion", "Salt"] N/A
}
```

Relevance score

```
12.698752
+ 2.5530345 (8.510115 * 0.3)
+ 3.5013837 (11.671279 * 0.3)
-----
= 18.7531707
```

Lecture summary

- The `multi_match` query performs full text searches on multiple fields
 - A document matches if at least one field is matched
- Individual fields can be relevance boosted by modifying the field name (^)
- Internally, Elasticsearch rewrites the query to simplify things for us 💙
- By default, the best matching field's relevance score is used for the document
 - Can be configured with the `type` parameter (see attached link)
- A tie breaker can be added to factor in every matching field's relevance score