COMPASS2

Table of Contents

COMPASS2.....1
About Ovitas.....1
Sesam4.....1
Resources.....2
Source code.....2
The demo server.....2
Compass2.....3
The idea behind the Compass2.....3
What's implemented?.....3
User manual.....4
Architecture and components.....9
Conclusions.....10
Semantic technologies.....10
The Compass project.....10
The data.....11
The future.....11

About Ovitas

Ovitas helps you make the most of your knowledge! We provide complete solutions for content and document management. All solutions are tailor-made and on proven technology platforms, so you can revamp[1] how you create, maintain, publish and retrieve information. Ovitas consultants are XML based semantic enterprise content management, information architecture design, Topic Maps and semantic search.

Ovitas delivers solutions and services within the areas of Content and Knowledge Management. Our customers benefit from cost-effective tools that help them organise, manage and retrieve valuable information.

Ovitas provides expert design, development, and deployment of content life cycle solutions. Using proven technology for content management, search & retrieval and workflow, we are able to build solutions that fit your needs.

The Ovitas International Network consists of offices in USA, Hungary and Norway. It enables us to provide additional services, solutions, and expertise to our customers – regardless of location.

Contact: info@ovitas.no and www.ovitas.no

Sesam4

Ovitas' semantic search engine, Compass2, based on Topic Maps and RDF technology, enhances findability and makes it easier to find relevant information. It is a part of the Sesam4 project which has as overall goal to make semantic technology understandable and accessible for organizations and small/medium-sized enterprises.

[1] Informal term for maintenance, repair and operations

Resources

Source code

Compass2 is distributed under a license called the GNU General Public License version 3.0. The source code and documentation can be found and downloaded at https://github.com/ovitas/compass2

The demo server

Administration interface: http://compass.ovitas.no
Client interface: http://compass.ovitas.no/compass2-client/
Knowledge base upload: http://compass.ovitas.no/compass2-ng/KBUpload.html
Web service: http://compass.ovitas.no/compass2-ws/

Compass2

The idea behind the Compass2

The Compass framework provides ability to increase relevance of the results presented to the users who are searching for the information in huge information pools.

Using Compass, the end user can get useful information related to a query term, even when this term does not occur in any document in the search domain. The model assists the information retrieval action in such a way that the user will get relevant hits without having deep knowledge of the domain terminology. Compass expands the query with similar terms and synonyms, and returns more hits that are semantically related to the query term. These hits are weighted, so it is possible to present the relevant hits only and ignore the non-relevant ones.

Basically, if two keywords are semantically inter-related, then using one keyword make also a related keyword a “valid” one. How “valid” it's - depends, beside other things, also of the strength of relation between those two keywords (or topics). So if keyword is for example “Vehicle” then the related words Truck, Van, Bus, Tank, Bicycle, Motorcycle etc. are also a possible candidates of becoming a search keywords. In order for this to work, a searched term has to have an entry in used semantic map – one or many.

Most aspects of such semantically enhanced search engine are tunable, even dynamically.

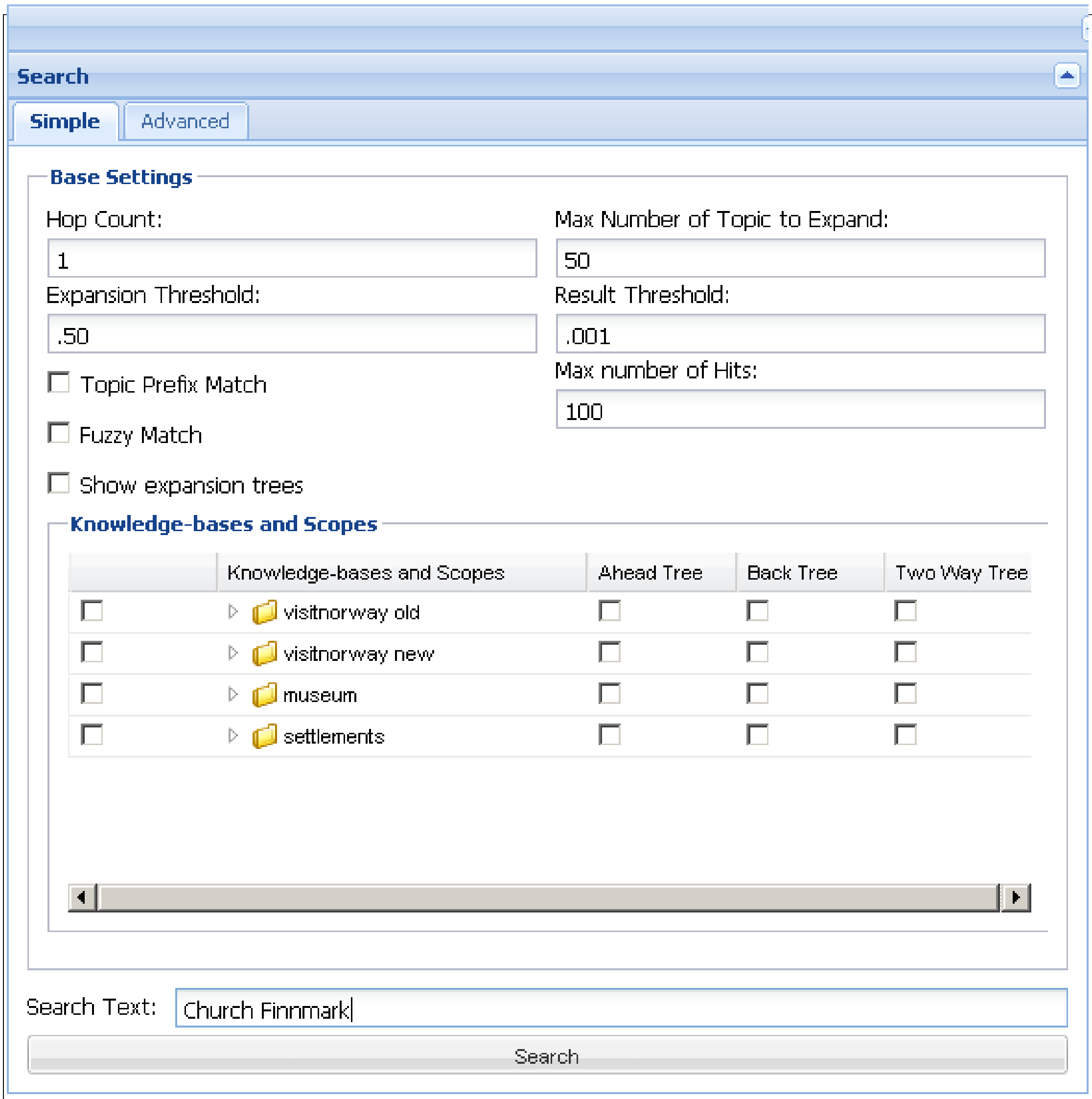
What's implemented?

We took an open source search engine Lucene and some other open source tools and decided to find out if we can implement “Score Boosting” by using available semantic technology and relevant open standards.

The web interface for Compass is not meant as a commercial solution, but rather as an experimental interface in order to test the abilities of such a framework. Actually the beauty is rather to be found under the hood then on the surface of the system.

User manual

To start to use the application, open your web browser and enter “compass.ovitas.no” in the address field. What you see is the web based admin interface to search through VisitNorway web pages by using given keywords as an input at the bottom of screen under the “Search Text”.
This interface is not intended to the end user. This is an admin tool, enables the project team to configure, fine tune and thoroughly test the parameterization of the implemented solution.



Base Settings

Hop Count: 1

Max Number of Topic to Expand: 50

Expansion Threshold: .50

Result Threshold: .001

Max number of Hits: 100

□ Topic Prefix Match

□ Fuzzy Match

□ Show expansion trees

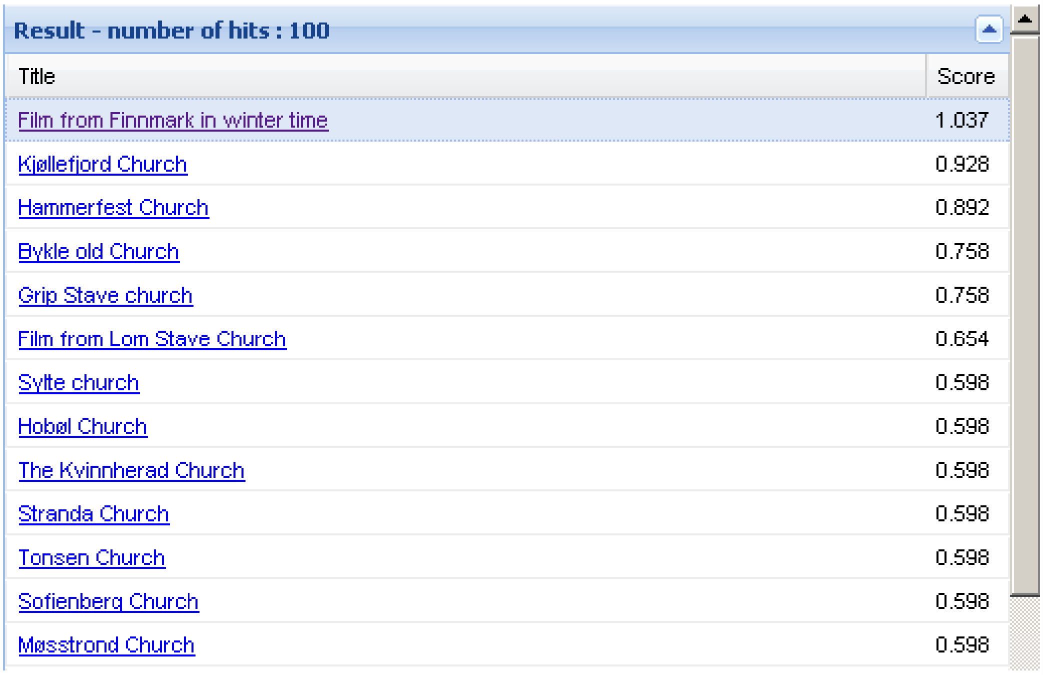
Knowledge-bases and Scopes

	Knowledge-bases and Scopes	Ahead Tree	Back Tree	Two Way Tree
□	▷ visitnorway old	□	□	□
□	▷ visitnorway new	□	□	□
□	▷ museum	□	□	□
□	▷ settlements	□	□	□

Search Text: Church Finnmark

Illustration 1: Compass2 current user interface.

Lets say, you are planing to visit Finnmark, a northern province in Norway and you are interested in visiting local churches. You can fill in “Church Finnmark” and hit the “Search” button. The result, per default, is the same as you'd be using the plain Lucene full-text search engine - as it is configured per default.



Result - number of hits : 100

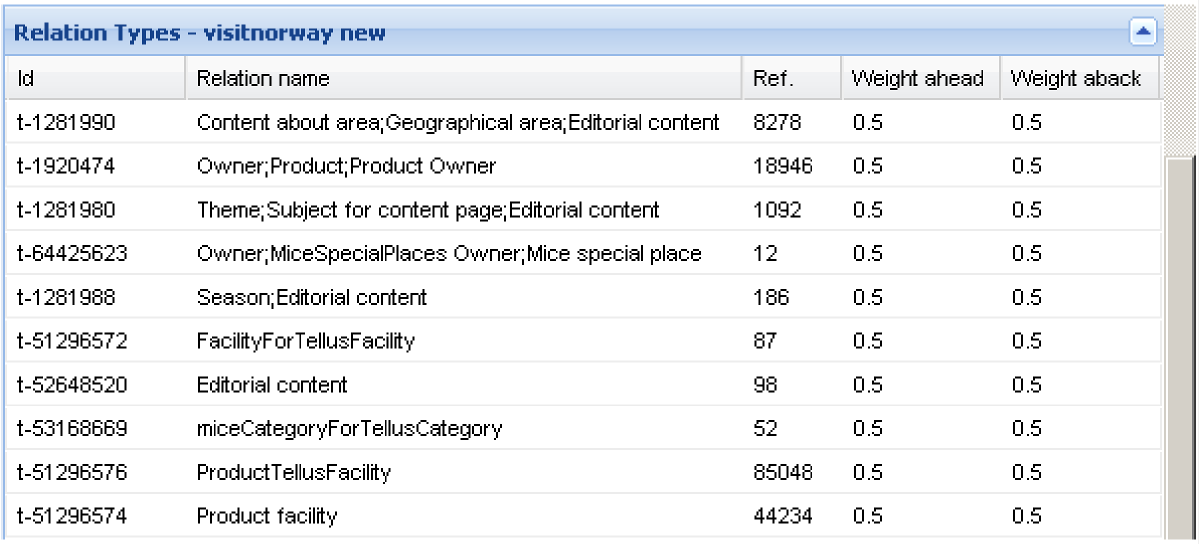
Title	Score
Film from Finnmark in winter time	1.037
Kjøllefjord Church	0.928
Hammerfest Church	0.892
Bykle old Church	0.758
Grip Stave church	0.758
Film from Lom Stave Church	0.654
Sylte church	0.598
Hobøl Church	0.598
The Kvinnherad Church	0.598
Stranda Church	0.598
Tonsen Church	0.598
Sofienberg Church	0.598
Møsstrond Church	0.598

Illustration 2: The default search result.

The results shows the items - relevant pages and the Lucene score for each “hit”.

The real fun begins when you “plug-in” usage of provided semantic maps under “Knowledge-bases and Scopes”. There are four of them, available for your convenience. The two on the top are obtained by VisitNorway at different points in time as a Topic Navigation Maps. The third one contains a semantic map regarding Norwegian museums and the last one contain relations between the provinces in Norway and towns. Each semantic map represents a certain scope of relations between the keywords or ontology.

If you choose, for example, “visitnorway new” you are using a relations between the keywords based on VisitNorway's topic map, English version to be noticed. The list of all “Relation Types” for chosen scopes appears at the bottom of the page. As relations between the keywords can have different weights, depending of the direction, you can choose “Ahead” relations, “Back” relations or both. Clearly a term “Car” is a 100% a “Vehicle” but “Vehicle” is maybe only in 10% of cases a “Car”. So the the relations between these two terms may have different weight or importance. What is the difference – it's up to you to define. As a default, all weights are set to 50% (0.5).



Relation Types - visitnorway new

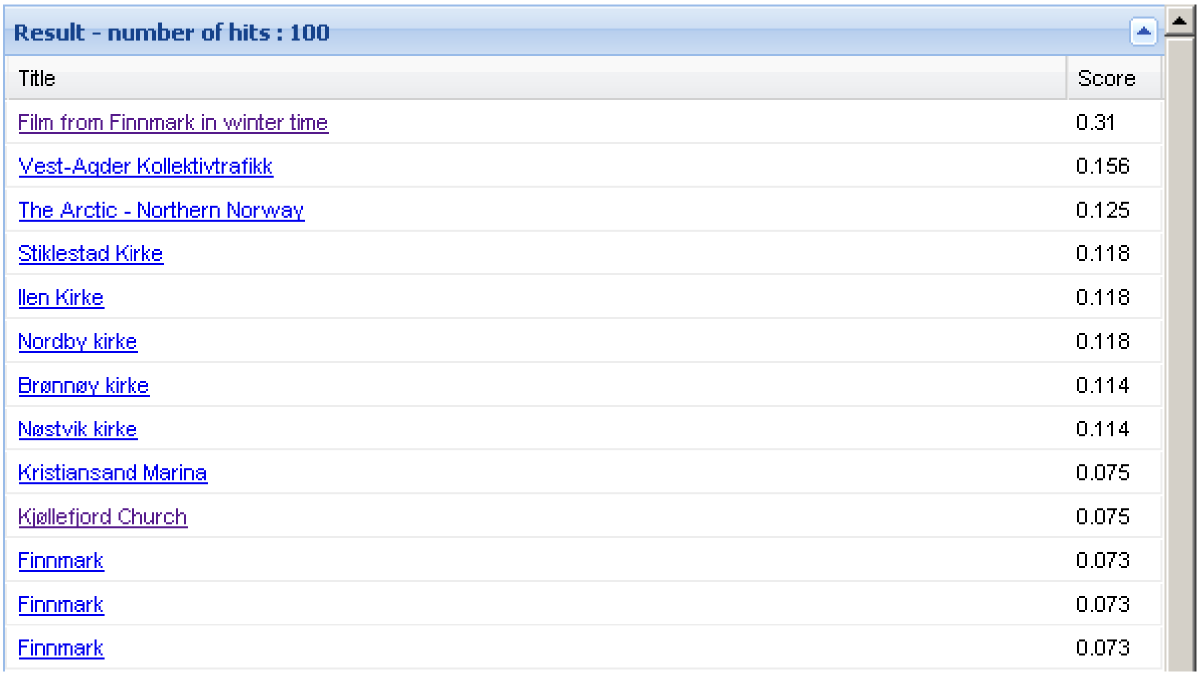
Id	Relation name	Ref.	Weight ahead	Weight aback
t-1281990	Content about area;Geographical area;Editorial content	8278	0.5	0.5
t-1920474	Owner;Product;Product Owner	18946	0.5	0.5
t-1281980	Theme;Subject for content page;Editorial content	1092	0.5	0.5
t-64425623	Owner;MiceSpecialPlaces Owner;Mice special place	12	0.5	0.5
t-1281988	Season;Editorial content	186	0.5	0.5
t-51296572	FacilityForTellusFacility	87	0.5	0.5
t-52648520	Editorial content	98	0.5	0.5
t-53168669	miceCategoryForTellusCategory	52	0.5	0.5
t-51296576	ProductTellusFacility	85048	0.5	0.5
t-51296574	Product facility	44234	0.5	0.5

Illustration 3: Relation Types for "visitnorway new" semantic map

You can manually change a value for weight by double-clicking at the value for the respective “Ahead/Aback” column. A max value of 1 (100%) tells the system to treat these two relative terms as equal or synonyms. A min value of 0 means that the system should ignore this relation. A value of for example 0.2 may be interpreted as: there is a certain relation between the terms, but its not a significant one.

Remember that the relevance between the terms is directional, so you have to choose which direction to traverse. Choosing the “Ahead Tree” will enable expansion of keywords that are search after by also those that are related by looking ahead, and only those weights values will be used for calculating the level/degree of relations.

After you chose also a “Ahead Tree”, hitting the “Search” button again, will change the occurrence and relevance of items in the “Result” list.



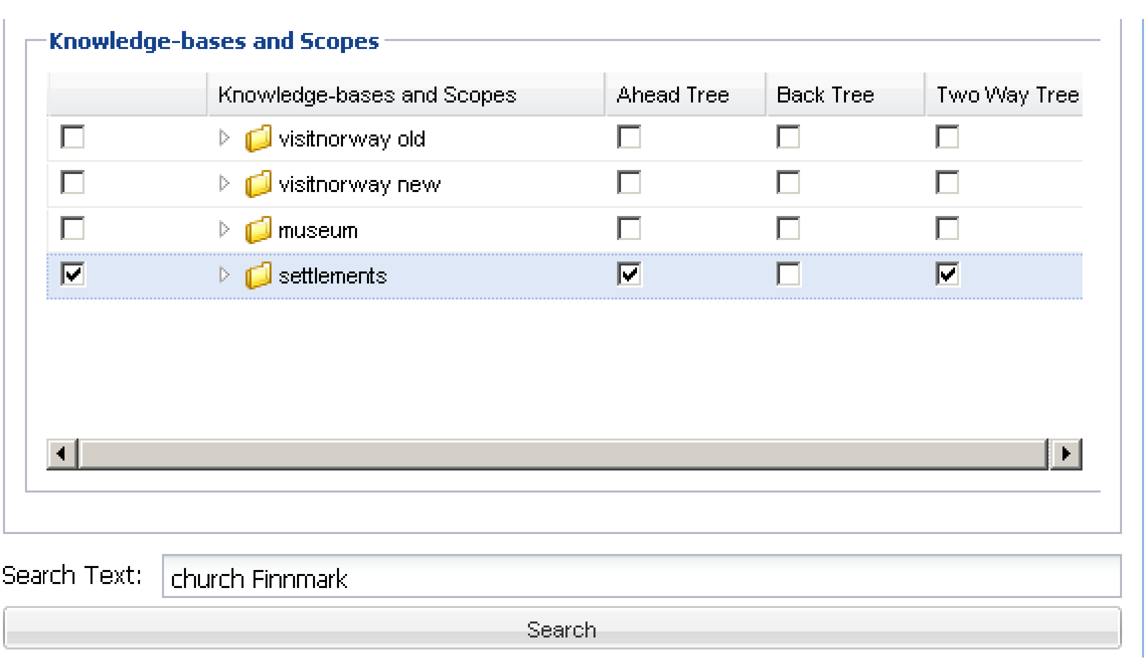
Result - number of hits : 100

Title	Score
Film from Finnmark in winter time	0.31
Vest-Agder Kollektivtrafikk	0.156
The Arctic - Northern Norway	0.125
Stiklestad Kirke	0.118
Ilen Kirke	0.118
Nordby kirke	0.118
Brønnøy kirke	0.114
Nøstvik kirke	0.114
Kristiansand Marina	0.075
Kjøllefjord Church	0.075
Finnmark	0.073
Finnmark	0.073
Finnmark	0.073

Illustration 4: The search result using "vivsitnorway new" ahead relations.

Well, the first item is the same, but all others are picked now ad sorted out based on VisitNorways defined semantic relations and their weight which decide a final score for each result item. As more the used semantic map are "intelligent" as more relevant results are shown at the top of the result list.

How this works in practice. If you choose, for example a "settlements" knowledge-base, there is only one relation type "Member of" which records that a given town is a member of a given province. Choose "Two Way Tree" option and the screen should look like this:



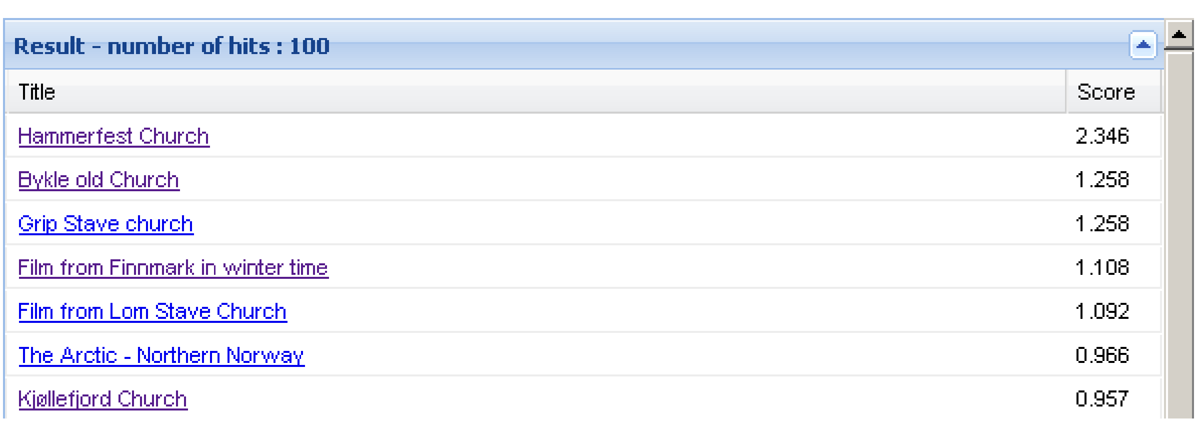
Knowledge-bases and Scopes

	Knowledge-bases and Scopes	Ahead Tree	Back Tree	Two Way Tree
☐	visitnorway old	☐	☐	☐
☐	visitnorway new	☐	☐	☐
☐	museum	☐	☐	☐
☑	settlements	☑	☐	☑

Search Text: church Finnmark

Illustration 5: Setup for using "settlements" knowledge base

And by hitting the "Search" again the result will show something like this:



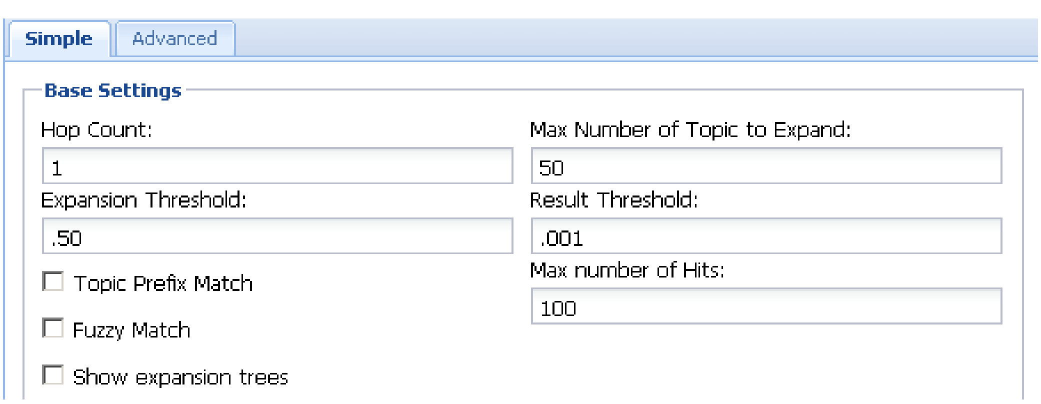
Result - number of hits : 100

Title	Score
Hammerfest Church	2.346
Bykle old Church	1.258
Grip Stave church	1.258
Film from Finnmark in winter time	1.108
Film from Lom Stave Church	1.092
The Arctic - Northern Norway	0.966
Kjøllefjord Church	0.957

Illustration 6: Result for previous search

A new sorting order, different items in the result list and different scores. If we, for example, hit the result under the "Hammerfest Church" it should be linked to the VisitNorway on line web page. Well, to notice, if those two are in sync, as VisitNorway updates their pages daily, but we update our indexes only now and then. By looking in to the web page we can clearly find the word "Church", but not the word "Finnmark". Still, the rank of the page is high thanks to the relation between the town Hammerfest and the province Finnmark from the "settlements" domain. This relation strengthen the rank in such a way so that the page achieve a top ranking.

Each search can be “tuned” bu adjusting the values at the top of the search page.



Base Settings

Hop Count: 1

Max Number of Topic to Expand: 50

Expansion Threshold: .50

Result Threshold: .001

Max number of Hits: 100

□ Topic Prefix Match

□ Fuzzy Match

□ Show expansion trees

Illustration 7: Search options.

Hop Count: The maximal number of “jumps” from one topic to another.

Max Number of Topic to Expand: The maximal amount of topics to be traversed

Expansion Threshold: The topic is added to the key words only if the product of all weights along the path are higher or equal this value. It's expressed as a percentage.

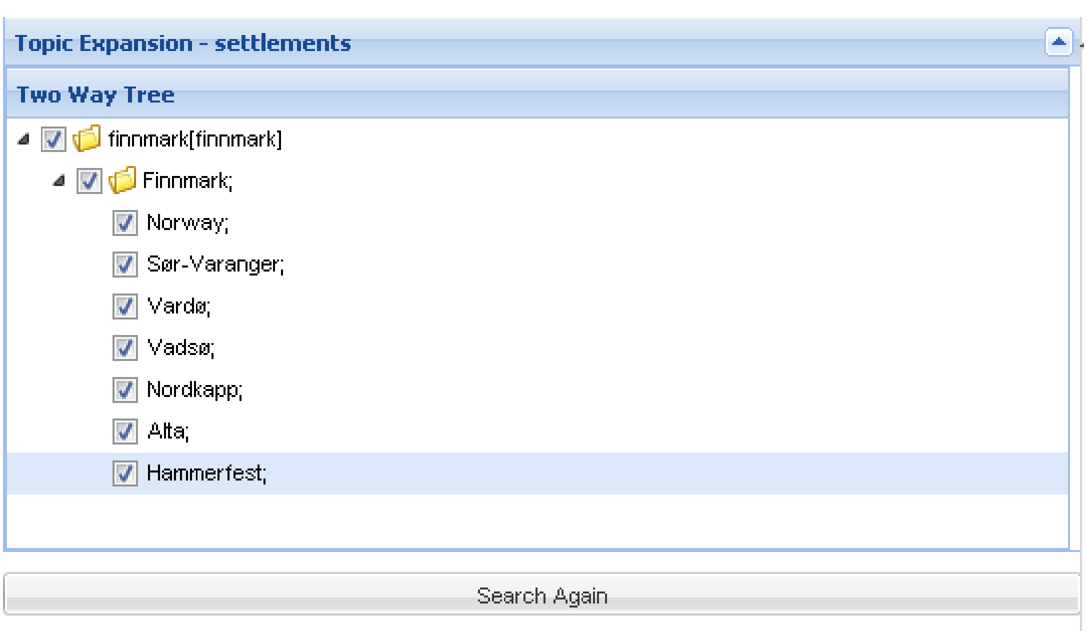
Result Threshold: Show only the result above or equal the given (Lucene) score.

Max Number of Hits: Maximal number of items ti the result list to be returned.

Topic Prefix Match: A topic (from the knowledge model) is match if the search word is a prefix of the topic.

Fuzzy Match: Uses Lucene's fuzzy search, which is based on the Levenshtein distance algorithm.

Show expansion trees: I turned on, the list of all relevant tree fragments traversed are shown at the top of the listing of results window.



Topic Expansion - settlements

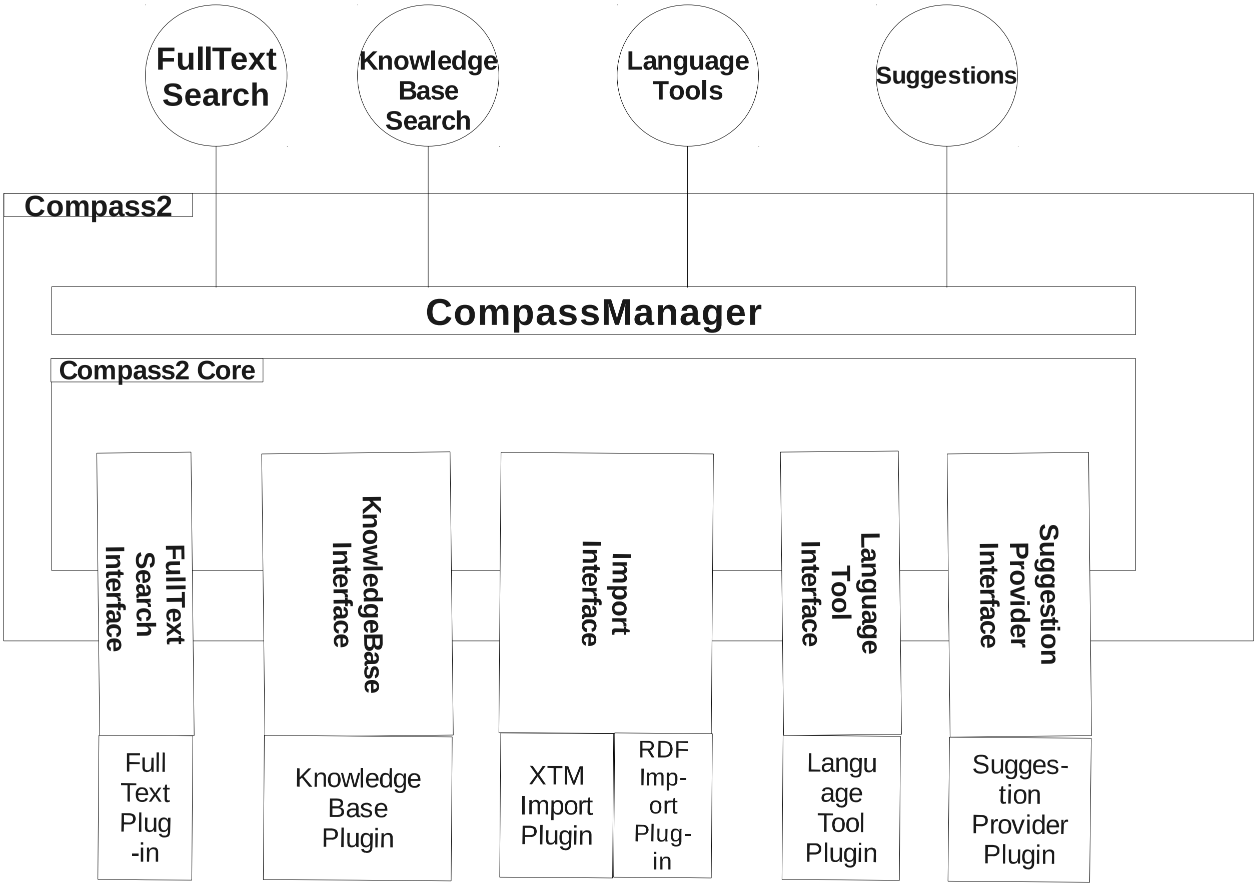
Two Way Tree

- ☑ finnmark[finnmark]
  - ☑ Finnmark;
    - ☑ Norway;
    - ☑ Sør-Varanger;
    - ☑ Vardø;
    - ☑ Vadsø;
    - ☑ Nordkapp;
    - ☑ Alta;
    - ☑ Hammerfest;

Illustration 8: Expansion tree

Here you may choose to remove (unchecked) some of the used "Expansions" or added key words. For example, removing "Norway" and "Alta" will reduce the score for second entry in the result list "Bykkle old Church" (see illustration 7). This is a church located in other province then Finnmark and this is obviously wrong answer. Why it will reduce? Because almost every web page at VisitNorway has several occurrences of the keyword "Norway" and the mentioned page contain a keyword "altar" which is hit by the search term "Alta" (a town in Finnmark) and consequentially increased the page relevance. Human see it at once, but the computer have difficulties in distinguishing between English words and Norwegian names.

Architecture and components



- Five APIs make up Compass. The main API is called Compass and this manages the other APIs which are responsible for Import interaction, KnowledgeBase interaction, FullTextSearch interaction and LanguageTools interaction.
- The KnowledgeBase API handles interaction with the KnowledgeBase, manages importing and updating process and it provide searching facilities.
- The FullTextSearch API handles interaction with whatever full text search engine is being used. It should expose methods for performing searches, updating the index and adding/deleting documents from the index.

- The LanguageTools API should expose methods for stemming and spelling suggestions.
- The Import API handles KnowledgeBase files import, be it RDF/OWL or TM.
- The Compass API is a single entry point to the other interfaces. It exposes one single method, search.It starts by querying the KnowledgeBase for related topics to the search, then queries the FullTextIndex for the constructed query from the entered search and the related topics to the search. It then returns both the results from the FullTextSearch and the result from the KnowledgeBaseQuery for further processing by web applications.

Conclusions

Semantic technologies

It's all about knowledge. When you enter a tourist information office in Oslo, and simply ask about the best way of traveling to the town Ålesund - you are expecting to get the answer which is best aligned with your needs. If you leave the office with the decisions of how to proceed, then you took part in a **knowledge transfer** and this knowledge represents value for you. If you leave the office with a full bag of time-tables and brochures then you took part in an **information transfer** and the bag represent only a potential value for you.

One other thing to observe is the ability of human to gather, process and communicate the information. The clerk at the desk will automatically assume that you want to travel from Oslo to Ålesund. He/she will also assume that the time of the travel will be in the nearby future – how near, it will be determinate by the by the tempo you enter the office and intensity in your voice. If you are with a company, it may signal that you are traveling with a soccer team or with a family, which may greatly affect the content of the knowledge passed to you. Few control questions may narrow the available alternatives to the manageable number. To emulate such service by contemporary automated informational systems is still a technological dream. Especially if the hardware and logic implemented in the software operates on semantically incoherent information pool. Semantics, in the informatics is all about formally expressing what things are and how they are interrelated – the meaning of thing.

Getting all the available information acquires the information/knowledge. Administration, processing and delivery of knowledge is the most challenging tasks for the owners of information pools and information engineers acting on their behalf as the modern information systems do not possess the ability of associative reasoning as the humans does. The automated systems can only deal with explicitly coded data structures – if not, you are destined to implement the reasoning capacity of the human mind and such a task is at the best very resource demanding. A human can at - the glance on the sheet of paper containing the highly formated text - recognize the structure in a fragment of a second, components and even the meaning of this document. For the computer, even recognizing the single letters by their graphical representation represent a immense processing task, not to mentioned dates, names, structure and even the meaning of the document. Semantic technologies are one of ways to address such a problem and is gaining momentum in later years.

The Compass project

Clearly, the idea behind the project is simple. As you hit your search words, they are expanded by all semantically relative terms. Those new terms may contribute in increasing the relevance of possible hits by “adding” a weight to them or even introducing new members in the result set. The “strength” of semantic relations determinate if the candidate term is added to the set of search words by calculations done along the relational path between the terms. If the “weight” is under a given threshold, the relevance is assumed negligible and both the term and further traverse is abandon. When sorting the result of the search, the additional weight is added to the each item in the result sett based on the weight of semantic relevance for the term that caused the “hit”.

The Compass system can be dynamically tuned so that each search may have it's own logic regarding the direction of traversal as well as the relevance threshold.

The weight of relevance between the terms may also depend on area of concern you are performing your search under – or domain. In the domain of “soccer” the weight of direct or indirect relations between the towns Madrid and Manchester is presumably different then in the domain of “tourism”. Same goes for relations between the people – the weight can be different if consider under the of biological, social, professional, medical or legal domains. The Compass system can operate on several such domains – each can be turned “on” or “off” for each search.

The data

Well, playing with Compass is a great fun, but when we started to analyze the results we soon discovered that the value gained is proportional to the quality of the underling data.

First, the web pages had to be “washed” for all coding supporting the layout, ads, links and other funny stuff which may increase readability for humans but are real nightmare for the indexing engine as they introduce a waste amount of keywords hit that are completely irrelevant with the core content of that same page. This is a the same problem for traditional indexing of web content. The obvious solution is to have access to “clean” core data which is more meaningful to index as well as the address of target web page this data will end up on. We didn't have access clean content for now.

Then there is a quality of imported semantic maps. We pretty quickly found out that Topic Navigation Maps provided by the visitnorway.com are “flat”, “unintelligent” and basically follows the structure of web pages. Still we managed to achieve some intelligent response from the system and it was a joyful task to analyze the reasons behind the meaningful answers the Compass delivered thought the result set and the ranking. The “knowledge” could always be traced back to the quality of underling data sets.

We clearly underestimated the resources needed to “wash” the data and provide the semantic maps with sufficient level of “intelligence”. In this round, we concentrated ourself on implementation details. One wisdom to pass is: it does not matter how smart your application is if the data on which the application operates is “messy”.

But still, the results are encouraging and we at Ovitas should like to elevate the current framework to the level of commercialization – if we manage the scramble resources for doing so. Finding a suitable use-case (data pool and usefulness), high quality knowledge data and implementing a fancy GUI on top is the way to go - in order to provide a clear business case.

The future

There is no doubt in our mind that this is a way to go regarding how to organize the information and make it universally accessible and useful. By introducing the relational weight and threshold on the umbrella topic map, we are actually emulating how neural networks work, a principle used by the human brain to perform its cognitive tasks.
By simply loading a Topic Navigation Map containing bilingual dictionary, you may get relevant hits on pages in all languages you understand. By loading a semantic map containing knowledge

regarding fishing, for example you can get more relevant hits if this is your domain of interest. System can also enable determination of target advertisements so they are relevant to their context and the user that is viewing them, based on relevance. So you are spared for flashy stake house advertisement when you enter the word “vegetarian” in your search.
There are another possibilities. By monitoring the user navigation, you may increase (or decrease) the weight between the given keyword in each respective page and keep that record in its own domain, so that subsequent user may benefit on previous navigation patterns. This means that if many users jump from less relevant pages to hyper-linked page with greater relevance in respect to entered search terms, the system may “correct” itself over time. Here we are actually emulating the back-propagating neural networks and accumulation of “knowledge”. Possibilities are endless.

More and more semantic data is available for free on the net. The content provider who owns the best of them will be the winers of tomorrows effort to commercialize a knowledge as a service based on axiomatization tools – or plainly, computers.