MOVIE RATINGS AND REVIEWS

PRI 2021/2022

Caio Nogueira
Carlos Lousada
Tomás Mendes

Milestone 1

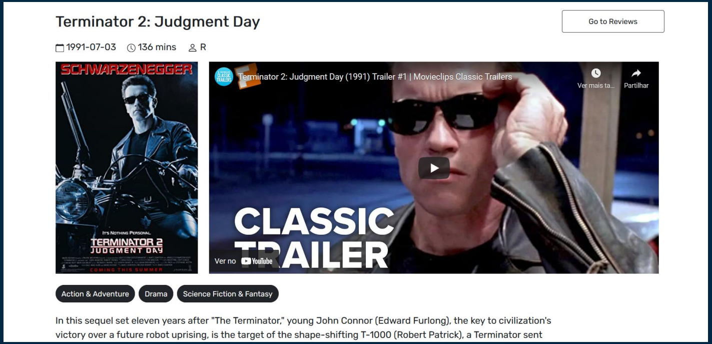
- Data collected in Kaggle scrapped from IMDB and RottenTomatoes;
- Dataset contains over 9,000 movies with up to 20 reviews per movie (after preprocessing);
- Statistical analysis was performed to better understand the dataset

Milestone 2

- Introduction to Solr because it suits our problem better;
- Usage of a single document with all information to be retrieved
- Indexation of relevant fields for information retrieval
- Evaluation of 3 different systems - Schemaless, Schema-only and Schema+Weights

Improvements Introduced

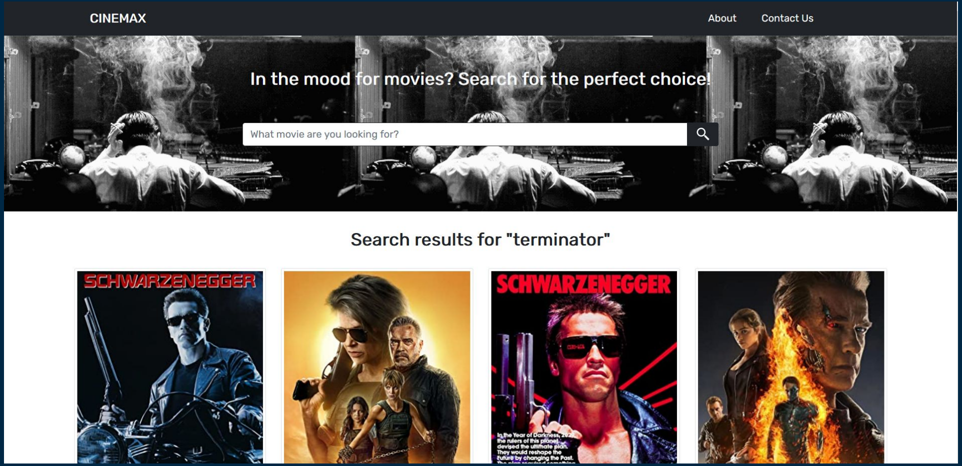
- Synonyms for query expansion
- OpenNLP to perform multiple tasks
  - Named-Entity Recognition
  - Chunker
  - Parts of Speech
- Learning to Rank
- Graphical Interface


Terminator 2: Judgment Day

1991-07-03 136 mins R

In this sequel set eleven years after "The Terminator," young John Connor (Edward Furlong), the key to civilization's victory over a future robot uprising, is the target of the shape-shifting T-1000 (Robert Patrick), a Terminator sent

Go to Reviews
Terminator 2: Judgment Day (1991) Trailer #1 | Movieclips Classic Trailers
CLASSIC TRAILER
Action & Adventure
Drama
Science Fiction & Fantasy


In the mood for movies? Search for the perfect choice!

Search results for "terminator"

Indexing Process – OpenNLP Field types

Field Type	Filter
standard_text	ASCIIFoldingFilterFactory LowerCaseFilterFactory SynonymGraphFilterFactory EnglishPossessiveFilterFactory* EnglishMinimalStemFilterFactory*
daterange	DateRangeField
nlp_text*	All Filters Used in standard_text OpenNLPPOSFilterFactory OpenNLPChunkerFilterFactory

Synonyms

```
{
filter
"class": "solr.SynonymGraphFilterFactory",
"synonyms": "synonyms.txt",
"expand": "false",
"ignoreCase": "true"
}
```

thief => stealer
40s => forty, 40, XL
blender => liquidizer, liquidiser
monotony => humdrum, sameness
(...)

Named-Entity Recognition

- **People:** people's names - identifies characters, actors,... (e.g., "Barack Obama", "James Bond");
- **Organizations:** Organizations' names - identifies organizations relevant in the movies (e.g., "FBI", "Army");
- **Dates:** Important dates in the movies - identifies weekdays, months or holidays (e.g., "1940s", "Sunday").

Learning to Rank (feature extraction)

Name	Params
maximize_votes	q: {!func}scale(total_votes, 0, 1)
maximize_rating	q: {!func}scale(total_votes, 0, 1)
review_bm25	q: {!dismax qf=’review_content’}${text}
description_bm25	q: {!dismax qf=’movie_info’}${text}
original_score	{ }

Learning to Rank (training SVM Rank model)

- **SVM** variant adapted to Information Retrieval problem.

- Combines documents in pairs (comparable) => **Pairwise Transformation.**

- Weights are given by the model's coefficients (hyperplane coordinates).

Results comparison

To compare results, we used 2 different systems:

- Using a schema and applying **weights;**
- Using an improved schema, LTR and applying weights.

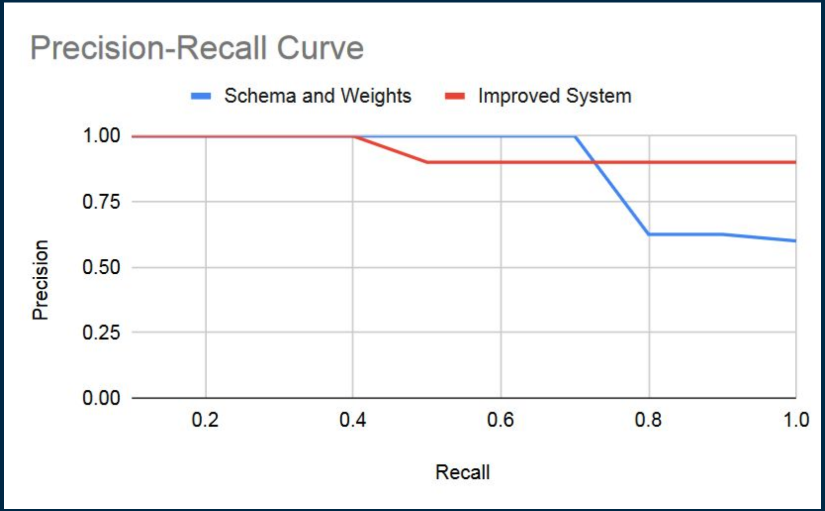
Information Needs - Comparison

IN2 - Movies about slavery

Query (q): slave

Query filters (qf): original_title^1, movie_info^5, review_content^3

	Milestone 2 (Boosted)	Milestone 3 (Improved)
P@10	0.60	0.90
AvP	0.870833	0.906041



Information Needs – Comparison

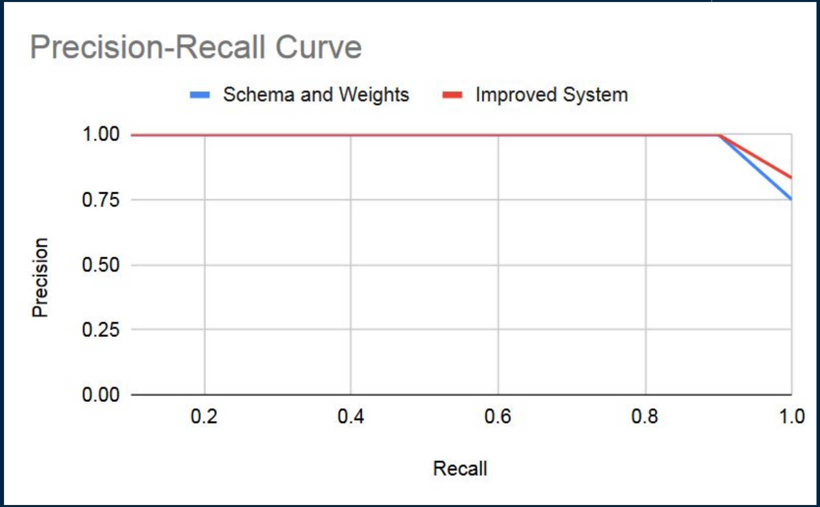
IN4 - Movies about true crime stories

Query (q): true crime story

Query fields (qf): movie_info^3, review_content^5

Phrase Slop (ps): 3

	Milestone 2 (Boosted)	Milestone 3 (Improved)
P@10	0.60	0.50
AvP	0.96	0.97



Information Needs - Comparison

IN5 - Christmas movies for the family

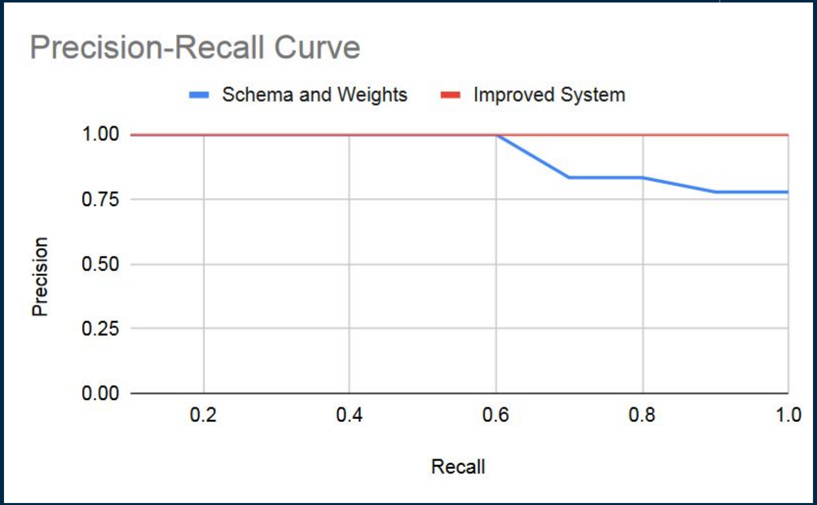
Query (q): Christmas time

Phrase Slop (ps): 5

Query fields (qf): original_title^4, movie_info^3, review_content^2

Filter query (fq): genres: "Kids & Family"

	Milestone 2 (Boosted)	Milestone 3 (Improved)
P@10	0.70	0.90
AvP	0.91	1.00



Conclusions

Mean Average Precision (5 information needs)

Boosted System (M2)	Improved System (M3)
0.9228332	0.9577162

Future Work

- Train our own models used in semantical analysis (OpenNLP)
- Feed more data to *rankSVM* ranking model, collected through user feedback