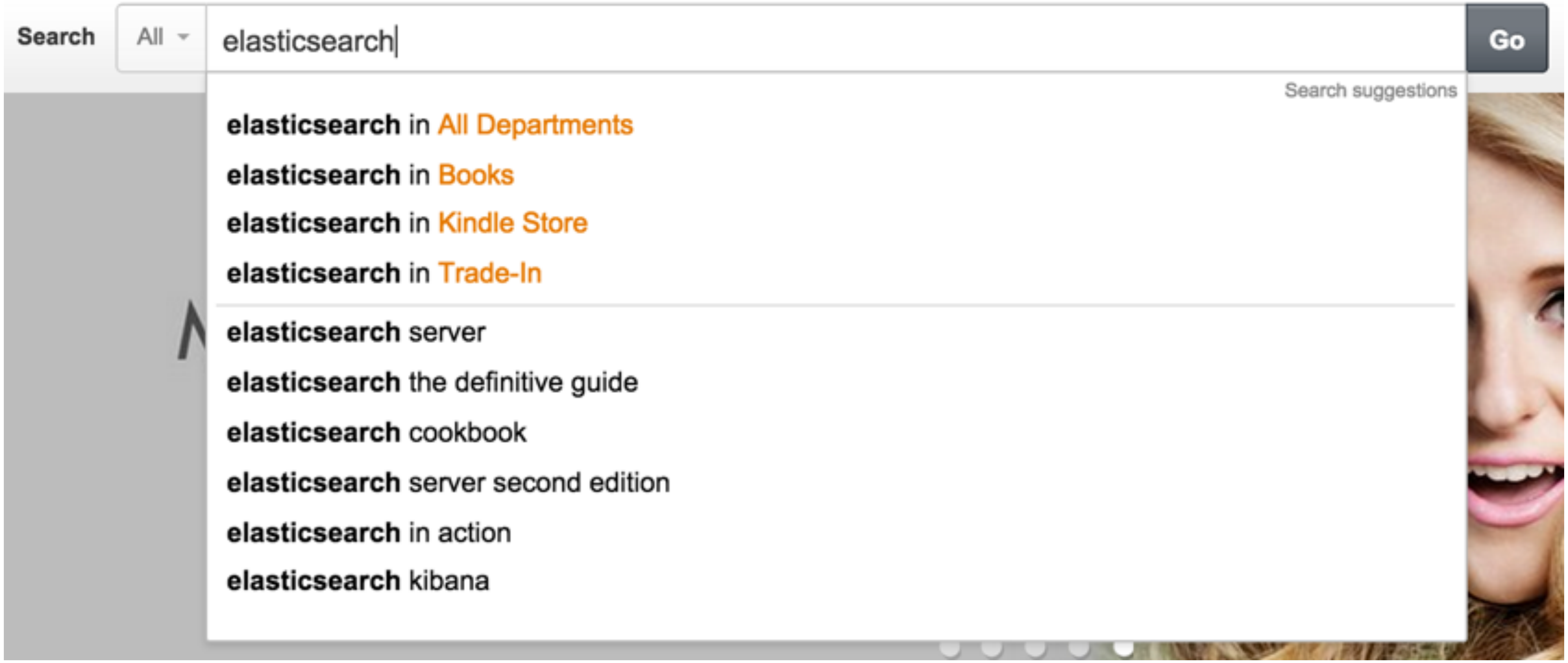
elasticsearch
for developer

N-gram

book_09_ngram.txt

Auto-suggestion with Elasticsearch

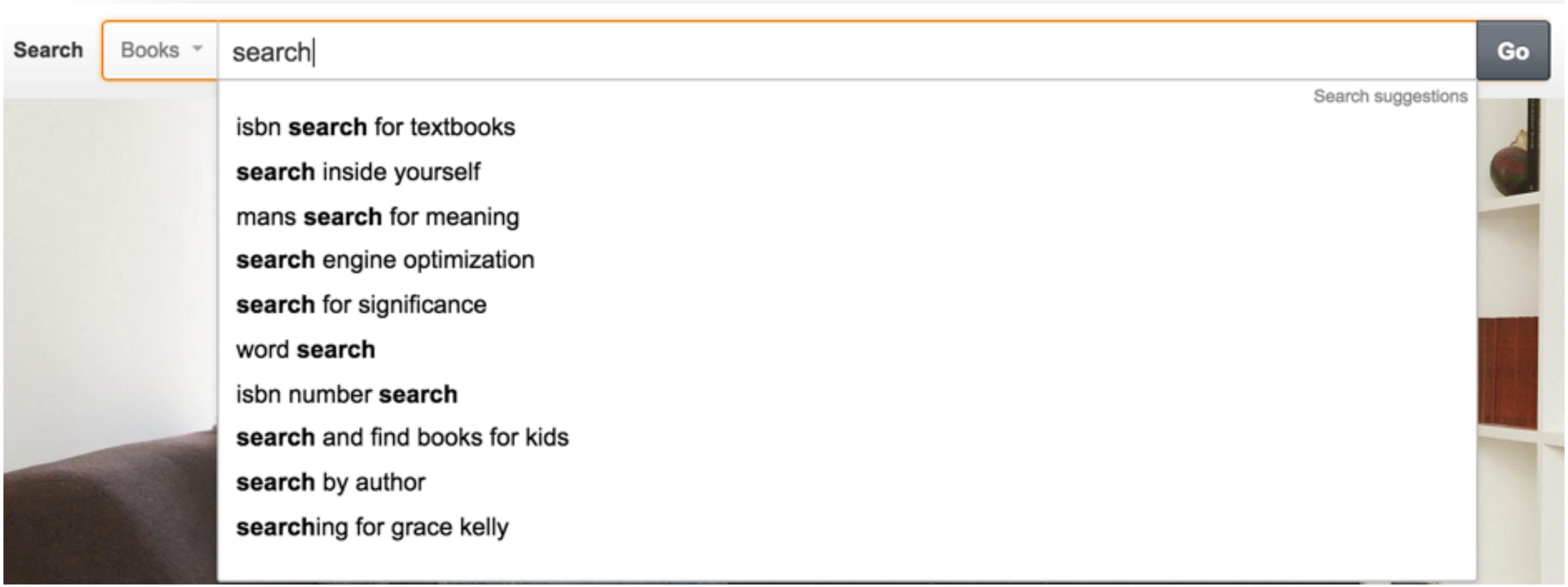


elasticsearch

elasticsearch in All Departments
elasticsearch in Books
elasticsearch in Kindle Store
elasticsearch in Trade-In

elasticsearch server
elasticsearch the definitive guide
elasticsearch cookbook
elasticsearch server second edition
elasticsearch in action
elasticsearch kibana

Auto-suggestion with Elasticsearch



search

isbn **search** for textbooks
search inside yourself
mans **search** for meaning
search engine optimization
search for significance
word **search**
isbn number **search**
search and find books for kids
search by author
searching for grace kelly

N-gram

- Partial matching
- Moving window on a word
- N = length of window

quick

Length	Name	Result
1	unigram	q,u,i,c,k
2	bigram	qu, ui, ic, ck
3	trigram	qui, uic, ick
4	four-gram	quic, uick
5	five-gram	quick

Plain N-gram

- Matching somewhere within a word

How to ?

1. Create index with N-gram
2. Test analyzer
3. Add data
4. Try query

Create index with N-gram

ทำการ custom filter และ analyzer

```
"filter": {
        "ngram": {
          "type": "ngram",
          "min_gram": 2,
          "max_gram": 15
        }
}
```

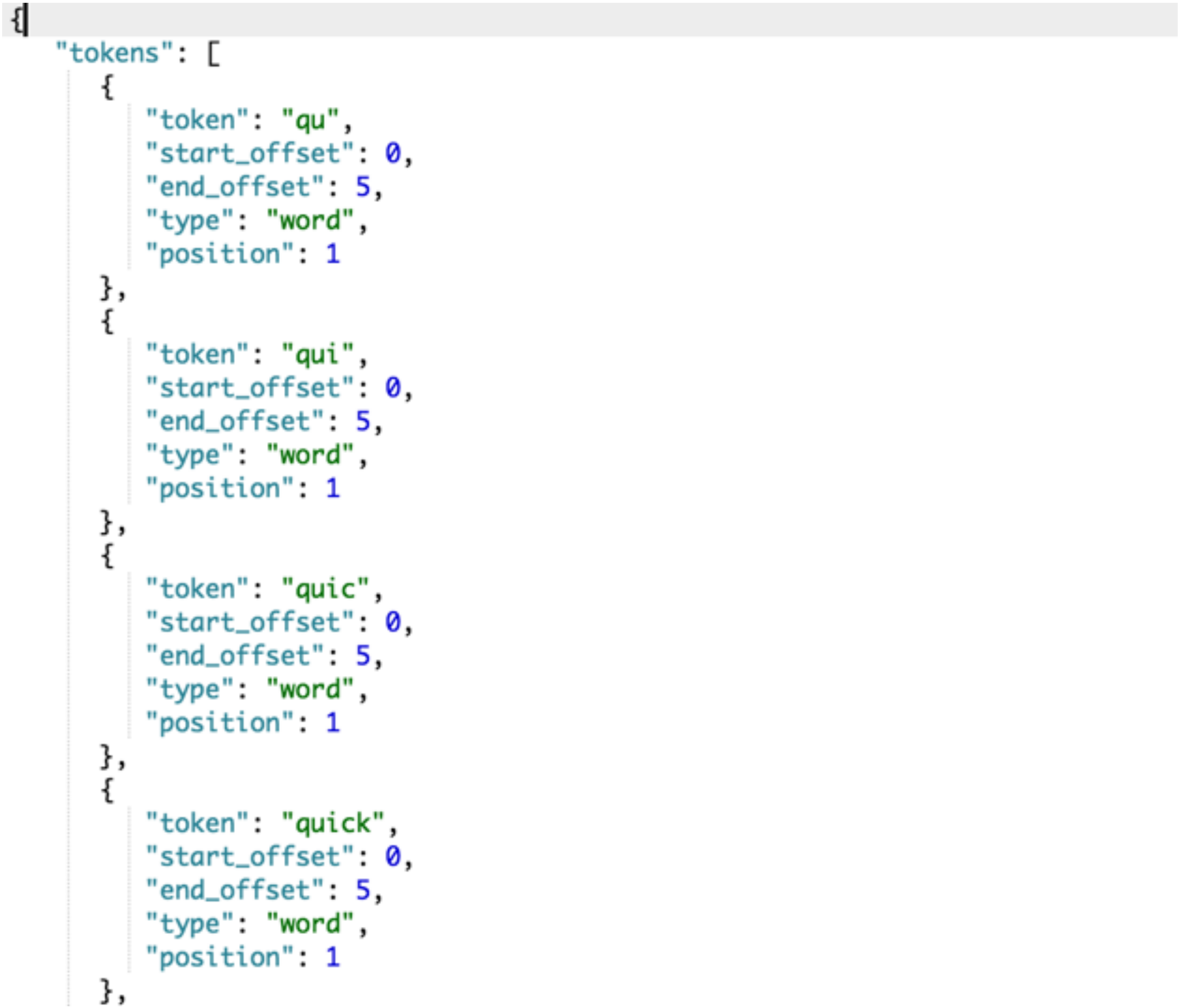
Create index with N-gram

ทำการ custom filter และ analyzer

```
"analyzer": {
    "autocomplete": {
      "type": "custom",
      "tokenizer": "standard",
      "filter": [
        "standard",
        "lowercase",
        "stop",
        "kstem",
        "ngram"
      ]}}
```

Test analyzer

GET /demo_ngram/_analyze?analyzer=autocomplete&text=quick



```
{
  "tokens": [
    {
      "token": "qu",
      "start_offset": 0,
      "end_offset": 5,
      "type": "word",
      "position": 1
    },
    {
      "token": "qui",
      "start_offset": 0,
      "end_offset": 5,
      "type": "word",
      "position": 1
    },
    {
      "token": "quic",
      "start_offset": 0,
      "end_offset": 5,
      "type": "word",
      "position": 1
    },
    {
      "token": "quick",
      "start_offset": 0,
      "end_offset": 5,
      "type": "word",
      "position": 1
    },
```

Update mapping with autocomplete

```
PUT /demo_ngram/_mapping/my_type
{
  "my_type": {
    "properties": {
      "name": {
        "type": "string",
        "analyzer": "autocomplete"
      }
    }
  }
}
```

Add documents

```
POST /demo_ngram/my_type/_bulk
{ "index": { "_id": 1}}
{ "name": "Elasticsearch book one"}
{ "index": { "_id": 2}}
{ "name": "Elasticsearch book two" }
{ "index": { "_id": 3}}
{ "name": "Elasticsearch book three" }
```

Try to search

```
GET /demo_ngram/_search
{
  "query": {
    "term": {
      "name": {
        "value": "las"
      }
    }
  }
}
```

Edge N-gram

(search-as-you-type)

Edge N-gram (search-as-you-type)

quick

q
qu
qui
quic
quick

How to ?

1. Create index with Edge N-gram
2. Test analyzer
3. Add data
4. Try query

Create index with Edge n-gram

ทำการ custom filter และ analyzer

```
"filter": {
        "autocomplete_filter": {
          "type": "edge_ngram",
          "min_gram": 1,
          "max_gram": 20
        }
}
```

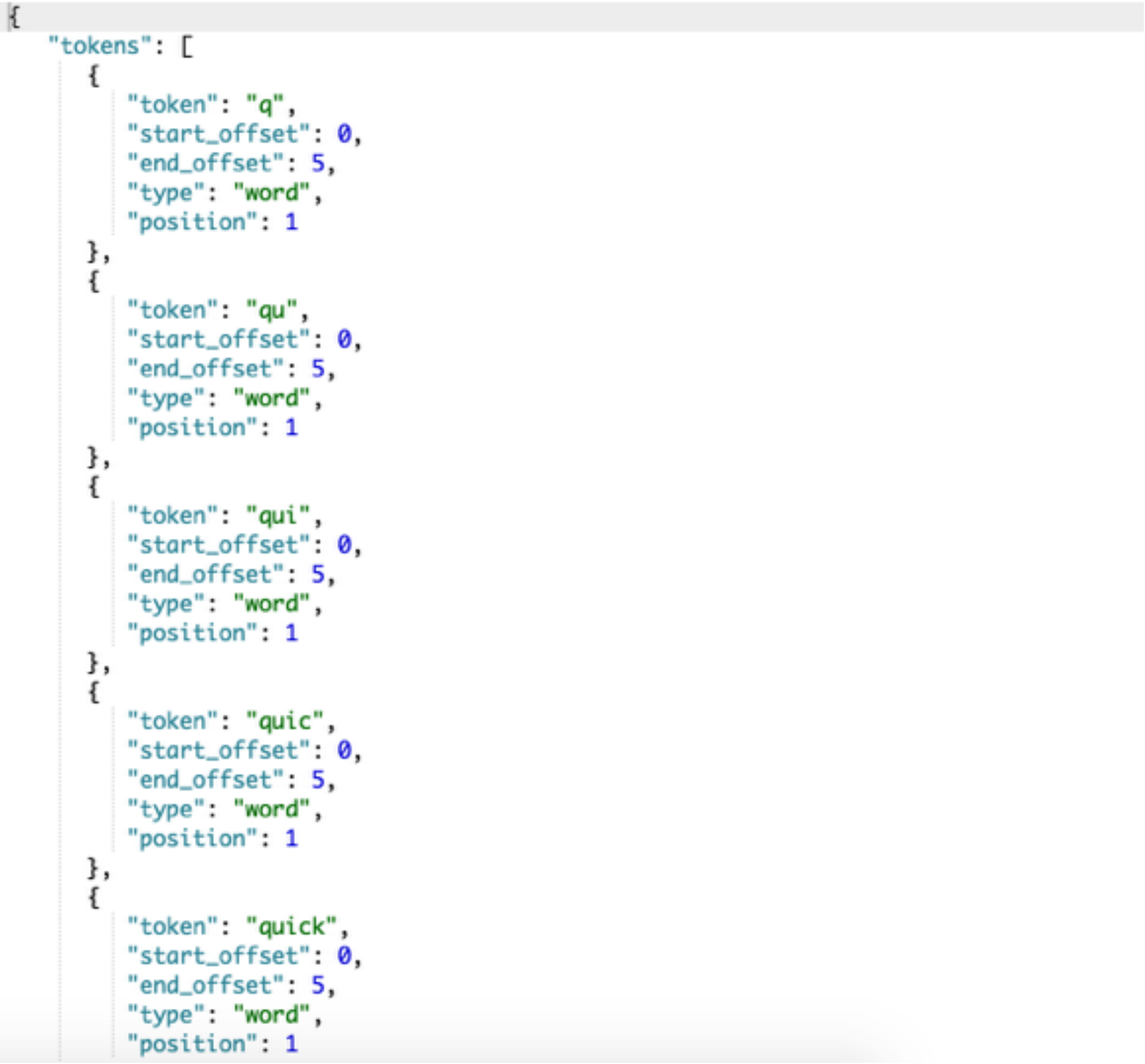
Create index with Edge n-gram

ทำการ custom filter และ analyzer

```
"analyzer": {
    "autocomplete": {
      "type": "custom",
      "tokenizer": "standard",
      "filter": [
        "lowercase",
        "autocomplete_filter"
      ]
    }}
```

Test analyzer

GET /demo_edge/_analyze?analyzer=autocomplete&text=quick



```
{
    "tokens": [
        {
            "token": "q",
            "start_offset": 0,
            "end_offset": 5,
            "type": "word",
            "position": 1
        },
        {
            "token": "qu",
            "start_offset": 0,
            "end_offset": 5,
            "type": "word",
            "position": 1
        },
        {
            "token": "qui",
            "start_offset": 0,
            "end_offset": 5,
            "type": "word",
            "position": 1
        },
        {
            "token": "quic",
            "start_offset": 0,
            "end_offset": 5,
            "type": "word",
            "position": 1
        },
        {
            "token": "quick",
            "start_offset": 0,
            "end_offset": 5,
            "type": "word",
            "position": 1
```

Update mapping with autocomplete

```
PUT /demo_edge/_mapping/my_type
{
  "my_type": {
    "properties": {
      "name": {
        "type": "string",
        "analyzer": "autocomplete"
      }
    }
  }
}
```

Add documents

```
POST /demo_edge/my_type/_bulk
{ "index": { "_id": 1}}
{ "name": "Elasticsearch book one"}
{ "index": { "_id": 2}}
{ "name": "Elasticsearch book two" }
{ "index": { "_id": 3}}
{ "name": "Elasticsearch book three" }
```

Try to search

```
GET /demo_edge/_search
{
  "query": {
    "term": {
      "name": {
        "value": "boo"
      }
    }
  }
}
```