Watson Concept Insights

A Conceptual Association Framework

Michele M. Franceschini, Livio B. Soares, Luis A. Lastras Montaño
IBM T. J. Watson Research Center
1101 Kitchawan Road
Yorktown Heights, New York 10598
franceschini@us.ibm.com, lsoares@us.ibm.com, lastrasl@us.ibm.com

ABSTRACT

Watson Concept Insights (WCI) is a service that was recently made publicly available by IBM. WCI provides an information retrieval framework that is designed to facilitate search and exploration of text documents, and is particularly effective on sparse data sets. Its methodology consists of first defining a dictionary of concepts which are interconnected in a concept graph and then modeling a document by predicting its relevance to any given concept in the concept graph using the concepts that are directly mentioned in the document itself. This technique in effect increases the document recall for any given query, even for very sparse data sets, exposing the user to a variety of connections between their query and a data set of interest.

Keywords

cognitive computing; information retrieval;

1. INTRODUCTION

Consider the following questions: given a topic, who can help me learn more about it? What has been published in the web in the last day relevant to it? What has a specific author ever written on it? What has this company patented about it? Is there anyone in my vicinity who has blogged about law? What does this set of books say about it?

In these questions, there is a simultaneous interest in the query and the data set from which the allowable responses can come from. For example, the only people that I can learn from are those that I have effectively access to (e.g., people that I work with), and a company's patent portfolio will necessarily be limited compared to the set of all possible patents ever filed. Whereas in open ended web scale search problems the technical task is to accurately rank a large number of documents that match a given query, the technical challenge when the target data set is sparse is instead to illuminate whatever reasonable connections can be found between a query and the documents, with the aim of teaching the user of the tool about those connections and potentially re-set his or her expectations about what a data set says about a given topic. It is important to note that data sparsity need not exist due to an intrinsic limitation on the available data, and instead is a property of a user's information retrieval goals; for example an investigation limited to all web content that is published by a specific author can also be repeated for a different author and those people who are in my vicinity right now won't be the same as those that are in my vicinity tomorrow.

In this article we introduce a system with attributes that are likely important for systems that aim to specifically answer questions such as the ones posed above. A key element of our proposal is an *all-encompassing knowledge model* that can be used to bridge the gap between the knowledge of the user and the knowledge contained in the data set being perused. A concrete example could be that of a person that has only a basic understanding of mathematics searching for references to math topics in a data set. The person will likely use query terms such as math or algebra. If the data set may contain a document that mentions fast Fourier transforms, which a modern information retrieval system should be able to retrieve and properly rank. The key element is that such an interaction between the person and the system has a fundamentally different nature than that of a person searching directly for fast Fourier transform: the interaction has become *exploratory* in nature. Interacting with the system can now cause *surprise* and *interest*.

The resulting experience is aimed as much at engaging the user as it is at providing the information that the user initiated the interaction with the system for.

Watson Concept Insights (WCI) takes an opinionated approach at providing such an experience by mainly focusing on a specific set of features extracted from documents: *concept mentions*. WCI uses a built-in knowledge model, the **concept graph** that provides a scoring mechanism for how two concepts (or 2 sets of concepts) are related to each other that is computed using the personalized page rank algorithm [5].

This allows WCI to approach each new data set with the model given by the full concept graph, so there is no need for building a new model for each new data set. Therefore relatively deep relationships are discovered even in smaller data sets.

The system provides APIs that are meant to facilitate the interaction with an end user performing exploration on a data set. In Section 2 we'll cover the overall architecture of the system. Section 3 describes the demonstration setup.

Copyright is held by the author/owner(s).
WWW'16 Companion, April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4144-8/16/04.
http://dx.doi.org/10.1145/2872518.2890548 .

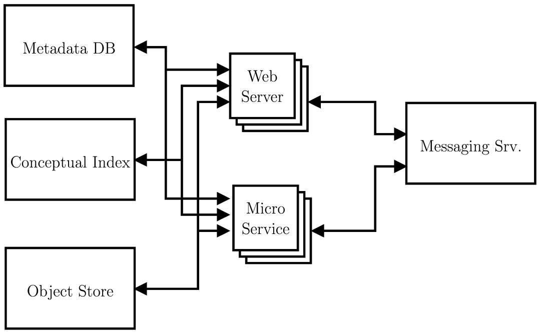


Figure 1: WCI high level system architecture.

2. SYSTEM OVERVIEW

2.1 System Architecture

WCI is implemented as a distributed system providing a web service accessible through a number of REST endpoints [1]. Corpora and documents inside corpora are entities accessed and manipulated through a Create, Read, Update, and Delete (CRUD) interface. Figure 1 illustrates the internal structure of the system, based on a microservice architecture (see, e.g., [4] and references therein). The system includes i) a metadata database ii) an object store iii) a messaging server iv) an array of web servers, v) a number of stateless microservices, and vi) a conceptual index database. Everything in the system is designed for dynamic scale-out. The microservices communicate through the messaging server and have direct connection to the databases and the object store.

2.2 Document Processing Pipeline

In Figure 2, we illustrate the document processing pipeline of WCI. After a corpus (a document collection) is created documents are added through the corresponding REST endpoint. When a document enters the system, it is first stored then staged for indexing. Indexing happens as a background process and includes the following stages:

1. concept mentions extraction;
2. document vector model construction;
3. insertion of the document into a reverse index data structure for fast search.

In the following we briefly describe these 3 phases. An in-depth description is beyond the scope of this demonstration and will be available as a separate publication.

2.2.1 Concept Mentions Extraction

Concept mention extraction in WCI is performed using an in-house algorithm akin to algorithms used in *wikification* [3]. The algorithm allows to isolate mentions to concepts known in the system in the text of a document. It features an advanced disambiguation capability which enables low input (conceptual) noise introduced in the system. The model construction performs further denoising as explained in the following.

2.2.2 Document Model Construction

WCI uses a graph of interconnected concepts based on Wikipedia. As the graph is fairly sparse a diffusion method based on the solution of a Markov chain on the concept graph is used as a proxy to evaluate the relationship of each given concept to all other concepts. The internal document model is a vector that ranks the document for each concept in the concept graph. In this sense it represents a large embedding in a space in which each dimension expressed in the canonical basis has a human understandable meaning. Mentions to concepts are treated as independent observations of the relevance of a document to each concept in the graph. An information combining [2] methodology is then used to obtain a final estimate of the document relevance for each concept in the concept graph.

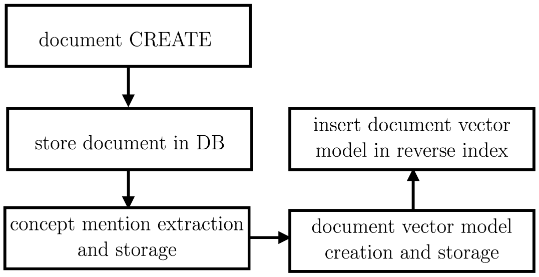

document CREATE
store document in DB
concept mention extraction and storage
document vector model creation and storage
insert document vector model in reverse index

Figure 2: WCI document processing pipeline.

2.2.3 Indexing and Conceptual Search

Once a document vector model is constructed, each dimension representing the relevance of the document for a specific concept in the concept graph, the vector is submitted into a reverse index, which indexes the most relevant documents for each concept in the concept graph, irrespective of whether the document contained a direct reference to a given concept. Retrieving the most relevant documents is an operation denoted in the system with the term *conceptual search*. Conceptual search for a single concept becomes then a simple index lookup. For multi-concept queries and document queries (which are just a form of multi-concept queries), multiple lookups into the index and a reranking algorithm are used to generate the list of most relevant documents on the spot. This methodology allows the system to generate responses with a typical latency well under one second, which helps fostering a smooth interactive experience when the system functionality is exposed directly to a UI user.

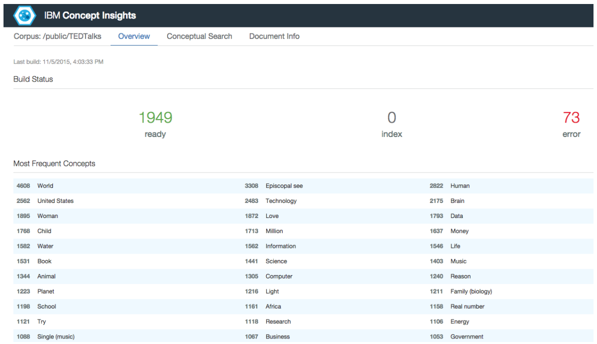
3. DEMONSTRATION

The demonstration of is WCI in 2 parts, one for the experience at the developer and content management side and one at the end user side.

3.1 WCI: Development and Content Management

Although the programmatic interaction with WCI through API allows full control of the system [6], the service has been designed to be monitored through a control dashboard. The dashboard allows to monitor the document insertion process, providing number of documents ready for search, number of document that are being processed for indexing and number of documents in error[1]. In addition the service dashboard

[1] A document is considered to be in error if no conceptual model can be constructed for it, e.g., if a document is empty



Last build: 11/5/2015, 4:03:33 PM

Build Status

1949 ready

0 index

73 error

Most Frequent Concepts

4608	World	3308	Episcopal see	2822	Human
2562	United States	2483	Technology	2175	Brain
1895	Woman	1872	Love	1793	Data
1768	Child	1713	Million	1637	Money
1582	Water	1562	Information	1546	Life
1531	Book	1441	Science	1403	Music
1344	Animal	1305	Computer	1240	Reason
1223	Planet	1216	Light	1211	Family (biology)
1198	School	1161	Africa	1158	Real number
1121	Try	1118	Research	1106	Energy
1088	Single (music)	1067	Business	1053	Government

Figure 3: A screen shot of the WCI Dashboard for the TED talks corpus, showing document counts and the most mentioned concepts.

provides high level statistics that are tightly coupled with the concept-based approach in the system. In particular:

- a summary of the most mentioned concepts in the corpus;
- a histogram representing the distribution of mention counts for the mentioned concepts;
- a histogram of the number of concept mentions per document;
- a histogram of the document sizes.

Figure 3 shows the `read/index/error` counts and the most mentioned concepts for a corpus containing all TED Talks transcripts. The latter allows a human person to quickly gauge the representation that WCI is building of the overall corpus—although additional uses could be conceived, such as the creation of a rich human readable summarization of the salient aspects of the corpus.

In Figure 4, the histograms of the mentions per concept count and the concept mentions per documents are shown, respectively. The former allows to understand the conceptual richness of the corpus: a significant concept diversity enhances the capability of the system to make more nuanced distinctions between documents. The latter gives an idea of the richness of individual documents: although it will work with fewer mentions, in order operate at its best WCI requires at least 10 concepts per document to enhance the confidence of the vector model built for the document.

In Figure 5, we illustrate the *conceptual search* view of the WCI dashboard. This view can be used to explore a corpus through concept queries. In this instance, a corpus that indexes National Public Radio (NPR) news from the past 7 days is searched from the concept common law. The view returns a list of documents each in a tile containing the document label, in this case the news article title, as well as an *explanation* of why the document was returned for such query. An explanation of a result is a necessity when introducing an information retrieval system that performs retrieval in a way not trivially understandable by an end user. This need becomes obvious when the system returns documents that do not contain text that can be derived explicitly from the query. To help in building trust between the end user and the application using WCI, each conceptual search returns an explanation object for each matching document. The explanation object includes the list of most relevant concepts mentioned in the document sorted by relevance for the query that retrieved the document. In the example in Figure 5 no direct mentions of common law were found, however the most relevant documents are explained to be important by the system because they contiained concept mentions that together made the document relevant for the query, such as: expungement, felony, self incrimination, statute.

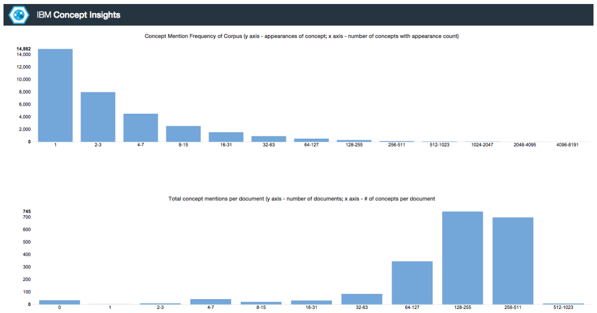
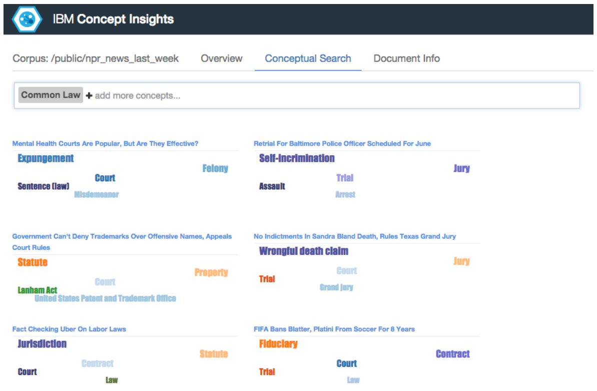


Figure 4: A screen shot of the WCI Dashboard for the TED talks corpus, showing mention per concept and concept mentions per document histograms.



Mental Health Courts Are Popular, But Are They Effective?

Retrial For Baltimore Police Officer Scheduled For June

Government Can't Deny Trademarks Over Offensive Names, Appeals Court Rules

No Indictments In Sandra Bland Death, Rules Texas Grand Jury

Fact Checking Uber On Labor Laws

FIFA Bans Blatter, Platini From Soccer For 8 Years

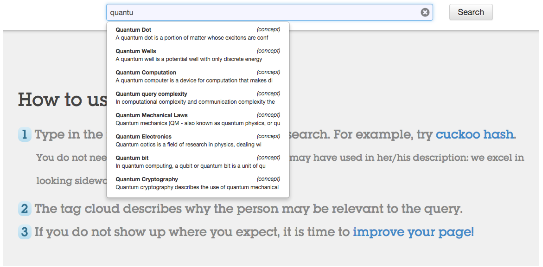
Figure 5: WCI Dashboard *conceptual search* view: searching for last week's news articles related to the concept common law in the NPR news dataset.

It is easy to see how this search experience can be used to foster exploration and even learning, as only explanation concepts only partly familiar to the end user appear, interest arises in learning about those concepts *per se*, as well as new keys to explore the corpus.

3.2 WCI: End User Experience

To demonstrate the end user interaction with the system we will focus on two example applications built on top of WCI: an *expert location* app and a *content recommendation* app.

or no concepts could be extracted it will be counted among the documents in error.



quantu

Quantum Dot *(concept)*
A quantum dot is a portion of matter whose excitons are conf

Quantum Wells *(concept)*
A quantum well is a potential well with only discrete energy

Quantum Computation *(concept)*
A quantum computer is a device for computation that makes di

Quantum query complexity *(concept)*
In computational complexity and communication complexity the

Quantum Mechanical Laws *(concept)*
Quantum mechanics (QM - also known as quantum physics, or qu

Quantum Electronics *(concept)*
Quantum optics is a field of research in physics, dealing wi

Quantum bit *(concept)*
In quantum computing, a qubit or quantum bit is a unit of qu

Quantum Cryptography *(concept)*
Quantum cryptography describes the use of quantum mechanical

How to us

1 Type in the search. For example, try cuckoo hash.

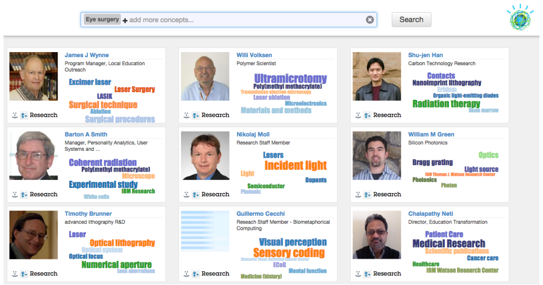
You do not nee may have used in her/his description: we excel in

looking sidewo

2 The tag cloud describes why the person may be relevant to the query.

3 If you do not show up where you expect, it is time to improve your page!

Figure 6: Expertise location using WCI: web app/user interaction for the construction of a query.



Eye surgery + add more concepts...

James J Wynne
Program Manager, Local Education Outreach

Willi Volksen
Polymer Scientist

Shu-jen Han
Carbon Technology Research

Barton A Smith
Manager, Personality Analytics, User Systems and ...

Nikolaj Moll
Research Staff Member

William M Green
Silicon Photonics

Timothy Brunner
advanced lithography R&D

Guillermo Cecchi
Reseach Staff Member - Biometaphorical Computing

Chalapathy Neti
Director, Education Transformation

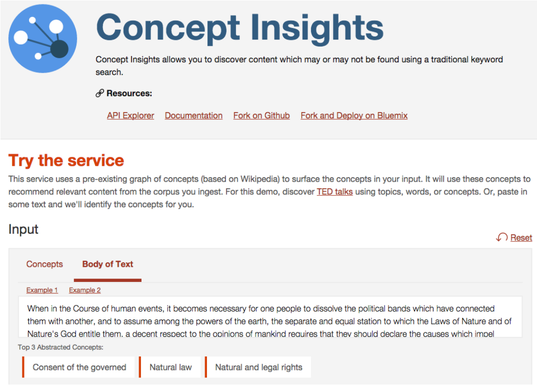
Figure 7: Expertise location using WCI: search results.

3.3 Expertise location with WCI

A basic approach to building an expert location system using WCI is to create a corpus that contains one document per person, describing the expertise of the person. For a large fraction of different expertise, it is generally possible to directly use at least part of the direct output from people. Due to the internal model for concept relationships WCI is effective at ”filling the gaps,” which results in deriving a picture of someone’s expertise even if the description is not as complete. In the example shown in figures 6 and 7 we illustrate an expertise locator for IBM researchers. For this particular corpus, the use of the direct product of the expert’s work as a document is particularly efficient (in this case the research publications). The actual data set consists of a profile page each researcher is maintaining together with the list of publications of each researcher.

The first problem to solve when interacting with the system is to help the end user construct a usable query in an intuitive way. The biggest help comes here from the fact that the concepts used in the system are identifiable by their English name (or in certain instances, short description). However not every English word or construct is a valid concept in the system therefore the chosen approach is to provide an intuitive auto-complete widget that proposes a list of the closest concept names together with a short abstract for the concept in order to enable disambiguation on sight for the end user (a task the average end user happens to be extremely good at). This is illustrated in Figure 6 where a query on quantum computing is being auto completed.

An example results page for the query eye surgery is shown in Figure 7. Note the explanation tags on this query: thanks to both WCI and the richness of the dataset highly related terms, potentially unknown or vaguely familiar to the user, show up. Experience shows that a common human reaction to this is to proceed and explore the meaning of the terms (each one is clickable and links to Wikipedia).



Concept Insights

Concept Insights allows you to discover content which may or may not be found using a traditional keyword search.

Resources:

API Explorer Documentation Fork on Github Fork and Deploy on Bluemix

Try the service

This service uses a pre-existing graph of concepts (based on Wikipedia) to surface the concepts in your input. It will use these concepts to recommend relevant content from the corpus you ingest. For this demo, discover TED talks using topics, words, or concepts. Or, paste in some text and we'll identify the concepts for you.

Input

When in the Course of human events, it becomes necessary for one people to dissolve the political bands which have connected them with another, and to assume among the powers of the earth, the separate and equal station to which the Laws of Nature and of Nature's God entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel

Top 3 Abstracted Concepts:

Consent of the governed | Natural law | Natural and legal rights

Figure 8: Content recommendation using WCI: constructing a query from textual user input.

3.4 Content Recommendation with WCI

An additional example is content recommendation. In this case we selected TED talks as a rich data set encompassing a large span of human interests. Recommendations can be driven by directly engaged user input or gathered textual evidence. In Figure 8 we show the content recommendation application gathering concepts from an excerpt of text. The most relevant three concepts are shown in the bottom left of the figure. We do not show the resulting recommendations due to lack of space.

4. CONCLUSIONS

We introduced Watson Concept Insights, an information retrieval system designed to foster end-user exploration of semantically and linguistically sparse corpora of documents. The system uses as key features mentions to concepts familiar to humans. The demonstration practically illustrates these characteristics focusing on both developer and end-user experience.

5. REFERENCES

[1] R. T. Fielding. *Architectural styles and the design of network-based software architectures*. PhD thesis, University of California, Irvine, 2000.

[2] I. Land, S. Huettinger, P. Hoeher, J. B. Huber, et al. Bounds on information combining. *Information Theory, IEEE Transactions on*, 51(2):612–619, 2005.

[3] R. Mihalcea and A. Csomai. Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 233–242. ACM, 2007.

[4] D. Namiot and M. Sneps-Sneppe. On micro-services architecture. *International Journal of Open Information Technologies*, 2(9):24–27, 2014.

[5] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: bringing order to the web. 1999.

[6] L. Soares, M. Franceschini, and L. Lastras. Watson concept insights api explorer, 2015. [Online; accessed 22-December-2015].