Analogy Queries in Information Systems – A New Challenge

- Pre-Publication Copy - to appear in JIKM volume 12 issue 3 -

Christoph Lofi
National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku
Tokyo 101-8430, Japan, lofi@nii.ac.jp

Keywords
Analogy Processing; Human-Centred Queries; Information and Knowledge Systems; Semantic Query Processing; Knowledge-based Queries

Abstract

Besides the tremendous progress in Web-related technologies, interfaces to access the Web or large information systems have largely stayed at the level of keyword searches and categorical browsing. This paper introduces analogy queries as one of the essential techniques required to bridge the gap between today's interfaces and future interaction paradigms. The intuitive concept of analogies is directly derived from human cognition and communication practices, and is in fact often considered to be the core concept of human cognition. In brief, analogies form abstract relationships between concepts, which can be used to efficiently exchange information and knowledge needs or transmit even complex concepts including important connotations in a strictly human-centered and natural fashion. Building analogy-enabled information systems opens up a number of interesting scientific challenges, e.g., how does communication using analogies work? How can this process be represented? How can information systems understand what a user provided analogy actually means? How can analogies be discovered? This paper aims at discussing some of these questions and is intended as a corner stone of future research efforts.

1. Introduction

The Web as a global Information System has revolutionized everyday life. As one of the most disruptive technologies of the last decades, the Web was responsible for drastic technological, economical, and social developments: it is established as main source of information and entertainment, but is also the most influential infrastructure for commerce and business. Especially, the advent of information retrieval technology had far reaching consequences and search engines revolutionized effective information access and navigation in large knowledge spaces. Since then, the Web has prospered, diversified and developed into the largest and omnipresent information source.

However, the respective interaction and navigation methods have been progressing comparatively slowly, and even modern systems use techniques for retrieving, finding, and navigating information like those developed a decade ago. Techniques like e.g. hierarchical categorization, list browsing, filtering, online forms, or simple keyword searches are still predominant as sole method of accessing content or services. These techniques represent a *system-centric* approach towards retrieving information and focus on efficient implementation. However, at least since the recent advent of Web 2.0 applications, the Web began to change and adopted a more *user-centric* approach. Web 2.0 focuses on people's participation: in contrast to just consuming content, now everybody can contribute something to existing Web content and services. While this development had immediate impact on our usage of the Web (social networking, Wikis, blogging, etc.), it had only a minor impact on navigation and access patterns to information. In brief, the availability of user provided tags, votes, comments, tweets, blogs, or social network links did allow for some new techniques like e.g. popularity votes, social recommendations, or tag cloud summaries. However, the main means of searching for and locating information are still keyword search and categorical browsing.

The central message of this paper is that also user interaction methods with respect to finding relevant information, items, or services has to progress to the next level. Interaction with Web platforms and information systems should be more human-centered, and should aim at adopting natural interaction and communication patterns commonly used by humans: In detail, we propose the need for focusing on developing techniques and algorithms for performing *analogy queries*. While observing human communication, it has been discovered that one of the most common and fundamental concepts of human communication is the use of analogies (Hofstadter 2001). Within certain limits, analogies allow for a very efficient description, explanation, and exchange of complex concepts. For example the famous Rutherford analogy claimed that "atoms are like the solar system", and thus quickly and efficiently communicated the inner workings of the newly discovered microcosm to public. Moreover, they also allow for descriptive querying ("I am looking for a flat in a district of Berlin, which is like Soho in London"). Hence, analogy queries are a natural extension of system-centric interaction patterns but represent a significant step beyond commonly used keyword search.

Unfortunately, up to now analogy queries have been largely neglected by the information system and database community. One reason is that discovering sensible analogies in data needs the analysis of vast amounts of user provided semantic data or general information available via the Web to find common conceptualizations. Furthermore, an analogy-enabled information system should not only be intended to discover analogies in a given domain and provide facilities for efficiently retrieving items with analogy queries, but should also be able explain the underlying analogy of retrieved items even across domains, thus also catering for trans-disciplinary or inter-disciplinary tasks. These requirements were traditionally hard to fulfill, therefore most works on analogies deal with psychological or philosophical issues, and those few analogy-enabled systems designed up to now usually focused on small examples in tightly circumvented domains. However, with the rise of Big Data and Linked Open Data, vast and untapped data sources are available for exploitation by future analogy-enabled information systems.

The goal of this paper is to provide fundamental insights and discussions for further advancing the vision of analogy-enabled information systems, and should be seen as a positional work to serve as a basis for future research. Especially, the following issues will be addressed:

- We will provide an overview of *related work* in the field of analogy modeling and computer-based analogy processing. Here, we will especially focus on *philosophical*, *psychological*, and *linguistic research*. These fields used to be the driving communities in analogy research, but we will also highlight some of the few works from the information systems community.
- We will outline the benefits, *motivations*, and challenges for the application of analogy-enabled information processing for human-computer interactions, especially within the information systems area.
- We will briefly present and discuss a generic *architecture* for designing analogy-enabled information systems.
- We will design a *language* for *analogy programs* called analogy^E. Using analogy^E, *analogy problem* and *analogy queries* can be easily represented.

2. Motivations and Famous Analogies

In this section, we will discuss the expressive power of analogies by showcasing some famous analogies. Especially, these examples show that analogies excel at several (closely related) central tasks:

- *Visually explaining* complex concepts by mapping them to simpler and easier to understand concepts with similar functions. While these simpler concepts usually are not enough to cover the complex concept in its whole (i.e. the analogy will "break" at some point), explaining by analogy can represent the central workings and important facts quite well (usually, knowing these is enough for non-domain experts).
- *Teaching* unknown concepts by mapping them to known concepts. As an extension of the previous point, this outstanding ability secures analogies a central role in modern pedagogics.
- *Describing concepts* without using the respective domain vocabulary or domain knowledge. This ability of analogies has a special importance in nowadays work life, as interdisciplinary teams become more common and need to interface quickly.

2.1. Rutherford Analogy: An Atom is like the Solar System

Initially, this analogy was developed by Earnest Rutherford in the early 20th century, and was used to explain the basic workings of the microcosm to Rutherford's peers in academia, but also to the wider public. Rutherford built a direct analogy between atoms and the Solar system, equating the sun with the nucleus and claiming that the movement of electrons is similar to the circular movements of planets. Furthermore, the analogy also encodes the imbalance in the mass distributions: nearly all mass of the Solar system (~97%) is in the sun, as most mass in an atom is in the nucleus (99.9%). Later, the physicist Niels Bohr was awarded the Nobel Prize in Physics for using a refined version of the Rutherford analogy known as Bohr-Rutherford model to explain the hydrogen atom. While this analogy quickly breaks for atoms larger than hydrogen, and fails to explain some more complex quantum mechanical effects, it still quickly offers some basic insights into the microcosm and is therefore frequently used in elementary physics education. This analogy is a prime example how the essence of complex and foreign concepts can be communicated and explained easily.

2.2. Drain Pipe Analogy: Electrical Circuits are like Hydraulic Circuits

This famous analogy dates back to the mid-19th century; a time where electricity was just discovered and scientist were still struggling hard to understand its basic properties. This analogy sets *electrical circuits* (direct and alternating current) in relation to the much easier-to-explain and visualize hydraulic circuits. Early on, the analogy was faithfully championed by Sir William Henry Preece, who used it extensively for though experiments. Despite its limitations, it is still widely popular in early school education before moving to more complex models.

We will go into more detail with this analogy, as it nicely illustrates *analogical mapping*: basically, the overall analogy is built by claiming that wires are like closed pipes with removable caps. When two wires are connected, this is represented by connecting two pipes (and prior removing the closing caps at ends to be connected). A closed electrical circuit with a power source (either a voltage source or current source) can be seen as filling the pipe system (which is laid out horizontally for ignoring gravity effects) with water and having suitable pump (the 'power source'). Here, water volume takes the place of electrical charge; water flow rate corresponds to electrical current, and the pressure difference between various sections of the pipe system is treated as voltage. Basic electrical components can be modelled as follows:

- A *resistor* is represented by a *constricted pipe*, which restricts the water flow. Also, all normal pipes have some small flow resistance, as wires have electrical resistance.
- A *capacitor* is represented by an *elastic, impervious membrane* within the pipe, dividing it into two separate sections. When water is forced into the pipe, equal water is forced out the other side, yet no water penetrates the membrane. Energy is stored by the stretching of the membrane, and is released when the pressure subsides.
- A *diode* is realized by using *one-way check valve* with a slightly leaky valve seat. Water can flow mostly only into one direction (in the other direction, only a minor water flow is possible due to the leak), and a little pressure is needed to open the valve. Like a diode, the valve can easily be destroyed by too much pressure / voltage.
- Most other basic components like transistors, switches, memristors, or grounding can be modeled in a similar fashion.

This analogy serves well for explaining basic properties and phenomena, and even allows for deriving some fundamental equations like Ohm's Law. However, as with all analogies, also the drain-pipe analogy breaks when going into too much detail: for example, electrical fields or quantum-mechanical effects of electrical circuits cannot be explained using pipes; similarly, leaking pipes do not translate to wires as they do not loose electrons.

2.3. Interdisciplinary Research and Work: Bio-informatics

While the previous two analogies were used to explain newly discovered (and at that time quite confusing) natural phenomenon in the time of their discovery (and are still used in school education), analogies nowadays are also very frequently used to summarize the essence of complex topic quickly and efficiently in simple words. This feature is of very high importance in interdisciplinary research or interdisciplinary work. We will highlight this issue briefly using the example of information systems in bio-informatics. Here, biologist and computer scientists quickly need to understand at least the core issues of both disciplines in order to cooperate and understand each other. However, deeper understanding of the domain knowledge of the other discipline is usually not required. For example, when looking up the quite complex concept of "Mitochondria" in Wikipedia, one of the first sentenced will be "Mitochondria are sometimes described as 'cellular power plants' because they generate most of the cell's supply of adenosine triphosphate (ATP), used as a source of chemical energy." This simple analogy contains all information necessary for a basic understanding and additional knowledge, e.g. with respect to synthesis, detailed functions, or byproducts, are usually not necessary for a computer scientist collaborating with a cell biologist.

3. Foundations of Analogy

Most human cognition is based on processing similarities of conceptual representations. During nearly all cognitive everyday tasks like e.g., visual perception, problem solving, or learning, we continuously perform *analogical inference* in order to deal with new information (Gentner and Markman 1997) in a flexible and cross-domain fashion. It's most striking feature is that analogical reasoning is performed on high-level relational or even perceptional structures and properties. Moreover, in contrast to formal reasoning, deduction, or formal problem solving, the use of analogies (and also analogical inference) appears to be easy and natural to people. As analogical reasoning plays such an important role in many human cognitive abilities, it has been suggested that this ability is the "core of cognition" (Hofstadter 2001) and the "thing that makes us smart" (Gentner 2003). Due to its ubiquity and importance, there is long-standing interest in researching the foundations and principles of analogies, originally in the fields of philosophy, but later on also in linguistics and mainly in cognitive sciences. Therefore, the goal of this section is to highlight some of the aspects which are commonly agreed upon, but also showing common and sometimes even contradicting models which were developed. This discussion helps understanding the basic concepts of analogies and provides rationales for the design decisions of our analogy^{E} analogy language.

3.1. What is "Analogy"?

In general, an analogy is a cognitive process of transferring some high-level meaning from one particular subject (often called the *analogue* or the *source*) to another subject, usually called the *target*. When using analogies, one usually emphasizes that the "*essence"* of source and target is similar, i.e. their most discriminating and prototypical behaviors are perceived in a similar way. Actually identifying this essence of objects is one of the great challenges in building an analogy-enabled information system.

One of the early and more formal and narrow notions of analogy date back to Immanuel Kant (Kant 1790), whose understanding of analogy is closely based on the literal meaning of the Greek word αναλογια, meaning 'proportionally'. Kant defined analogies as two pairs of terms, whereas there is a relation between the terms of each pair. If there is the *same* relation between the terms of both pairs (i.e. the pairs are proportional), then the term pairs form an analogy. This very strict understanding of the concept of analogy is often referred to as *identity of relation* model. This classical model, while still frequently used in philosophy, is quite limiting as the relations between the terms of the pair are indeed required to be the same and are not *mutatis mutandis*, i.e. just retaining some aspects and ignoring others. Therefore, usually only analogies within the *same domain* are possible as the more powerful and interesting *cross-domain analogies* do not share the same relation, but similar ones (Juthe 2005).

This problem can be rectified by relaxing the requirement of relational identity to relational similarity respecting the *mutatis mutandis* semantics. The resulting model became to be known as the *4-term analogy model*. For example, consider the following analogy question: "What is to the sky as is a ship to the ocean?" The obvious answer is 'an airplane', as airplanes are used to travel the sky as ships are used to travel the oceans. The actual differences in their physical properties (like the shape, colour, or the material) and their non-prototypical relations to other concepts are ignored. However, the "travels" relation in both term pairs is not the same, as traveling water and traveling air is different in many aspects.

This simple type of analogy is also actively researched in linguistics, because many aspects of language evolution and the development of new words via neologisms are based on such simple analogies (think of the time when the word 'spaceship' appeared for the first time: while nobody knew the word, many aspects of its meaning are still immediately clear). While the example of ship is to ocean as airplane is to sky is quite illustrative, analogical reasoning can quickly become ambiguous when relational similarity between terms is either not strong enough or too many candidate pairs with similar relational strength exist (e.g. consider "What is to land as is a ship to the ocean?"; here, a correct answer is hard to determine without further information, should the answer be a car, a bus, a truck, an ox cart?).

In contrast to the 4-term analogy, early Greek philosophy under the guidance of Plato and Aristotele proposed a much wider and harder-to-grasp notion of analogy based on two concepts having a *shared abstraction* (Shelley 2003). Therefore, an analogy can be formed even without shared relations if two concepts share some common ideas, perceptions, attributes, effects, or functions. This lead to early jurisdictional reasoning in Roman law and is also still used in modern laws, e.g. German law allows for analogical reasoning in its civil code ("Privatrecht"). The notion of analogy by shared abstraction was particularly popular in theological reasoning, often resulting in analogies as "God is like the Sun".

Other approaches see analogies as a variant of formal logics, i.e. analogy is seen as a special case of *induction* (Shelley 2003) or for performing *hidden inductions* (Juthe 2005). For example, if *A* is analogous to *B*, and *A* and *B* are known to share certain properties, analogical induction can claim that most probably also the non-shared properties hold true for both *A* and *B*. As approaches based on formal logics usually rely on consistent knowledge bases (a requirement which we envision can rarely be fulfilled for analogy-enabled information systems which rely on automatically extracted semantics: see later sections), this kind of analogy model is not the best choice for pursuing in information systems research.

Currently, the most popular model for analogies comes from the field of contemporary cognitive sciences and clarifies some of the vague concepts of Aristotle's view on analogies, and is commonly known as the *structure mapping theory* (Gentner 1983). Structure mapping is assuming that knowledge is explicitly provided in form of propositional networks of nodes and predicates and claims that there is an analogy whenever large parts of the structural representation of relationships and properties of one object (the source) can be mapped to the representation of the other object (the target). This model resulted in several theoretical computational models (e.g. (Gentner and Gunn 2001)). The structural mapping theory serves as a very strong starting point for building analogy-enabled information systems, with ontologies and Linked Open Data as potential knowledge bases. These issues will be discussed in more detail in later sections of this paper.

While the structural mapping theory is quite popular, it is also heavily criticized for relying on explicit propositions and ontologies, and claim that analogies should be drawn between objects or situations which are similar on a *high-level of perception* (Chalmers, French, and Hofstadter 1992). While this does not directly contradict structural mapping, high-level perception is definitely a similar aspect of general analogy problems (Morrison and Dietrich 1995). In later sections of this paper we will briefly outline the benefits and synergies a holistic approach covering both structural and perceptual information would have, and how it could be realized with nowadays information system and database technologies.

This view on *perceptual analogies* also aligns with our own observation that analogies are not static, i.e. the 'correctness' of an analogies may be dependent on the context and the knowledge background or even personal opinions of the actual involved in the communication, and usually analogies rely on a *consensual agreement* with respect to the structure and relations chosen for the analogy. Therefore, analogies can easily be misunderstood or even change their meaning during time or between different persons. This observation will play a major role for analogy-enabled information systems when being confronted with user-generated content from the Social Web. As a recent example (at a time shortly before the release of Microsoft Windows 8), consider the statement "Windows 8 is like Windows Vista". In nearly all instances at the current time, when this statement is used in the Social Web, the author implies an analogy (and not an similarity comparison) in which he tries to communicate that he does not like Windows 8 and prematurely considers it to be a failure, as Windows Vista is considered being a failure by many. However, this analogy can usually only be understood by like-minded individuals with similar technological backgrounds and also similar opinions on the topic (or at least knowing about those opinions). Furthermore, 'Windows Vista being a failure' is far from being a fact; it is only a frequently mouthed opinion which reached a certain level of consensus. These issues need to be considered when building an analogy enabled information systems. Fortunately, the Social Web provides are large corpus of opinions which can be minded for capturing perceptional semantics (see later sections).

3.2. Analogy and Similarity

The relation between analogy and similarity is by many people considered to be a confusing one. This is due to the fact that while analogy is not the same as similarity, analogical reasoning heavily relies on various 'flavors' of similarity (Lofi 2013). Therefore, sometimes it is claimed that there is the concept of *generic similarity* (Brown 1989), for which the commonly used *property similarity* (what people usually mean when referring to 'similarity'), is just a special case. Other special cases of generic similarity are analogy in terms of relational similarity (e.g. 4-term analogy) and structural similarity (e.g. complex analogy).

Like analogy, similarity also establishes a certain relation between a source concept and a target concept. In this sense, when assuming knowledge provided in form of prepositional networks as in the structure mapping theory, 'normal' (property) similarity can be defined when most of the attributes/properties and the relations of the source are similar to those of the target (Gentner 1983), e.g. if claiming that the "Kepler-30 star system is like our solar system[1]", this is a similarity statement because both are star systems, both have similar suns, and both show similar planetary trajectories (albeit Kepler-30 has less planets with different properties). In contrast, claiming that "Atoms are like our solar system" is indeed an analogy, as atoms and the solar systems have no similar attributes, but do have similar relations between their related concepts. This difference between analogy and similarity is quite significant, as similarity is a well-researched problem in information systems, with many efficient and mature implementations already published. Furthermore, similarity can be computed much easier as it mostly relies on attribute values, which are usually readily available. In contrast, computing relational similarity is a little researched problem, and also requires vast and diverse semantic knowledge bases which are difficult to obtain. Therefore, relational similarity is an entirely different challenge.

4. Analogies and Information Systems

The intuitive and high-level properties of analogies open promising applications for human-centered interaction with information systems. These range through a wide spectrum from e-commerce, web information retrieval, to open-domain question answering. In this section, we will highlight just a few of these possibilities.

4.1. State-of-the-Art

Unfortunately, despite the great potential of analogies, they rarely received attention by the information systems community (Lofi 2013). Most early knowledge-based systems focused on a *first-principles* style of reasoning, i.e. relying on a set of axioms and rules for inferring the answer of a query via strict logical means. The few exceptions trying to accommodate analogies used them only as fallback solution when strict inference failed, e.g., (Blythe and Veloso 1997; VanLehn and Jones 1993). Some other systems were based on specialized case-based reasoning techniques (Leake 1996), introducing a measure of similarity into reasoning. Only recently, dealing with analogies became a more important topic in the formal AI community: approaches are usually based on ontologies (Forbus, Mostek, and Ferguson 2002) or neural networks (Hummel and Holyoak 2005). Most promising seem to be approaches using natural language processing (NLP), and they have been proven to be very successful in certain areas of analogy processing. These systems rely on interpreting large text collections via NLP and employ statistics (Ishizuka 2010; P. Turney 2008) to extract relationships which can be interpreted as analogies and are especially important for applications on the Web. Often, these systems are evaluated using test sets from the US-based SAT challenges (P. D. Turney et al. 2003), such as (Bollegala, Matsuo, and Ishizuka 2009; Davidov 2008).

While not being very complex analogies, the SAT challenges deserve special attention due to their importance as a testbed for evaluation algorithms and heuristics. The SAT test is a standardized test for general college admissions in the United States. It features major sections on analogy challenges

---

[1] http://web.mit.edu/newsoffice/2012/far-off-solar-system-0725.html

to assess the prospective student's general analytical skills. These challenges are expressed as 4-term analogy problems (but with multiple choices answers): out of a choice of five word pairs, one pair has to be found which is analogous to a given word pair. As an example, consider this challenge:

legend is to *map* as is a) *subtitle* to *translation* b) *bar* to *graph*
c) *figure* to *blueprint* d) *key* to *chart* e) *footnote* to *information*.

Here, the correct answer is d) as a key helps to interpret the symbols in a chart as does the legend with the symbols of a map. While it is easy to see that this answer is correct when the solution is provided, actually solving these challenges seems to be a quite difficult task for aspiring high school students as the correctness rates of the analogy section of SAT tests is usually reported to be around 57%.

Another reason for the popularity of research aimed at the SAT test is that solving these challenges is significantly easier than dealing with general analogies: basically, the relational similarity for each of the answer choices to the source term pair is computed, and the most similar one is picked. But still, this type of system is a valuable building block for future analogy-enabled information system.

4.2. Information Systems for E-Commerce

In recent years, the amount of products available on online e-commerce platforms has increased tremendously. Modern online platforms enable customers to buy or rent a vast selection of mainstream as well as long-tail titles, far beyond the capabilities brick and mortar stores. Due to this vast selection of titles and the convenient shopping experience compared to physical shops, these platforms enjoy an ever increasing popularity. However, this freedom comes at a price: users are often overwhelmed by the sheer amount of offers and have a hard time deciding on a certain product (Homoceanu et al. 2011). This problem is strongly aggravated for users who are not particularly knowledgeable in the domain of the product they aim at purchasing, especially if they do not know desirable specifications, manufacturers, or product properties. In this case, high-level verbal communication with the sales staff is the most effective solution. For non-expert customers, this communication usually relies on analogies for describing the desired product; likewise, the use of analogies allows the experts to easily illustrate complex domain knowledge without the use of difficult-to-understand domain language.

For example, consider explaining to somebody agnostic to the late developments of consumer electronics what an Apple iPod is. This task can easily be achieved by stating "For the last few years the iPod did the same, as the Walkman did in the 80s." Assuming that the person knows about the Sony Walkman, this short statement conveys efficiently a lot of information: the Walkman was not just a portable device for playing music, but was also a fashionable lifestyle product with high prestige among younger people, and which also changed how music was consumed and purchased. Similarly, analogies can be used for queries: consider for example the statement "I am looking for the Ferrari of home-entertainment systems." This does not only transport the information that a high-end system in terms of quality is desirable, but also carries notions of stylish design and allows for including high-price product segments (however, it does not imply that one looks for a red home-entertainment system although many people connect the color red to Ferrari's – here, capturing the common understanding, i.e. focusing on the defining relations of an analogy is of high importance). Thus, by using such analogies, even complex queries can be efficiently stated without exact a-priori knowledge of the product properties as required by traditional interaction approaches.

Unfortunately, most e-commerce sites only have access to tabular product-metadata, usually stored in relational databases. While this data can be used to realize similarity queries (Lofi, Nieke, and Balke 2010) (e.g. "I am looking for something like an iPhone", which might result in a Samsung Galaxy S mobile phone), mining analogies from this kind of data is rarely possible as there is no information on the relation between products, attributes, use cases, companies, or other related concepts. Especially analogies on high perceptual level are complex and are heavily relying on additional data. Furthermore, the correctness of these analogies might be very context sensitive and individual. For example, the analogy "the iPad is the new iPod" might be correct in the context of hyped lifestyle technology gadgets, but an iPad will be a bad choice if one is just looking for a simple way to play music on the go. These issues opens up several interesting challenges like the question of perceived similarity, prototypes, typicality of characteristics, and different representations with respect to context, and will be highlighted in the next section.

4.3. Question Answering

Since more than one and a half decades, web-based information retrieval heavily relies on simple keyword searches. While most people nowadays successfully adapted to this simplistic interaction paradigm, it is far from being natural. In a natural dialogue, people would ask questions in order to obtain the information they need; in contrast, information retrieval usually only provides documents containing a keyword which can often lead to unsuitable documents, requiring careful consideration of the query results and continuous query refinement. In contrast, Question Answering directly aims at answering questions given in natural language. Functional question answering systems could therefore prove to be a very potent tool for natural and human-centric interaction with future Web and information systems.

Similar to analogy queries, this highly complex problem relies on huge data repositories and careful analysis of semantic data on entities and their relations obtained from ontologies, LOD, or the Web. While research in question answering started already decades ago, only recently significant progress could be made by the IBM Watson team (Ferrucci, David; Brown, Eric; Chu-Carroll et al. 2010), which consequently lead to an re-ignition of interest in the field by both academia and the general public.

One of the main features of most question answering systems is that they require special handling of each individual question type, i.e. questions for a geographical location are treated differently than questions for a date or an explanation. To our knowledge, no current question answering system dealt with the challenge of analogy questions yet. However, in order to provide a true natural language interface which provides similar question answering capabilities as humans can offer, analogy queries are also an important issue for future question answering systems.

5. Designing Analogy-Enabled Information Systems

In this section, we introduce a generic design for analogy-enabled information systems. This covers two major issues: a) the general architecture of such an information system b) and designing a suitable language for representing analogical knowledge and queries.

5.1. General System Design

In the following, we will present the major components of an analogy-enabled information system. Each of these components poses interesting and demanding research challenges from disciplines like information systems, natural language processing, knowledge engineering, or databases. In the presented architecture, the user basically provides a query as a simple analogy program, which is processed using vast knowledge repositories, and obtains an analogy program as a result. The resulting program will not only contain the solution, but will also contain some basic explanation of why the solution matches the query (see Figure 1).

In short, an analogy can be seen as a perceived prototypical relational similarity between some concepts / pairs of concepts. In any case, in order to allow for meaningful analogy queries in information systems, additional data sources need to be made accessible in addition to the raw product specifications. This data needs not only to focus on the products in the e-commerce system itself, but also have to focus on closely related concepts and should contain most relevant relations. One possible source of such data is provided by Linked Open Data (LOD) repositories, which contain large numbers of relational and inter-referable information. LOD especially covers common knowledge and can be contain sources like DBpedia (Auer et al. 2007), or YAGO (Suchanek, Kasneci, and Weikum 2007). Also, open dictionaries like WordNet or SentiWordNet (Baccianella, Esuli, and Sebastiani 2008) can prove to be valuable data sources.

However, for some applications, it will be necessary to generate specialized knowledge repositories. For example, consider e-commerce applications: here, detailed knowledge with respect the sold products is required. Often, this knowledge has to be specifically mined for this task from suitable sources (like for example expert blogs, discussion forums, online magazines, etc.) Therefore, *knowledge mining* using various techniques (e.g., relation extraction, natural language processing, and sentiment analysis) will also be a significant task for an analogy system. When mining the Social Web (in contrast to using factual sources like DBpedia or mining Wikipedia), often opinionated, personal, or non-factual knowledge is obtained. While knowledge mined from these sources needs to be handled with care and requires special treatment (e.g. majority votes, sentiment analysis), it provides the opportunity to also respect perceptual qualities which is often required for high-level analogies (see previous sections).

In addition to the knowledge repository, there is also a repository storing explicit analogical information (i.e. known analogies and their justifications). This repository can be provided from external sources, or can be generated during the system's runtime by archiving discovered analogies for later use.

The central component in our intended architecture is the analogy processing itself. The complexity of this component can vary tremendously with the expressiveness of the analogies which the system is expected to process. Simple analogies, e.g. simple 4-term analogy problems can be approached quite easily (Bollegala, Matsuo, and Ishizuka 2009), while complex analogies across different domains respecting high-level perceptions and opinionated consensus require significantly more effort.

In short, four major tasks for the analogical processing can be identified:

- *Entity Resolution* for aligning the entities in the various knowledge repositories
- *Entity Similarity Computation*: While analogies are usually defined by relational similarity, the similarity between entities is also sometimes required for analogy processing, e.g. when dealing with same-domain analogies, or as part of analogical reasoning in a structural-mapping fashion. For purely tabular data in metric spaces, very efficient algorithms are available, e.g. (Hjaltason and Samet 2003). However, there are also approaches which aim at capturing perceived similarity by not only using attribute values, but also incorporating the analysis of Social Web data in order to capture perceptual properties (e.g., (Lee et al. 2010) which relies on term co-occurrence). These perceptual approaches already implicitly incorporate some relational aspects, i.e. frequently co-occurring terms indicate that those terms are somehow related, either because they are highly similar with respect to their attributes, or they are just perceived as being closely related, which often means that they are similar with respect to their functions or relations. Therefore, this type of similarity already bridges into relational similarity, and can be used as simple heuristic for emulating some aspects of a real relational analysis.
- *Relational Similarity Computation*: This task is at the very core of analogical processing, as being able to reliably compute the similarity of two relations directly solves many 4-term analogy problems, but is also the central building block of more complex analogy models like structure mapping. In contrast to well-research property/attribute similarity, computing relational similarity relies on the analysis of sematic relational data or language patterns. Therefore, it is significantly more complex, and unfortunately also less well researched. Furthermore, relational similarity is context dependent and not necessarily symmetric. An heuristic for approaching this problem is for example the distributional hypothesis (Harris 1954) from linguistics, which claims that words frequently occurring in the same context also have similar meanings. An implementation of this heuristic can be found in (Bollegala, Matsuo, and Ishizuka 2009). Another heuristic approach relying on pattern extraction and set inclusion is given in (Nakashole, Weikum, and Suchanek 2012). However, while been proven effective to a certain degree, these approaches all rely on simple heuristics for estimating relational similarity. Therefore, the challenge of efficiently and reliably computing relational similarity is probably still the most pressing issue for future analogy-enabled information systems.
- *Prototype Analysis*: The task of prototype analysis is to determine which of the multitude of attributes and relations belonging to a certain entity are actually relevant (or *prototypical*) from a perceptual point of view. This knowledge is highly important as analogies involving non-prototypical relations are rarely understood by people. Unfortunately, prototype analysis is a comparably new field of research, and only few preliminary works exist. Here, the focus can either be on identifying 'typical' attributes for a set of entities (Selke, Homoceanu, and Balke 2012), and by this finding suitable prototypes which serve as a baseline for comparing and judging all related

entities, or it can be on identifying relevant relations for entities, and thus sorting out the unimportant ones, e.g. (Homoceanu and Balke 2012).

Analogical Inference: This component is actually responsible for solving an analogy problem. Basic solutions for some specialized cases have already been presented in (Gentner and Gunn 2001) and (Bollegala, Matsuo, and Ishizuka 2009) (for 4-term analogies: basically, the relational similarity between the source and all possible targets is computed and the target with the highest similarity is returned), but still, this challenging component will requires most of the future research efforts.

5.2. analogy^E: Analogy Programs

In this section, we introduce our analogy^E language for analogy programs. The purpose of this language is three-fold:

- *Representing analogical knowledge*: analogy^E can be used to represent and store analogies, e.g. for archival, documentation, or within the analogy repository of our architecture.
- *Posing analogy queries*: analogy^E allows for formulating queries for the most common analogy problems.
- *Explaining query results*: After an analogy query is processed, the query result can be encoded in analogy^E, including a further justification and explanation.

The design of the language has been inspired by the insights gathered in the structural-mapping theory (Gentner 1983), and it loosely lends some aspects of frame languages (Minsky 1975). analogy^E's primary design goal is to serve as a *representation* langue for both analogies and queries. There are no formal semantics attached to the language itself (as for example are with Datalog), and implementing these semantics is the responsibility of the system the language is used with. Therefore, its actual behavior and returned results may differ among systems. Furthermore, due to space restrictions in this paper, the basic constructs of the language are only briefly explained, and we rely on the examples later in this section to show how they interact.

The basic constructs of the language are *entities* (all constructs are listed in Table 1), which represent a real world entity or concept (please note: for the sake of simplicity, we do not distinguish between classes and instances as, for example, OWL does. Therefore, the entities *star system* and *solar system* are treated in the same way, although the solar system is an instance of a star system. The rationale behind this decision is that this allows for simpler statistical processing of analogies, and also pays respect to the fact that many automatically generated knowledge repositories also have problems with this distinction. Furthermore, there are few analogies which actually refer to instances (as e.g., the Rutherford analogy does); most analogies will deal only with classes anyway. Therefore, we refer the classes and instances collectively as *entities*.)

For each entity, *properties* (i.e. attributes) can be stored. These can either be literal properties, e.g., as stored in many product databases, or prototypical properties. The latter encodes a property in relation to its context, i.e. relative to the corresponding prototype. For example, one could say that the size of the Earth is 6,300 km in radius (literal property), or that it is medium sized (prototypical property within the reference frame of our solar system). Prototypical properties are particularly useful for analogies as people also intuitively embed properties in their context when forming analogies (e.g., consider the Rutherford analogy: the sun/nucleus is massive and huge, while planets/electrons are tiny and of miniscule mass). However, establishing the correct reference frame is still a challenging research question (see prototype analysis in previous section 5.1).

Relations between entities are named and only considered in their binary form, i.e. between two entities (somewhat similar to RDF triples). Furthermore, there is a special symbol for indicating subclass relationships, helping with more complex analogies. In addition to those language concepts which actually encode facts about entities and their relations, there are language constructs representing the results of analogical processing. First, there is the notion of two entities being *closely related* ($A_1 \overset{\circ}{\sim} A_2$), i.e. there are some prototypical and semantically important relation between those two entities. In a similar fashion, there is the notion of two entities being similar on a property level ($A_1 \approx A_2$). These similarity statements are especially useful when explaining analogies.

At last, analogy^E of course allows for analogy statements, generally denoted by the symbol :: (see Table 1). These come in three flavors, from very generic to quite specific. The first type is general analogy between two entities (e.g. the $atom$ is like the $solar\ system$), the second is analogy between two entity pairs (e.g. $ships$ and the $ocean$ are like $airplanes$ and the sky), and the last one is specific analogy between pairs (e.g. with respect to their $movement$ patterns, $planets$ are to the sun as $electrons$ are to the $nucleus$). The special feature of analogy^E now is that for similarity or analogy statements, also some justifications can be provided. These justifications are simply nested hierarchically after an analogy statement in rectangle brackets.

As an example, consider the following excerpt of the Rutherford analogy. This example could also easily be extended to also contain the attracting and repulsive forces.

$Atom\ ::\ Solar\ System$ [
$\quad \{Nucleus, Electron\} \overset{\circ}{\sim}\ Atom$
$\quad \{Sun, Planet\} \overset{\circ}{\sim}\ SolarSystem$
$\quad \sim(Nucleus, Electron) :: \sim(Sun, Planet)$
$\quad distance(Nucleus, Electron) :: distance(Sun, Planet)$
$\quad massRelation(Nucleus, Electron) :: massRelation(Sun, Planet)$
$\quad revolvesAround(Electron, Nucleus) :: revolvesAround(Planet, Sun)$
]

This analogy program read as: an atom is like the solar system, because the nucleus and the electrons (which are both central concepts for atoms) behave like the sun and planets. More specifically, they behave similar with respect to the relative distances, their relative mass, and their relative movement.

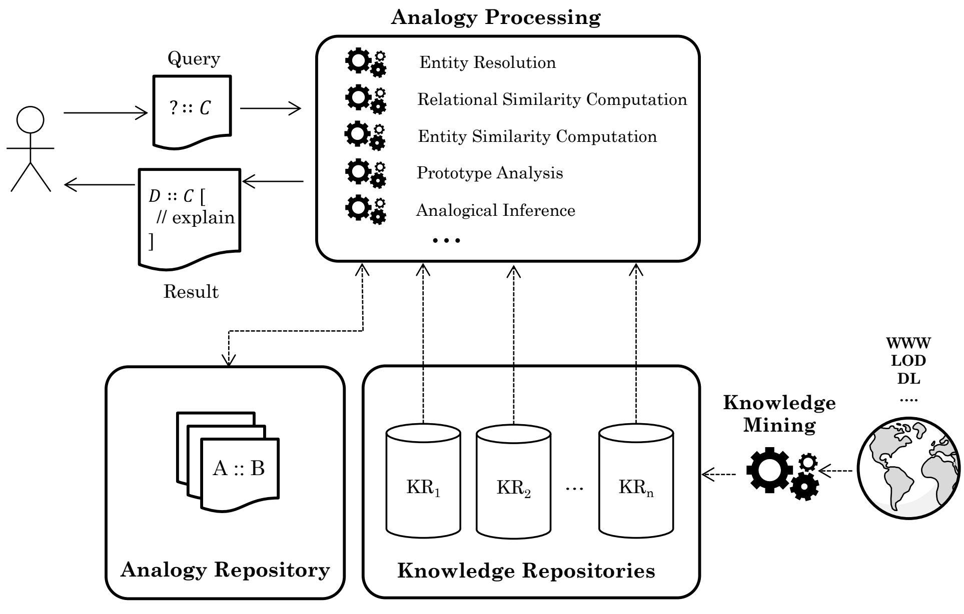

Figure 1. Generic Architecture for Analogy-Enabled Information Systems

Table 1: analogy^E constructs for encoding analogical knowledge

Concept	*Notation*	*Examples*
Entity	$A_1, A_2, \dots, B_i, \dots$	$Battery, Pump, Wire, Pipe,$ $Star, Planet, \dots$
Literal Property	$p(A_1, literal)$	$shape(Battery, circular)$ $diameter(AAA\ Battery, 10\ mm)$
Prototypical Property	$p(A_1), p(A_1, variable)$	$diameter(AAA\ Battery, small)$
Relation	$r(A_1, A_2)$	$provides(Battery, Voltage)$ $revolves(Planet, Star)$
Subclass Of	$A_1 \sqsubset A_2$	$AAA\ Battery \sqsubset Battery$ $Canoe \sqsubset Ship$
Closely Related (Relations)	$A_1 \mathring{\sim} A_2$	$Electrical\ Circuit \mathring{\sim} Wire$ $Star\ System \mathring{\sim} Planet$
Similar (Properties)	$A_1 \approx A_2$	$Solar\ System \approx X12$
Analog to (Entity Level)	$A_1 :: A_2$	$Electrical\ Circuit :: Hydraulic\ Circuit$ $Star\ System :: Atom$
Analog To (Entity Pair Level)	$\sim(A_1, A_2) :: \sim(B_1, B_2)$	$\sim(Planet, Star) :: \sim(Electron, Nucleus)$ $\sim(Bible, Christianity) :: \sim(Quran, Islam)$
Analog To (Relation Pair Level)	$r_A(A_1, A_2) :: r_B(B_1, B_2)$	$revolves(Planet, Star) ::$ $revolves(Electron, Nucleus)$ $restricts(Resistor, Current)$ $:: restricts(ConstrictedPipe, Flow)$

Please note that analogy programs are not intended to be complete, as they are purely representative in their nature. All complete information required to perform analogy processing is available in the union of the program and the knowledge base. In this context, analogy programs are intended to just highlight and summarize some of the implicit information of the knowledge base which is important for the current analogy (e.g. one might easily leave out that sun and planets are closely related to the solar system, but by providing this information, a certain emphasize on this fact is given which may be exploited by the analogy processor and helps humans to understand the program). To conclude this section, consider a more complex example encoding the drain pipe analogy:

```
ElectricCircuit :: HydraulicCircuit [
    // note: we will use EC and HC in the following for the two circuit types.
    {Wire, Electron, Voltage, Current, Potential, Charge} ˚~ EC
    {Voltage Source, Current Source, Switch, Resistor, Capacitor, Diode} ˚~ EC

    {Pipe, Water, Pressure Difference, Flow Rate, Potential, Volume} ˚~ HC
    {Dynamic Pump, Positive Displacement Pump, Valve, Restricted Pipe} ˚~ HC
    {Membrane Diaphragma, OneWay Valve} ˚~ HC
    ...
    // some basic analogy statements without further explanation
    ~(Voltage, EC) :: ~(Pressure Difference, HC)
    ~(Current, EC) :: ~(Water Flow, HC)
    ...
    // example analogy statement between significantly related entities: diode – check valve
    ~(Diode, EC) :: ~(Check Valve, HC) [
        oneWayFlow(Diode, Current) :: oneWayFlow(Check Valve, Water Flow)
        oneWayLeak(Diode, Current) :: oneWayLeak(Check Valve, Water Flow)
        conductiveWhen(Diode, Bias Voltage) :: opens(Check Valve, Opening Pressure)
        breaksOnHigh(Diode, Voltage) :: isBrokenByHigh(Check Vavle, Pressure)

    ]
]
```

Table 2: Results of analogous cars study

$? :: \sim(Volkswagen, Golf)$	$? :: \sim(Skoda, Fabia)$
$\sim(Opel, Astra)$	$\sim(Renault, Clio)$
$\sim(Ford, Focus)$	$\sim(Seat, Ibiza)$
$\sim(Audi, A3)$	$\sim(Ford, Fiesta)$
$\sim(Alfa\ Romeo, 147)$	$\sim(Fiat, 500)$
$\sim(Fiat, Bravo)$	$\sim(Opel, Corsa)$

$? :: \sim(Opel, Astra)$	$? :: \sim(Suzuki, Swift)$
$\sim(Volkswagen, Golf)$	$\sim(Volkswagen, Polo)$
$\sim(Ford, Focus)$	$\sim(Skoda, Fabia)$
$\sim(Mazda, 3)$	$\sim(Citroen, C1)$
$\sim(Toyota, Corolla)$	$\sim(Daihatsu, Cuore)$
$\sim(Alfa\ Romeo, 147)$	$\sim(Ford, Fiesta)$

5.3. Queries in analogy^E

Queries in analogy^E are simple analogy programs, where just some part of a statement is replaced by a question mark. Further hints or restrictions can be provided by nesting them under the statement. Consider the following common cases as an example:

1-of-2 term query

Pattern: $? :: A$
Example: $? :: Atom$ *Results in:* $Solar\ System$

1-of-4 term query

Pattern: $\sim(A_1, A_2) :: \sim(?, B_2)$
Example: $\sim(Ship, Ocean) :: (?, Sky)$ *Results in:* $Airplane$

1-pair query

Pattern: $? :: \sim(B_1, B_2)$
Example: $? :: \sim(Airplane, Sky)$ *Results in:* $\sim(Ship, Ocean)$

SAT-style multiple choice 1-pair query

Pattern: $\sim(A_1, A_2) :: ?\ [\sim(B_1, B_2);\ \sim(C_1, C_2);\ \sim(D_1, D_2);\ ]$
Example: $\sim(Airplane, Sky) :: ?\ [\sim(Bicycle, Road);\ \sim(Ship, Ocean);\ \sim(Car, Driver);]$
Results in: $\sim(Ship, Ocean)$

6. Preliminary Study

In this section, we present a brief pre-study employing some simpler techniques in order to heuristically perform analogical processing. The task of this study was to discover analogies for an e-commerce system focusing on used car sales. Here, only intra-domain analogies are considered, and the aim was for finding analogies of the form "car X of manufacturer M_X is like car Y of manufacturer M_Y". However, the 'like' comparison should not be based on property similarity, but on a relational similarity (i.e. the cars are intended for similar purpose, are in the market segment, etc.) These analogies are intended to allow for semantically meaningful navigation between car models and thus enable easier exploration of the dataset (e.g., if a user finds a car which he likes, but he is still not sure if it represents a good choice, he can navigate to relationally similar cars).

The knowledgebase for this evaluation was automatically generated by crawling and processing discussion threads in internet forums, namely 100 threads for each of the websites *gutefrage.net*, *carpassion.com*, and *autoplenum.de*. All selected discussion threads deal with purchase recommendations, i.e. one user asks what car to purchase for some given requirements, and other users suggest and discuss suitable models. For processing the source material, some initial natural language processing was performed. Especially, the forum posts where analysed and annotated by using the Stanford part-of-speech tagger [34] in order to identify potential entities. Then relation extraction techniques relying on entity co-occurrence in paragraphs were used, resulting in a set of shallow semantics relations (i.e. the resulting knowledge base contains that some entities are closely related to others, but without information on the exact relations, e.g., $Golf \overset{\circ}{\sim} Volkswagen$, or $Golf \overset{\circ}{\sim} Astra$). Furthermore, for each shallow relation, a weight was computed based on the strength of co-occurrence of the entity pairs across multiple posts of different authors, e.g., the weight was increased if certain entity pairs occurred frequently in different context and by different users. In addition, there was a list of all cars and car manufacturers on the market. Using this knowledge base, the analogy processing was performed using following heuristic: cars are considered analogous if they have a close relation to other cars of a different manufacturer, and the weight of that close relation is over a manually defined threshold.

The ranked results obtained for four different example queries are presented in in Table 2. Judging the ‘correctness’ of these results is difficult task, as the quality of the results may be highly subjective. Also, our simple heuristic approach assumed relational similarity if people where discussing two car models in the same paragraph without deeper semantic analysis. But still, when considering the presented results and our additional results obtained during the study, it is visible that most identified pairs for a given query belong to the same category (e.g. compact cars, sub-compact cars, sports cars, etc.). Belonging to the same category therefore seems to be the dominant relation which is captured by our results. The few exceptions from this observation come from (frequently reoccurring) statements where users recommended investing/saving some money, and climbing up/down one vehicle category instead of buying one of the suggested cars. While the workflow used in this evaluation might seem basic, the results provide already a substantial benefit for navigating the product space and indicate high potential for similar, more elaborate techniques in later works and also additional domains.

7. Summary and Outlook

In this paper, we provided discussions for supporting the further development of analogy-enabled information systems. Analogies for information systems are a central and significant technique towards future human-centric interaction paradigms, and allow for efficient and natural communication of abstract high-level information. In detail, we covered an overview of relevant literature covering different formal models and definitions for analogies, and discussed some potential use cases of analogies in information systems and their respective challenge. Furthermore, we introduced analogy^{E}, a language for expressing analogies and analogy queries alongside a generic reference architecture for analogy-enabled information systems. In our future works, the various challenges outlined by this paper will be approached and technical solutions will be developed and evaluated in order to realize the vision of analogy-enabled information system on the long term.

Funding Acknowledgements

This work was supported by funds provided by the DAAD FIT programme (http://www.daad.de/fit/).

References

Auer, Sören, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. “DBpedia: A Nucleus for a Web of Open Data.” *The Semantic Web* 4825: 722–735.

Baccianella, Stefano, Andrea Esuli, and Fabrizio Sebastiani. 2008. “SentiWordNet 3.0 : An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining.” *Int. Conf. on Language Resources and Evaluation (LREC10)* 0: 2200–2204. http://www.lrec-conf.org/proceedings/lrec2010/pdf/769_Paper.pdf.

Blythe, J., and M. Veloso. 1997. "Analogical Replay for Efficient Conditional Planning." In *Nat. Conf. on Artificial Intelligence (AAAI)*. Providence, Rhode Island, USA.

Bollegala, Danushka T., Yutaka Matsuo, and Mitsuru Ishizuka. 2009. "Measuring the Similarity Between Implicit Semantic Relations from the Web." In *18th Int. Conf. on World Wide Web (WWW)*. Madrid, Spain. doi:10.1145/1526709.1526797. http://portal.acm.org/citation.cfm?id=1526709.1526797.

Brown, W. R. 1989. "Two Traditions of Analogy." *Informal Logic* 11 (3): 160–172.

Chalmers, David J., Robert M. French, and Douglas R. Hofstadter. 1992. "High-level Perception, Representation, and Analogy: A Critique of Artificial Intelligence Methodology." *Journal of Experimental & Theoretical Artificial Intelligence* 4 (3) (July 1): 185–211. doi:10.1080/09528139208953747. http://www.informaworld.com/openurl?genre=article&doi=10.1080/09528139208953747&magic=crossref||D404A21C5BB053405B1A640AFFD44AE3.

Davidov, Dmitry. 2008. "Unsupervised Discovery of Generic Relationships Using Pattern Clusters and Its Evaluation by Automatically Generated SAT Analogy Questions." In *Ass. for Computational Linguistics: Human Language Technologies (ACL:HLT)*. Columbus, Ohio, USA.

Ferrucci, David; Brown, Eric; Chu-Carroll, Jennifer;, Aditya A.; Fan, James; Gondek, David; Kalyanpur, John; Lally, Adam; Murdock, J. William ; Nyberg, Eric; Prager, Chris Schlaefer, Nico; Welty, and John; Lally, Adam; Murdock, J. William; Nyberg, Eric; Prager. 2010. "Building Watson: An Overview of the DeepQA Project." *AI Magazine* 31 (3): 59–79. http://www.aaai.org/ojs/index.php/aimagazine/article/viewArticle/2303.

Forbus, Kenneth D., Thomas Mostek, and Ron Ferguson. 2002. "Analogy Ontology for Integrating Analogical Processing and First-principles Reasoning." In *Nat. Conf. on Artificial Intelligence (AAAI)*. Edmonton, Alberta, Canada.

Gentner, D. 1983. "Structure-mapping: A Theoretical Framework for Analogy." *Cognitive Science* 7: 155–170. http://www.mendeley.com/research/no-title-avail/.

Gentner, D, and V Gunn. 2001. "Structural Alignment Facilitates the Noticing of Differences." *Memory & Cognition* 29 (4) (June): 565–77. http://www.ncbi.nlm.nih.gov/pubmed/11504005.

Gentner, D. 2003. "Why We're So Smart." In *Language in Mind: Advances in the Study of Language and Thought*, 195–235. MIT Press.

Gentner, D., and A. B. Markman. 1997. "Structure Mapping in Analogy and Similarity." *American Psychologist* 52: 45–56.

Harris, Z. 1954. "Distributional Structure." *Word* 10: 146–162.

Hjaltason, Gisli R., and Hanan Samet. 2003. "Index-driven Similarity Search in Metric Spaces." *ACM Transactions on Database Systems (TODS)* 28 (4): 517 – 580.

Hofstadter, Douglas R. 2001. "Analogy as the Core of Cognition." In *The Analogical Mind*, 499–538.

Homoceanu, Silviu, and Wolf-Tilo Balke. 2012. "Chips Off the Old Block – Focused Structure Discovery for Web Entities Based on Family Resemblance." In *Technical Report on Http://www.ifis.cs.tu-bs.de/staff/silviu-homoceanu*. http://www.ifis.cs.tu-bs.de/staff/silviu-homoceanu.

Homoceanu, Silviu, Michael Loster, Christoph Lofi, and Wolf-tilo Balke. 2011. "Will I Like It ? – Providing Product Overviews Based on Opinion Excerpts." In *IEEE Conference on Commerce and Enterprise Computing (CEC)*. Luxembourg, Luxembourg.

Hummel, John E., and Keith J. Holyoak. 2005. “Relational Reasoning in a Neurally Plausible Cognitive Architecture. An Overview of the LISA Project.” *Current Directions in Psychological Science* 14 (3) (June): 153–157. doi:10.1111/j.0963-7214.2005.00350.x. http://cdp.sagepub.com/lookup/doi/10.1111/j.0963-7214.2005.00350.x.

Ishizuka, Mitsuru. 2010. “Exploiting Macro and Micro Relations Toward Web Intelligence.” In *11th Pacific Rim Int. Conf. on Trends in Artificial Intelligence (PRICAI)*. Daegu, Korea. http://portal.acm.org/citation.cfm?id=1884293.1884296.

Juthe, A. 2005. “Argument by Analogy.” *Argumentation* 19 (1): 1–27.

Kant, Immanuel. 1790. *Critique of Judgement*.

Leake, D. 1996. *Case-Based Reasoning: Experiences, Lessons, and Future Directions*. MIT Press.

Lee, J., S.-won Hwang, Z. Nie, and J.-R. Wen. 2010. “Product EntityCube: A Recommendation and Navigation System for Product Search.” In *26th IEEE International Conference on Data Engineering (ICDE)*. Long Beach, California, USA.

Lofi, Christoph. 2013. “Just Ask a Human? – Controlling Quality in Relational Similarity and Analogy Processing Using the Crowd.” In *CDIM Workshop at Database Systems for Business Technology and Web (BTW)*. Magdeburg, Germany.

Lofi, Christoph, Christian Nieke, and Wolf-Tilo Balke. 2010. “Mobile Product Browsing Using Bayesian Retrieval.” In *12th IEEE Conference on Commerce and Enterprise Computing (CEC)*. Shanghai, China.

Minsky, Marvin. 1975. “A Framework for Representing Knowledge.” In *The Psychology of Computer Vision*, 211–277.

Morrison, CT, and E Dietrich. 1995. “Structure-Mapping Vs. High-level Perception: The Mistaken Fight Over The Explanation of Analogy.” In *Conf. of the Cognitive Science Society*. Pittsburgh, Pennsylvania, USA.

Nakashole, Ndapandula, Gerhard Weikum, and Fabian Suchanek. 2012. “PATTY: A Taxonomy of Relational Patterns with Semantic Types.” In *Int. Conf. on Empirical Methods in Natural Language Processing (EMNLP 2012)*. Jeju Island, Korea.

Selke, Joachim, Silviu Homoceanu, and Wolf-Tilo Balke. 2012. “Conceptual Views for Entity-Centric Search: Turning Data into Meaningful Concepts.” *Computer Science: Research and Development* 27 (1): 65–79.

Shelley, Cameron. 2003. *Multiple Analogies In Science And Philosoph*. John Benjamins Pub Co.

Suchanek, Fabian M., Gjergji Kasneci, and Gerhard Weikum. 2007. “Yago: a Core of Semantic Knowledge.” In *16th International Conference on World Wide Web (WWW)*. Banf, Alberta, Canada. doi:10.1145/1242572.1242667. http://portal.acm.org/citation.cfm?id=1242572.1242667.

Turney, Peter. 2008. “A Uniform Approach to Analogies, Synonyms, Antonyms, and Associations.” In *22th Int. Conf. on Computational Linguistics (COLING)*. Beijing, China. http://nparc.cisti-icist.nrc-cnrc.gc.ca/npsi/ctrl?action=shwart&index=an&req=5764174&lang=en.

Turney, Peter D, Michael L Littman, Jeffrey Bigham, and Victor Shnayder. 2003. “Combining Independent Modules to Solve Multiple-choice Synonym and Analogy Problems.” In *Int. Conf. on Recent Advances in Natural Language Processing (RANLP)*. Vol. cs.CL/0309. http://arxiv.org/abs/cs/0309035.

VanLehn, K., and R. M. Jones. 1993. “Integration of Analogical Search Control and Explanation-based Learning of Correctness.” In *Machine Learning Methods for Planning*, 273ff.