How Does Full-Text Search Work under the Hood

Philipp Krenn @xeraa

elastic

Who uses a Database?

Who uses Search?


elastic

elasticsearch

elastic

Developer 

Store

Apache Lucene
Elasticsearch

elastic

Example

These are not the droids you are looking for.

`html_strip` Char Filter

These are not the droids you are looking for.

`standard` Tokenizer

These are not the droids you are looking for

`lowercase` Token Filter

these are not the droids you are looking for

`stop` Token Filter

droids you looking

elastic

`snowball` Token Filter

droid you look

Setup

elastic cloud

https://cloud.elastic.co

Email

Password

Log in

Forgot password?

Don't have an account? Sign up now.

elastic

Have questions? Email us at support@elastic.co

docker

elastic

Docker Compose

```
---
version: '2'
services:
  kibana:
    image: docker.elastic.co/kibana/kibana:$ELASTIC_VERSION
    links:
      - elasticsearch
    ports:
      - 5601:5601

  elasticsearch:
    image: docker.elastic.co/elasticsearch/elasticsearch:$ELASTIC_VERSION
    volumes:
      - esdata1:/usr/share/elasticsearch/data
    ports:
      - 9200:9200

volumes:
  esdata1:
    driver: local
```

Analyze

elastic

```
GET /_analyze
{
  "analyzer": "english",
  "text": "These are not the droids you are looking for."
}
```

```
{
  "tokens": [
    {
      "token": "droid",
      "start_offset": 18,
      "end_offset": 24,
      "type": "<ALPHANUM>",
      "position": 4
    },
    {
      "token": "you",
      "start_offset": 25,
      "end_offset": 28,
      "type": "<ALPHANUM>",
      "position": 5
    },
    ...
  ]
}
```

```
GET /_analyze
{
  "char_filter": [
    "html_strip"
  ],
  "tokenizer": "standard",
  "filter": [
    "lowercase",
    "stop",
    "snowball"
  ],
  "text": "These are <em>not</em> the droids you are looking for."
}
```

```
{
  "tokens": [
    {
      "token": "droid",
      "start_offset": 27,
      "end_offset": 33,
      "type": "<ALPHANUM>",
      "position": 4
    },
    {
      "token": "you",
      "start_offset": 34,
      "end_offset": 37,
      "type": "<ALPHANUM>",
      "position": 5
    },
    ...
  ]
}
```

Stop Words

a an and are as at be but by for if in into is it no not of on or such that the their then there these they this to was will with

https://github.com/apache/lucene-solr/blob/master/lucene/analysis/common/src/java/org/apache/lucene/analysis/en/EnglishAnalyzer.java#L44-L50

Always Use Stop Words?

To be, or not to be.

Lithuanian

Tai ne tie droidai, kurių ieškote.

elastic

Lithuanian

ne tie droid kur ieš̌k

elastic

Languages

Arabic, Armenian, Basque, Brazilian, Bulgarian, Catalan, CJK, Czech, Danish, Dutch, English, Finnish, French, Galician, German, Greek, Hindi, Hungarian, Indonesian, Irish, Italian, Latvian, Lithuanian, Norwegian, Persian, Portuguese, Romanian, Russian, Sorani, Spanish, Swedish, Turkish, Thai

Language Rules

English: `Philipp's` → `philipp`

French: `l'église` → `eglis`

German: `äußerst` → `ausserst`

More Language Plugins

Core: ICU (Asian languages), Kuromoji (advanced Japanese), Phonetic, SmartCN, Stempel (Polish), Ukrainian

Community: Hebrew, Vietnamese, Network Address Analysis, String2Integer,...

Lithuanian with the English Analyzer

tai ne tie droidai kurių ieškot

elastic

Lithuanian Stop Words

https://github.com/apache/lucene-solr/blob/master/lucene/analysis/common/src/resources/org/apache/lucene/analysis/lt/stopwords.txt

Detect Languages

https://github.com/spinscale/elasticsearch-ingest-langdetect

```
PUT _ingest/pipeline/langdetect-pipeline
{
  "description": "A pipeline to detect languages",
  "processors": [
    {
      "langdetect" : {
        "field" : "quote",
        "target_field" : "language"
      }
    }
  ]
}
```

```
POST _ingest/pipeline/langdetect-pipeline/_simulate
{
  "docs": [
    {
      "_source": {
        "quote": "Tai ne tie droidai, kurių ieškote."
      }
    }
  ]
}
```

```
{
  "docs": [
    {
      "doc": {
        "_index": "_index",
        "_type": "_type",
        "_id": "_id",
        "_source": {
          "language": "lt",
          "quote": "Tai ne tie droidai, kurių ieškote."
        },
        "_ingest": {
          "timestamp": "2018-10-26T00:06:42.320613Z"
        }
      }
    }
  ]
}
```

Phonetic

```
GET /_analyze
{
  "tokenizer": "standard",
  "filter": [
    {
      "type": "phonetic",
      "encoder": "beider_morse",
      "languageset": "any"
    }
  ],
  "text": "These are not the droids you are looking for."
}
```

Phonetic

... drDts drits drots iou ari ori loknk...

Another Example

Obi-Wan never told you what happened to your father.

Another Example

obi wan never told you what happen your father

Another Example

No. I am your father.

Another Example

i am your father

Inverted Index

	ID 1	ID 2	ID 3
am	0	0	1[2]
droid	1[4]	0	0
father	0	1[9]	1[4]
happen	0	1[6]	0
i	0	0	1[1]
look	1[7]	0	0
never	0	1[2]	0
obi	0	1[0]	0
told	0	1[3]	0
wan	0	1[1]	0
what	0	1[5]	0
you	1[5]	1[4]	0
your	0	1[8]	1[3]

To / The Index

```
PUT /starwars
{
  "settings": {
    "number_of_shards": 1,
    "analysis": {
      "filter": {
        "my_synonym_filter": {
          "type": "synonym",
          "synonyms": [
            "father,dad",
            "droid => droid,machine"
          ]
        }
      },
```

```
    "analyzer": {
      "my_analyzer": {
        "char_filter": [
          "html_strip"
        ],
        "tokenizer": "standard",
        "filter": [
          "lowercase",
          "stop",
          "snowball",
          "my_synonym_filter"
        ]
      }
    }
  }
},
```

```
"mappings": {
  "_doc": {
    "properties": {
      "quote": {
        "type": "text",
        "analyzer": "my_analyzer"
      }
    }
  }
}
```

Synonyms

Index `synonym` **or query time** `synonym_graph`

```
GET /starwars/_mapping

GET /starwars/_settings
```

```
PUT /starwars/_doc/1
{
  "quote": "These are <em>not</em> the droids you are looking for."
}
PUT /starwars/_doc/2
{
  "quote": "Obi-Wan never told you what happened to your father."
}
PUT /starwars/_doc/3
{
  "quote": "<b>No</b>. I am your father."
}
```

```
GET /starwars/_doc/1

GET /starwars/_doc/1/_source
```

Search

```
POST /starwars/_search
{
  "query": {
    "match_all": { }
  }
}
```

```
{
  "took": 1,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 3,
    "max_score": 1,
    "hits": [
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "2",
        "_score": 1,
        "_source": {
          "quote": "Obi-Wan never told you what happened to your father."
        }
      },
      ...
```

```
POST /starwars/_search
{
  "query": {
    "match": {
      "quote": "droid"
    }
  }
}
```

```
{
  "took": 2,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 1,
    "max_score": 0.39556286,
    "hits": [
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "1",
        "_score": 0.39556286,
        "_source": {
          "quote": "These are <em>not</em> the droids you are looking for."
        }
      }
    ]
  }
}
```

```
POST /starwars/_search
{
  "query": {
    "match": {
      "quote": "dad"
    }
  }
}
```

```
...
  "hits": {
    "total": 2,
    "max_score": 0.41913947,
    "hits": [
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "3",
        "_score": 0.41913947,
        "_source": {
          "quote": "<b>No</b>. I am your father."
        }
      },
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "2",
        "_score": 0.39291072,
        "_source": {
          "quote": "Obi-Wan never told you what happened to your father."
        }
      }
    ]
  }
}
```

```
POST /starwars/_doc/0/_explain
{
  "query": {
    "match": {
      "quote": "dad"
    }
  }
}
```

```
{
  "_index": "starwars",
  "_type": "_doc",
  "_id": "0",
  "matched": false
}
```

```
POST /starwars/_doc/1/_explain
{
  "query": {
    "match": {
      "quote": "dad"
    }
  }
}
```

```
{
  "_index": "starwars",
  "_type": "_doc",
  "_id": "1",
  "matched": false,
  "explanation": {
    "value": 0,
    "description": "no matching term",
    "details": []
  }
}
```

```
POST /starwars/_doc/2/_explain
{
  "query": {
    "match": {
      "quote": "dad"
    }
  }
}
```

```
{
  "_index": "starwars",
  "_type": "_doc",
  "_id": "2",
  "matched": true,
  "explanation": {
    ...
```

```
POST /starwars/_search
{
  "query": {
    "match": {
      "quote": "machine"
    }
  }
}
```

```
{
  "took": 2,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": 1,
    "max_score": 1.2499592,
    "hits": [
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "1",
        "_score": 1.2499592,
        "_source": {
          "quote": "These are <em>not</em> the droids you are looking for."
        }
      }
    ]
  }
}
```

```
POST /starwars/_search
{
  "query": {
    "match_phrase": {
      "quote": "I am your father"
    }
  }
}
```

```
{
  "took": 3,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 1,
    "max_score": 1.5665855,
    "hits": [
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "3",
        "_score": 1.5665855,
        "_source": {
          "quote": "<b>No</b>. I am your father."
        }
      }
    ]
  }
}
```

```
POST /starwars/_search
{
  "query": {
    "match_phrase": {
      "quote": {
        "query": "I am father",
        "slop": 1
      }
    }
  }
}
```

```
{
  "took": 16,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 1,
    "max_score": 0.8327639,
    "hits": [
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "3",
        "_score": 0.8327639,
        "_source": {
          "quote": "<b>No</b>. I am your father."
        }
      }
    ]
  }
}
```

```
POST /starwars/_search
{
  "query": {
    "match_phrase": {
      "quote": {
        "query": "I am not your father",
        "slop": 1
      }
    }
  }
}
```

```
{
  "took": 5,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 1,
    "max_score": 1.0409548,
    "hits": [
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "3",
        "_score": 1.0409548,
        "_source": {
          "quote": "<b>No</b>. I am your father."
        }
      }
    ]
  }
}
```

```
POST /starwars/_search
{
  "query": {
    "match": {
      "quote": {
        "query": "van",
        "fuzziness": "AUTO"
      }
    }
  }
}
```

```
{
  "took": 14,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 1,
    "max_score": 0.18155496,
    "hits": [
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "2",
        "_score": 0.18155496,
        "_source": {
          "quote": "Obi-Wan never told you what happened to your father."
        }
      }
    ]
  }
}
```

```
POST /starwars/_search
{
  "query": {
    "match": {
      "quote": {
        "query": "ovi-van",
        "fuzziness": 1
      }
    }
  }
}
```

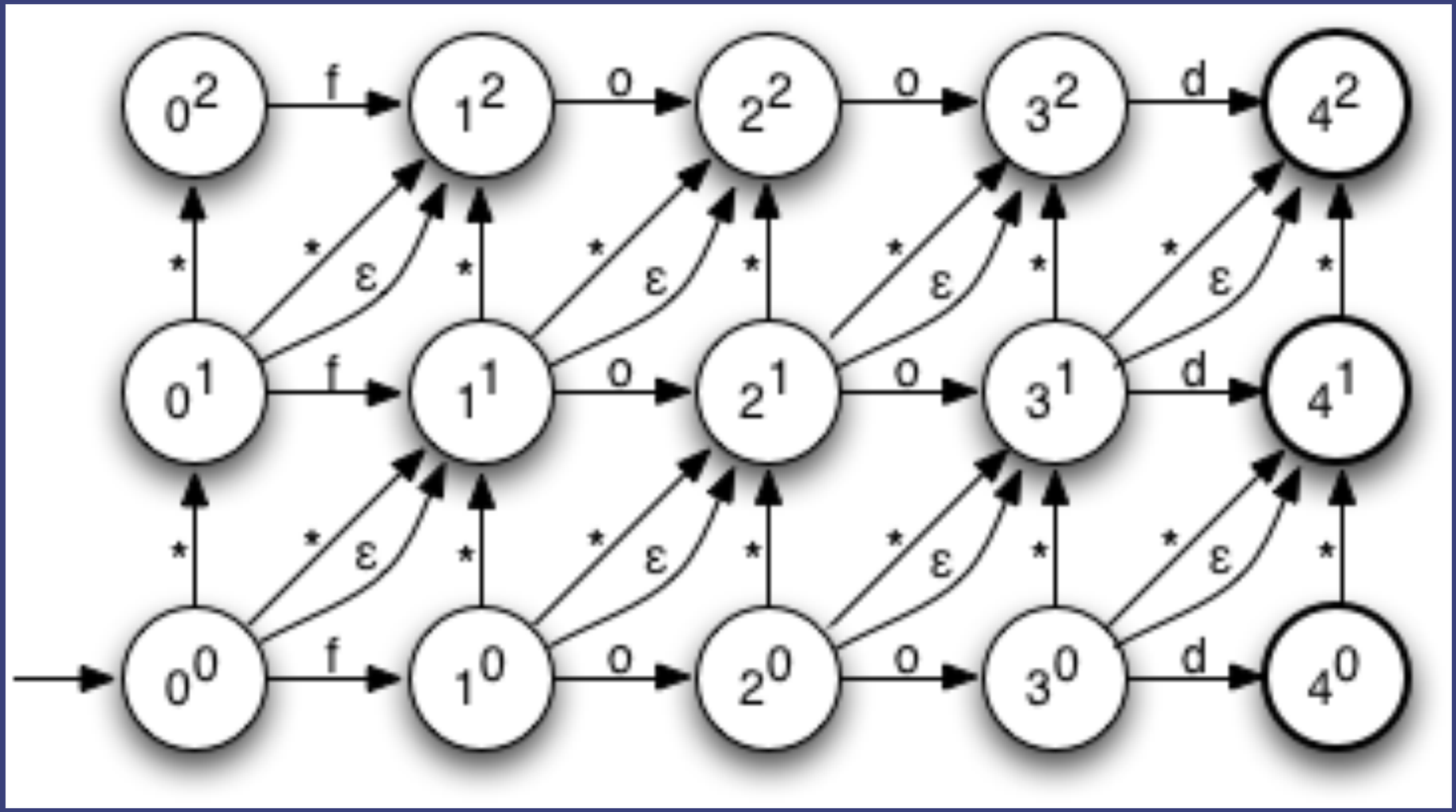
```
{
  "took": 109,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 1,
    "max_score": 0.3798467,
    "hits": [
      {
        "_index": "starwars",
        "_type": "_doc",
        "_id": "2",
        "_score": 0.3798467,
        "_source": {
          "quote": "Obi-Wan never told you what happened to your father."
        }
      }
    ]
  }
}
```

FuzzyQuery History

http://blog.mikemccandless.com/2011/03/lucenes-fuzzyquery-is-100-times-faster.html

Before: Brute force

Now: Levenshtein Automaton



http://blog.notdot.net/2010/07/Damn-Cool-Algorithms-Levenshtein-Automata

```
SELECT *
  FROM starwars
 WHERE quote LIKE "?an" OR
       quote LIKE "V?n" OR
       quote LIKE "Va?"
```

Score

Term Frequency / Inverse Document Frequency (TF/IDF)

Search one term

elastic

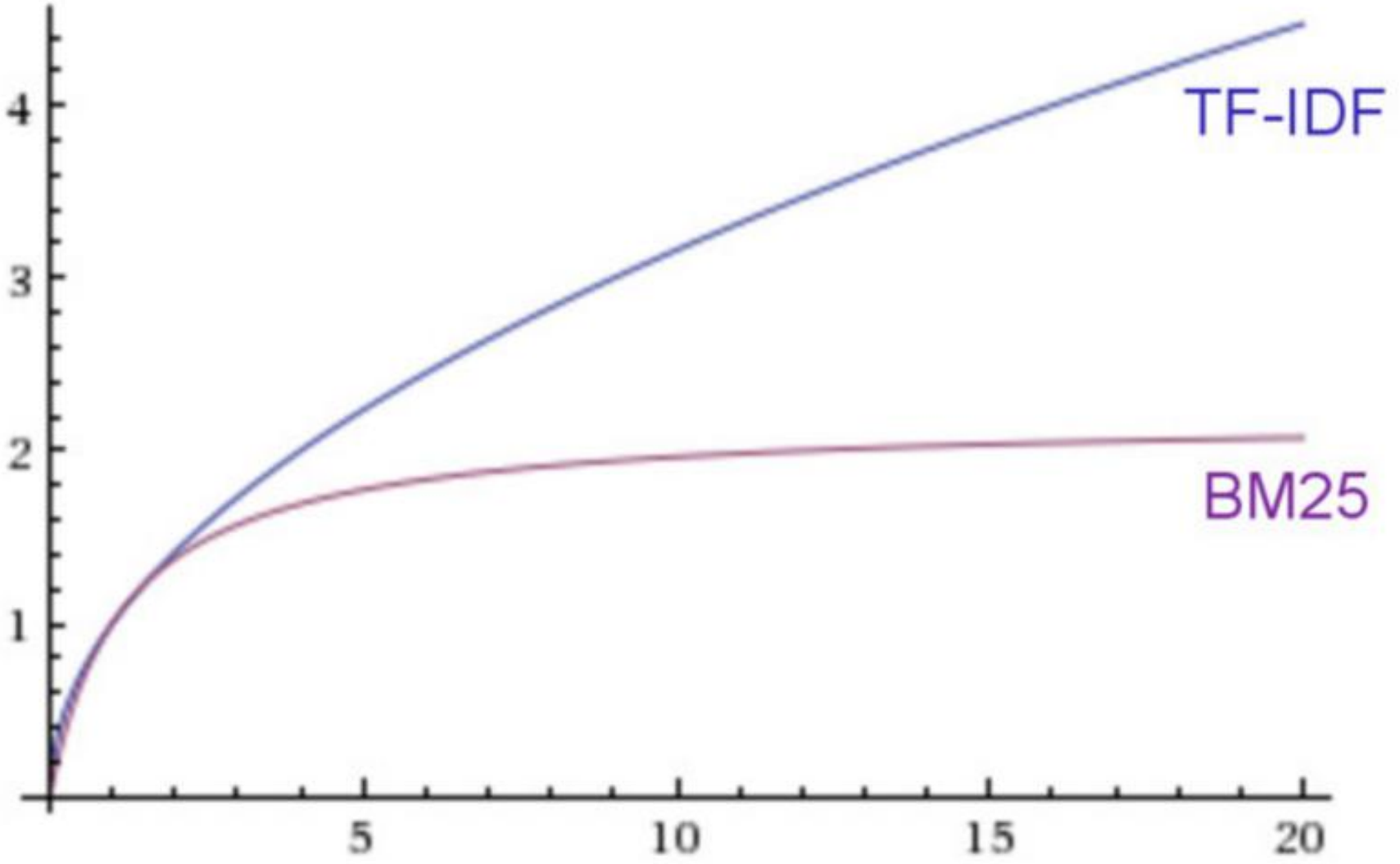
BM25

Default in Elasticsearch 5.0

https://speakerdeck.com/elastic/improved-text-scoring-with-bm25

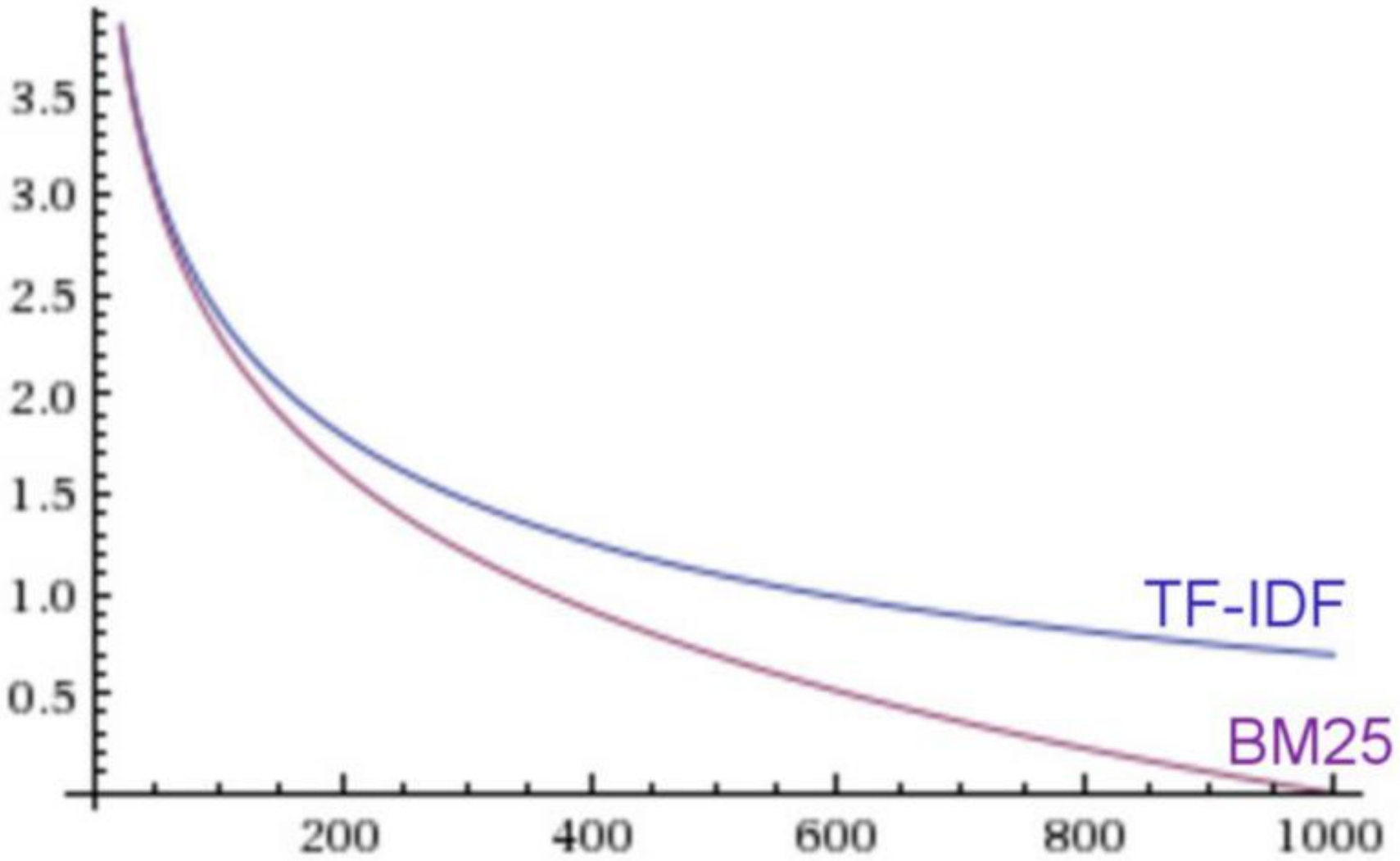
Term Frequency

$$tf(t\ in\ d) = \sqrt{frequency}$$



Inverse Document Frequency

$$idf(t) = 1 + log(\frac{numDocs}{docFreq + 1})$$



Field-Length Norm

$$norm(d) = \frac{1}{\sqrt{numTerms}}$$

```
POST /starwars/_search?explain=true
{
  "query": {
    "match": {
      "quote": "father"
    }
  }
}
```

```
...
"_explanation": {
  "value": 0.41913947,
  "description": "weight(Synonym(quote:dad quote:father) in 0) [PerFieldSimilarity], result of:",
  "details": [
    {
      "value": 0.41913947,
      "description": "score(doc=0,freq=2.0 = termFreq=2.0\n), product of:",
      "details": [
        {
          "value": 0.2876821,
          "description": "idf(docFreq=1, docCount=1)",
          "details": []
        },
        {
          "value": 1.4569536,
          "description": "tfNorm, computed from:",
          "details": [
            {
              "value": 2,
              "description": "termFreq=2.0",
              "details": []
            },
            ...
```

Score

0.41913947: i am your father

0.39291072: obi wan never told you what happen your father

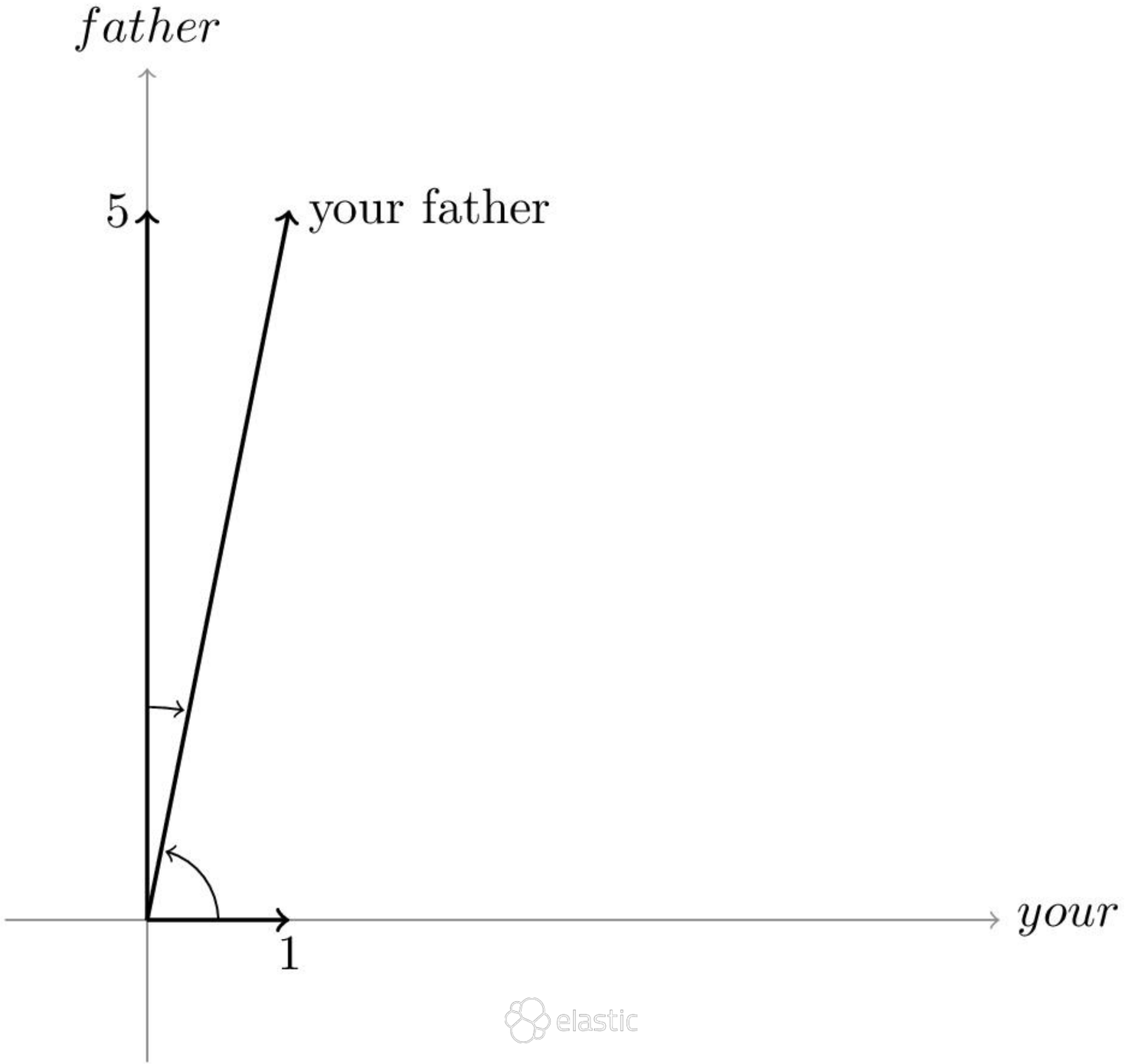
Vector Space Model

Search multiple terms

elastic

Search your father

elastic


father
5
your father
1
your

Coordination Factor

Reward multiple terms

elastic

Search for 3 terms

1 term: $X \cdot \frac{1}{3}$

2 terms: $(X + Y) \cdot \frac{2}{3}$

3 terms: $(X + Y + Z) \cdot \frac{3}{3}$

Practical Scoring Function

Putting it all together

elastic

```
score(q,d)  =
              queryNorm(q)
            · coord(q,d)
            · Σ (
                tf(t in d)
              · idf(t)²
              · t.getBoost()
              · norm(t,d)
              ) (t in q)
```

Function Score

Script, weight, random, field value, decay (geo or date)

```
POST /starwars/_search
{
  "query": {
    "function_score": {
      "query": {
        "match": {
          "quote": "father"
        }
      },
      "random_score": {}
    }
  }
}
```

Compare Scores

"100% perfect" vs a "50%" match

Don't do this. Seriously. Stop trying to think about your problem this way, it's not going to end well.

— https://wiki.apache.org/lucene-java/ScoresAsPercentages

```
GET /starwars/_analyze
{
  "analyzer" : "my_analyzer",
  "text": "These are my father's machines."
}
```

```
{ "tokens": [
    {
      "token": "my",
      "start_offset": 10,
      "end_offset": 12,
      "type": "<ALPHANUM>",
      "position": 2
    },
    {
      "token": "father",
      "start_offset": 13,
      "end_offset": 21,
      "type": "<ALPHANUM>",
      "position": 3
    },
    {
      "token": "dad",
      "start_offset": 13,
      "end_offset": 21,
      "type": "SYNONYM",
      "position": 3
    },
    {
      "token": "machin",
      "start_offset": 22,
      "end_offset": 30,
      "type": "<ALPHANUM>",
      "position": 4
    }
] }
```

```
PUT /starwars/_doc/4
{
  "quote": "These are my father's machines."
}
```

```
POST /starwars/_search
{
  "query": {
    "match": {
      "quote": "my father machine"
    }
  }
}
```

```
"hits": {
  "total": 4,
  "max_score": 2.92523,
  "hits": [
    {
      "_index": "starwars",
      "_type": "_doc",
      "_id": "4",
      "_score": 2.92523,
      "_source": {
        "quote": "These are my father's machines."
      }
    },
    {
      "_index": "starwars",
      "_type": "_doc",
      "_id": "1",
      "_score": 0.8617505,
      "_source": {
        "quote": "These are <em>not</em> the droids you are looking for."
      }
    },
    ...
```

2.92523 == 100%

```
DELETE /starwars/_doc/4

POST /starwars/_search
{
  "query": {
    "match": {
      "quote": "my father machine"
    }
  }
}
```

```
"hits": {
  "total": 3,
  "max_score": 1.2499592,
  "hits": [
    {
      "_index": "starwars",
      "_type": "_doc",
      "_id": "1",
      "_score": 1.2499592,
      "_source": {
        "quote": "These are <em>not</em> the droids you are looking for."
      }
    },
    ...
```

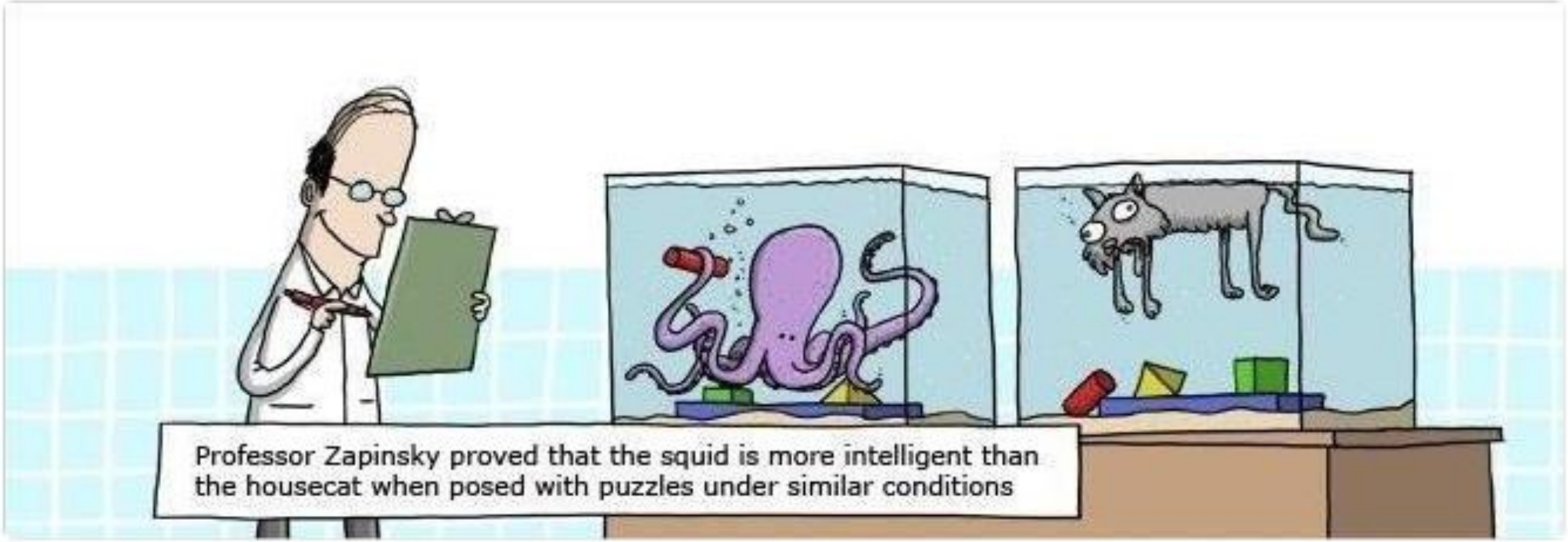
1.2499592 == 43%
or 100%?

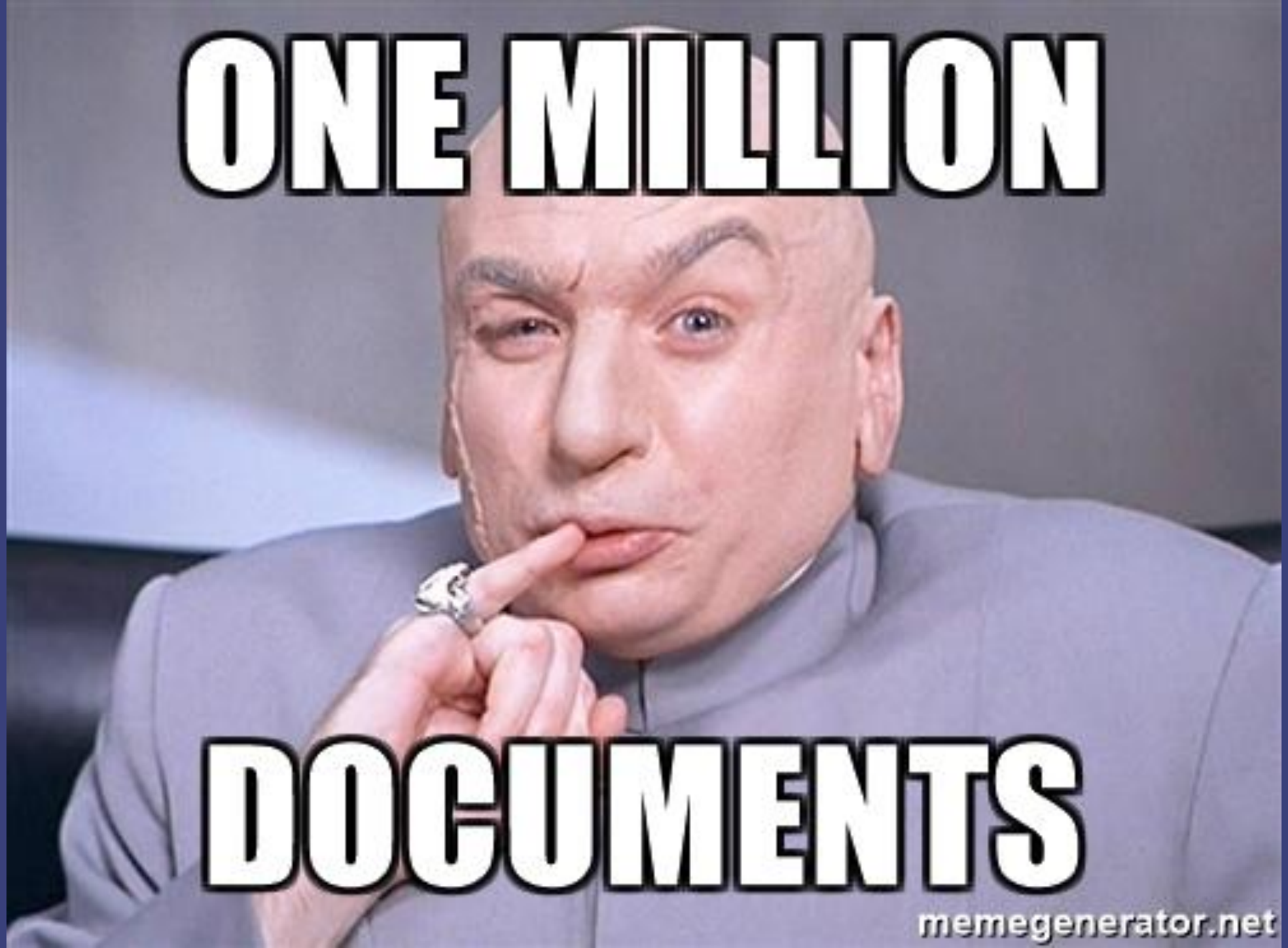
```
PUT /starwars/_doc/4
{
  "quote": "These droids are my father's father's machines."
}

POST /starwars/_search
{
  "query": {
    "match": {
      "quote": "my father machine"
    }
  }
}
```

```
"hits": {
  "total": 4,
  "max_score": 3.0068164,
  "hits": [
    {
      "_index": "starwars",
      "_type": "_doc",
      "_id": "4",
      "_score": 3.0068164,
      "_source": {
        "quote": "These droids are my father's father's machines."
      }
    },
    {
      "_index": "starwars",
      "_type": "_doc",
      "_id": "1",
      "_score": 0.89701396,
      "_source": {
        "quote": "These are <em>not</em> the droids you are looking for."
      }
    },
    ...
```

3.0068164 == 103%?

Performance


Professor Zapinsky proved that the squid is more intelligent than the housecat when posed with puzzles under similar conditions


ONE MILLION
DOCUMENTS
memegenerator.net

Conclusion

Indexing

Formatting

Tokenize

Lowercase, Stop Words, Stemming

Synonyms

Scoring

Term Frequency

Inverse Document Frequency

Field-Length Norm

Vector Space Model

app search

https://www.elastic.co/blog/elastic-app-search-beta-released

Thank You

Questions & Stickers

Philipp Krenn **@xeraa**