# Search

Tamas Piros

# Search in NoSQL

- For faster performance requires an index
- Text search features
  - Full text match
  - Wildcard search
  - Case (in)sensitive search
  - Diacritic (in)sensitive search
  - Weights
  - Tokenisation

# Text Search 101

- Term Frequency (TF)
  - How often a term appears in a document
  - More often = higher weight (important)
  - Mathematically: `tf(term in doc) = √frequency`
- Inverse Document Frequency (IDF)
  - How often does a term appear in all documents
  - More often = lower weight (less important)
  - Mathematically: `norm(d) = 1 / √number of terms`
- Final score: `log(TF) * (IDF)`

# Tokenisation and Weights

- Tokenisation is the process of producing (word) tokens
- Importance of a field, relative to other indexed fields
  - Used in final score calculation for search results
  - Different databases use different default values (0 or 1)

# Wildcard search

- Allows for the matching of patterns
  - ca*: cat, car, can

# Case and Diacritic (in)sensitive

- Case (in)sensitive
  - Different results for ‘Apple’ and ‘apple’
  - Exact vs non-exact match
- Diacritic (in)sensitive
  - Language specific search
  - e vs è vs é

# Stemming

- Some databases supports stemming
  - Find the root of the word, search for that as well
  - Different terms in languages stem to different roots
    - chat in French vs chat in English
  - Decompounding stemming
    - Especially useful for German language
    - “kopfkino”