CALCULATING SEMANTIC SIMILARITY BETWEEN FACTS

Sergey Afonin and Denis Golomazov
Institute of Mechanics, Moscow State University, Michurinskij av. 1, Moscow, Russia

Keywords: Semantic similarity, Facts, Events.

Abstract: The present paper is devoted to the calculation of semantic similarity between facts or events. A fact is considered as a single natural sentence including three parts, "what happened", "where" and "when". Possible types of mismatches between facts are discussed and a function calculating the semantic similarity is proposed. Very preliminary experimental results are presented.

1 INTRODUCTION

People use search engines to find information on any subject. Sometimes they search for facts. A fact is something that has happened in a certain place at a certain time. Why do people need information on facts? If they just want to check news they use news portals, not search engines. That's true, but every once in awhile people want to find additional information about a fact. For example, they have heard from a friend about some fact and want to read more about it. Or it can be a journalist creating a dossier for a person and they want to check if some rumours are true facts. Moreover, fact search provides a basis for task of revealing relations between facts and other data mining problems.

The problem is that fact search is a type of search that search engines currently cannot always handle properly. Let us consider some local news, for example, bank robbery in Livermore, California in July 2008. A user wants to find some additional information on the fact. Suppose they forgot the name of the city and ask Google with the query *California bank robbery in July 2008*. No relevant results are on the first page. The same situation occurs if we ask Google with the query *Livermore bank burglary in July 2008*. This probably happens due to the not great importance of the fact, and low page rank of the portal on which the news had been posted. The search engine ranks the page describing the event relatively low since there was not exact matching, and there were a lot of robberies in California during that period. Another problem of modern search engines is the inability to perform almost any analytics. For example, we can not get neither a list of all events in some city for some period, nor a list of cities in which bank robberies occured in July 2008, nor a list of dates on which robberies in California took place.

Fact search appears to be a more complicated task than ordinary keyword search. The main reason is that a fact can be represented in many ways, using synonyms, abbreviations, with some keywords included or omitted. The second reason is that a fact can be inferred basing on information distributed over sentences or even documents.

Let us discuss a virtual system that performs fact search. It operates as follows. First, it crawls the web and extracts facts from web pages. Second, it lets a user enter a sentence describing a fact and returns semantically similar facts from the database. During this process it somehow calculates the *trust rate* of the fact, i.e. how likely is the fact to be true. To construct such a system, we divide the fact search task into four steps.

- Extraction of facts from large number of texts in natural language.
- Calculating semantic similarity between facts.
- Calculating the trust rate of a fact.
- Efficiently perform similar facts search on a large database. For example, this can include development of index structures for facts.

In this paper we focus on the task of calculating semantic similarity between facts. We believe this task to be the cornerstone of the fact search problem. For example, the similar facts search task can be easily, though not efficiently solved with the help of a semantic similarity function and linear search. Having a semantic similarity function for facts, one can

Afonin S. and Golomazov D..
CALCULATING SEMANTIC SIMILARITY BETWEEN FACTS.
DOI: 10.5220/0003118705140517
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2010), pages 514-517
ISBN: 978-989-8425-28-7

apply it for such common tasks as fact classification and clustering.

The problem of semantic similarity calculation is a wide one. The objects to calculate similarity between can be single words (Bollegala et al., 2009), groups of words (Varelas et al., 2005), sentences or texts (Islam and Inkpen, 2008), (Li et al., 2006). In this paper we consider sentences of a special kind, namely that describe facts. We have not found any papers devoted to this particular problem, though some general algorithms mentioned above can be applied and there exist works on event description detection and classification, e.g. (Naughton et al., 2010).

2 SEMANTIC SIMILARITY BETWEEN FACTS

We consider facts consisting of three parts: *what* happened, *where* and *when*, so a fact F is a triple $F = (what, where, when)$. Our goal is a function $S(F_1, F_2)$ that calculates semantic similarity between facts F_1 and F_2. The function S takes values between 0 and 1, and higher value means higher similarity. In this section we discuss properties that such function should satisfy.

First of all, let us note that two facts should be treated similar if all their components are pairwise similar. It seems unlikely that there exists a "universal" semantic similarity function suitable for all three parts, so three separate functions s_t, s_r, and s_n measuring semantic similarity of what, where and when parts, respectively, should be defined. The fact similarity fuinction S should use values of these three functions. Note that functions s_t, s_r, and s_n should generally use all components of the compared facts.

As we assume that one of the facts, say F_1, is a user's query and that this query describes the same fact as F_2, but using different lexical means, we classify possible reasons for facts description mismatching.

Synonymy, Acronyms, Abbreviations etc. Two facts may be described differently due to synonymy, acronyms, abbreviations, or slang. For example, *A theft in X bank* describes the same fact as *A larceny in X bank*, and *armed robbery* may be replaced by a slang word *blagging*.

Underspecification. Quite often desriptions of geographic objects contain specifications like *small town X* that can be omitted in a query. It seems that in most cases descriptive words like *large* or *small* may be simply dropped without loosing any information about the fact itself. Nevertheless in some cases, such as *small city Moscow*, a descriptive word may be used for distinguishing the defined object from some other (well-known) object.

Vertical Taxonomy Relations. By vertical taxonomy relation we mean hyponym-hypernym relation. A user formulating the query may have only a fuzzy knowledge about the fact they are looking for. For example, they may not be aware of the type of crime happened, or the date, or place. If a query requests for a robbery in *a small California town* the fact describing a robbery in *Livermore, CA* should be considered as relevant. Similarly, hypernym relation may exist between what-parts of two facts, e.g. *burglary – larceny – crime*, and when-parts, e.g. *June – summer*.

We expect this type of mismatching to be one of the most frequent in real applications.

Horizontal Taxonomy Relations. This type corresponds to the case when a user provided information on the same level of abstraction, but it does not match the fact precisely. Two facts refer to different concepts (e.g. *robbery – burglary*), but these concepts share a common hypernym (*crime*). Similarly, toponyms like *Livermore* and *Hartford* may be considered similar if they have similar description (in this example both names correspond to small towns in California). Clearly not every pair of words having the same hypernym are similar. For example, both *murder* and *stealing* are crimes, but the facts *A murder in an X's office* and *A stealing in an X's office* are not similar. This means that some additional constraints should be applied. For example, both words may be required to be similar in the sense of the next type.

General Similarity. Both types of taxonomy relations described above are special cases of semantic similarity between terms. We have separated them into specific classes because if such relations exist then one can expect strong semantic relation between corresponding facts. Nevertheless, the facts may be similar even if taxonomic relations are not present. For instance, one can expect that facts about *robbery of a bank* and *shooting in a bank* are similar.

3 EVALUATION

We calculate the semantic similarity between two facts $F_1 = (what_1, where_1, when_1)$ and $F_2 = (what_2, where_2, when_2)$ using the following

formula:

$$S(F_1,F_2) = \min\{s_t(t_1,t_2), s_r(r_1,r_2), s_n(n_1,n_2)\}$$

This function simply returns the worst mismatch of what-, where-, and when- parts of the two facts. In the sequel we briefly describe the functions s_t, s_r, and s_n.

The *What* Part. Possible types of semantic relations between terms in the *what* part of an event description can be divided into three classes: *vertical taxonomy relations*, (e.g. Livermore – small town, robbery – crime), *horizontal taxonomy relations*, terms that have a common direct hypernym, and *other semantic relations*. We calculate the semantic similarity between two terms $what_1$ and $what_2$ using the following formula.

$$\begin{aligned} s_t(t_1,t_2) \quad = \quad & \max\{s_{vert}(t_1,t_2)\times C_1, \\ & s_{horiz}(t_1,t_2)\times C_2, \\ & s_{stat}(t_1,t_2)\times C_3\}, \end{aligned}$$

where the function s_{vert} calculates the estimation of the fact that one of the terms $what_1$, $what_2$ is a hypernym of the other one. The function s_{horiz} calculates the estimation of the fact that terms $what_1$, $what_2$ have a common hypernym (i.e. they have a horizontal taxonomy relation). The function s_{stat} calculates statistical (corpus-based) similarity between the terms. C_1, C_2, and C_3 are weight coefficients that help implement the idea that the vertical taxonomy relation is more important than the horizontal taxonomy relation and the latter is more important than the "default" semantic similarity calculated statistically. In our experiments we took $C_1 = 1, C_2 = 0.8, C_3 = 0.8$.

For hypernym estimation we use lexical patterns approach, similar to (Bollegala et al., 2009). Statistical similarity function s_{stat} calculates the *Normalized Google Distance* (Cilibrasi and Vitanyi, 2007) by means of the YahooBOSS API[1]. We calculate the estimation of a fact that the terms $what_1$ and $what_2$ have a common hypernym using the following formula.

$$s_{horiz} = \max_{h_1\in H_1, h_2\in H_2} s_{stat}(h_1,h_2), \qquad (1)$$

where H_1 and H_2 are sets of possible hypernyms of $what_1$ and $what_2$, respectively. The sets H_1 and H_2 are constructed by means of lexical patterns approach.

The *Where* and *When* Parts. The specificity of the *where* part of a fact is that it usually contains geographical labels that can be mapped to latitude/longitude coordinates. Let w_1, w_2 be the strings representing the *where* parts of two facts F_1, F_2. The semantic similarity between w_1 and w_2 is calculated using the following formula.

$$s(w_1,w_2) = 1 - \frac{\min\limits_{g_1\in G(w_1), g_2\in G(w_2)} dist(g_1,g_2)}{MAX\_DIST},$$

where $G(w)$ is a set of all geographical objects matching the string w, $dist(g_1,g_2)$ is the great-circle distance between geographical objects g_1 and g_2, and MAX_DIST is the maximum distance between two points on the Earth surface, which is about 20018 km.

To calculate semantic similarity between two strings w_1, w_2 representing the *when* parts of the facts F_1, F_2, we use the following idea. We map the strings into dates and then calculate relative time interval between the dates applying some normalization. We use the following formula.

$$s(w_1,w_2) = 1 - \frac{d(w_1,w_2)}{d(w_1,w_2) + \min\{d(w_1,D), d(w_2,D)\}},$$

where $d(w_1,w_2)$ is the time interval (in seconds) between two dates matching the strings w_1 and w_2. D is the date representing the current moment. Normalization by D is used to implement the idea that a one-year interval 1000 years ago should be considered less important than the same interval nowadays, e.g. between January 1, 2009 and January 1, 2010.

Experimental Results. To evaluate the function proposed, we ran the following experiment. We manually extracted the following facts from the news (*what; where; when*).

1. robbery; Livermore; 28 July 2008.
2. burglary; California; July 2008.
3. deposit; Fremont; November 2, 2007.
4. anniversary; small town in California; summer 2007.
5. shootout; California; January 3, 1997.
6. crime; Hartford; August 27, 2007.
7. kill; West Yorkshire, England; February 21, 2010.
8. wine country festival; Livermore; 2008.
9. traffic; on main street in Pleasanton; Tuesday August 13, 2008.
10. armed robbery; 901 S. Main St. in Hartford, KY; On Friday July 13, 2007 at approximately 11:15 A.M.

The corresponding similarity matrix for the *what*, *where* and *when* parts of the facts is presented in Table 1.

One can see that for very short descriptions the results are meaningful. For example, the most relevant neighbours for the term *shootout* are *robbery* and

[1] http://developer.yahoo.com/search/boss

Table 1: Similarity between the what/where/when-parts of the test facts.

#	1	2	3	4	5	6	7	8	9	10
1	1/1/1	.5/1/1	.1/1/.7	0/1/.6	.2/1/.1	.9/.8/.7	0/.6/.2	0/1/.9	0/1/1	.7/.8/.6
2	.5/1/1	1/1/1	0/1/.7	0/1/.6	0/1/.1	.9/.8/.7	0/.6/.2	0/1/.9	.1/1/1	.4/.9/.6
3	.1/1/.7	0/1/.7	1/1/1	0/1/.9	0/1/.2	0/.8/.9	0/.6/.1	0/1/.7	0/1/.7	0/.8/.9
4	0/1/.6	0/1/.6	0/1/.9	1/1/1	0/1/.2	0/.8/.9	0/.6/.1	.2/1/.7	0/1/.6	0/.9/1
5	.2/1/.1	0/1/.1	0/1/.2	0/1/.2	1/1/1	0/.8/.2	.1/.6/0	.1/1/.1	.1/1/.1	.3/.9/.2
6	.9/.8/.7	.9/.8/.7	0/.8/.9	0/.8/.9	0/.8/.2	1/1/1	.9/.7/.1	0/.8/.7	.8/.8/.6	.9/.9/.9
7	0/.6/.2	0/.6/.2	0/.6/.1	0/.6/.1	.1/.6/0	.9/.7/.1	1/1/1	0/.6/.2	0/.6/.2	.1/.7/.1
8	0/1/.9	0/1/.9	0/1/.7	.2/1/.7	.1/1/.1	0/.8/.7	0/.6/.2	1/1/1	.1/1/.9	0/.8/.7
9	0/1/1	.1/1/1	0/1/.7	0/1/.6	.1/1/.1	.8/.8/.6	0/.6/.2	.1/1/.9	1/1/1	0/.8/.6
10	.7/.8/.6	.4/.9/.6	0/.8/.9	0/.9/1	.3/.9/.2	.9/.9/.9	.1/.7/.1	0/.8/.7	0/.8/.6	1/1/1

armed robbery, and the closest term to *anniversary* is *festival*.

4 CONCLUSIONS AND FUTURE WORK

In this paper we have described the task of fact search and proposed a function for calculating semantic distance between facts that are represented by single sentences and consist of three parts (*what*, *where*, and *when*). Some experimental results are provided, justifying the proposed function.

One direction of future work includes applying some methods from ontology theory. For instance, one can further detail the *what* part of the fact, e.g. applying the subject-predicate-object ("who-did-what") model of knowledge representation, that is extensively used in ontologies.

A function comparing the *where* parts could distinguish geographical names from some abstract descriptions, e.g. *in an American school* or *in a small town near the West Coast* and compare them somehow. The complex task of disambiguation of geographical names meaning several places (e.g. there are at least five cities named Moscow) can also be approached.

Finally, comparing parts of facts we did not take into account the context, i.e. the other parts. For example, if the *what* parts of facts are about politics, we can compare the *where* parts in some special way.

REFERENCES

Bollegala, D., Matsuo, Y., and Ishizuka, M. (2009). A relational model of semantic similarity between words using automatically extracted lexical pattern clusters from the web. In *EMNLP '09*, pages 803–812.

Cilibrasi, R. L. and Vitanyi, P. M. B. (2007). The google similarity distance. *IEEE Transactions on Knowledge and Data Engineering*, 19(3):370–383.

Islam, A. and Inkpen, D. (2008). Semantic text similarity using corpus-based word similarity and string similarity. *ACM Trans. Knowl. Discov. Data*, 2(2):1–25.

Li, Y., McLean, D., Bandar, Z. A., O'Shea, J. D., and Crockett, K. (2006). Sentence similarity based on semantic nets and corpus statistics. *IEEE Trans. on Knowl. and Data Eng.*, 18(8):1138–1150.

Naughton, M., Stokes, N., and Carthy, J. (2010). Sentence-level event classification in unstructured texts. *Information Retrieval*, 13(2):132–156.

Varelas, G., Voutsakis, E., Raftopoulou, P., Petrakis, E. G., and Milios, E. E. (2005). Semantic similarity methods in wordnet and their application to information retrieval on the web. In *WIDM '05*, pages 10–16.