Content Selection Challenge - University of Aberdeen Entry

Roman Kutlak
Chris Mellish
Kees van Deemter
Department of Computing Science
University of Aberdeen
Aberdeen AB24 3UE, UK
r.kutlak, c.mellish, k.vdeemter @abdn.ac.uk

1 Introduction

Bouayad-Agha et al. (2012) issued a content determination challenge in which researchers were asked to create systems that can automatically select content suitable for a first paragraph in a Wikipedia article from an RDF knowledge base of information about people. This article is a description of the system built at the University of Aberdeen.

Our working assumption is that the target text should contain information that is commonly known about the target person. The Wikipedia's manual of style mentions that "The lead [section] serves as an introduction to the article and a summary of its most important aspects[1]." What is most important about a person is likely to be often mentioned in biographies and hence it is more likely to be commonly known.

Our system was motivated by the notion of common ground, especially the way it was accounted for by (Clark and Marshall, 1981). Clark and Marshall (1981) introduce two categories of common ground: *personal common ground* shared by a small group of individuals and *communal common ground* shared by a community of people. We are most interested in the concept of communal common ground, which arises from the exposure to the same information within a community. For example, if there is a statue in front of your work place, you expect your colleagues to also know about this statue and so the information that there is a statue in front of you workplace becomes a part of the community knowledge (where the community are people who work at the same place).

Our hypothesis is that if we take a corpus of documents produced by some large community (e.g., English speakers), we should be able to approximate the community's knowledge of certain facts by counting how frequently they are mentioned in the corpus. For example, if a corpus contains 1000 articles about Sir Isaac Newton and 999 of the examined documents mention the property of him being a physicist and only 50 documents mention that he held the position as the warden of the Royal Mint in 1696 we should expect more people to know that he was a physicist.

We implemented the heuristic for approximating communal common ground and tested it in an experiment with human participants to measure whether there is a correlation between the heuristic's predictions and actual knowledge of people (Kutlak et al., 2012). In our implementation, we used the Internet as a corpus of documents and we used the Google search engine for counting the number of documents containing the properties. Although the number of hits is only an estimate of the actual number of documents containing a particular term, the heuristic achieved a Spearman correlation of 0.639 with $p < 0.001$ between the knowledge of people and the numbers of hits returned by Google.

Although there are some issues with the use of a proprietary search engine such as Google (for example, the search engine can perform stemming; see Kilgarriff (2007) for a discussion) search engines have been successfully used previously (Turney, 2001; Goudbeek and Krahmer, 2012).

2 Algorithm

The submitted system employs the heuristic outlined in in the previous section. The input is a collection of files containing information about people and a collection of human readable strings for each of the files. The data were taken from Freebase - a community created repository of information about people, places and other things. Each file is a small knowledge base containing a set of RDF triples describing the entity.

[1] http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Lead_section

Proceedings of the 14th European Workshop on Natural Language Generation, pages 208–209,
Sofia, Bulgaria, August 8-9 2013.

The data is encoded in machine-readable form (e.g., the fact that Newton was an astronomer is encoded as `ns:m.03s9v ns:type.object.type ns:astronomy.astronomer .`) so in order to find collocations in a human written text, each RDF triple has to be "lexicalised." This is done by mapping the RDF values to human produced strings provided by Freebase. After substituting the lexicalisations and removing some unnecessary information the algorithm adds the name of the target, which results in text such as *Isaac Newton type Astronomer*.

The algorithm reads one file at a time and creates a human readable string for each of the properties in the file. In the second step, the system removes disambiguations (text in brackets) and filters out properties that have the same string representation (duplicates). Additionally, properties with certain attributes are filtered out to reduce the number of queries[2].

In the third step, the system uses Google custom search API (a programming interface to the search engine) to estimate the score of each property. Properties that contain the name of the entity are penalised. This is done to reduce the importance of properties such as the target's parents or relatives. For example, if the algorithm was ranking properties of Sir Isaac Newton and a property contained the string *Newton*, the score assigned to that property was multiplied by 0.75. The properties were then ordered by the number of corresponding hits in descending order.

In the last step the algorithm selects the top ranked properties. The number of properties to select was calculated by the following equation $5 * log(|properties|)$. This equation was chosen by intuition so that a larger proportion of properties was selected for entities with a small number of properties than for entities with a large number of properties. The set of properties in the above equation is the set obtained after the filtering.

To prevent the system from selecting too many properties with the same attribute and to introduce variation, the system selected only five properties with the same attribute (e.g., five films, five books).

[2]For example, the knowledge base describing Antonín Dvořák contains 5670 properties of which 5154 have the attribute `music.artist.track`.

3 Concluding Remarks

The implemented system uses a simple document-based collocation heuristic to decide what properties to select. This makes it prone to favouring properties that contain common words or the name of the described entity. The advantage is that the system is relatively simple and versatile. The "common ground" heuristic could be combined with another heuristic that assigns negative score to properties that contain common words or a heuristic that estimates how interesting the property is.

Finally, we do not expect the system to perform better than machine learning based approaches such as that of Duboue and McKeown (2003) but it will certainly be interesting to see how far one can get with a simple heuristic.

References

Nadjet Bouayad-Agha, Gerard Casamayor, Leo Wanner, and Chris Mellish. 2012. Content selection from semantic web data. In *Proceedings of INLG 2012*, pages 146–149, Stroudsburg, PA, USA. Association for Computational Linguistics.

Herbert H. Clark and Catherine Marshall. 1981. Definite reference and mutual knowledge. In A. K. Joshi, B. L. Webber, and I. A. Sag, editors, *Elements of discourse understanding*, pages 10–63. Cambridge University Press, New York.

Pablo A. Duboue and Kathleen R. McKeown. 2003. Statistical acquisition of content selection rules for natural language generation. In *Proceedings of the 2003 EMNLP*, pages 121–128, Morristown, NJ, USA. Association for Computational Linguistics.

Martijn Goudbeek and Emiel Krahmer. 2012. Alignment in interactive reference production: Content planning, modifier ordering, and referential overspecification. *Topics in Cognitive Science*, 4(2):269–289.

Adam Kilgarriff. 2007. Googleology is bad science. *Comput. Linguist.*, 33:147–151, March.

Roman Kutlak, Kees van Deemter, and Chris Mellish. 2012. Corpus-based metrics for assessing communal common ground. *Proceedings of the 34th Annual Meeting of the Cognitive Science Society*.

P. Turney. 2001. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of the twelfth european conference on machine learning (ecml-2001)*.