E-TIPSY: Search Query Corpus Annotated with Entities, Term Importance, POS Tags, and Syntactic Parses

Yuval Marton and Kristina Toutanova
Microsoft Corporation
Redmond, WA, USA
{yumarton, kristout}@microsoft.com

Abstract

We present E-TIPSY, a search query corpus annotated with named **E**ntities, **T**erm **I**mportance, **P**OS tags, and **SY**ntactic parses. This corpus contains crowdsourced (gold) annotations of the three most important terms in each query. In addition, it contains automatically produced annotations of named entities, part-of-speech tags, and syntactic parses for the same queries. This corpus comes in two formats: (1) *Sober Subset*: annotations that two or more crowd workers agreed upon, and (2) *Full Glass*: all annotations. We analyze the strikingly low correlation between term importance and syntactic headedness, which invites research into effective ways of combining these different signals. Our corpus can serve as a benchmark for term importance methods aimed at improving search engine quality and as an initial step toward developing a dataset of gold linguistic analysis of web search queries. In addition, it can be used as a basis for linguistic inquiries into the kind of expressions used in search.

Keywords: term importance; query analysis; query parsing

1 Introduction

Search engines can easily serve highly relevant documents for common ("head") queries, simply by using past user click data for these queries. But there is no sufficient click data, if any at all, for rare ("tail") queries and new queries. For such queries search engines have to resort to more sophisticated methods that analyze queries based on the sequence of terms they contain. Term weighting is used to prioritize relevant documents. Weights are assigned by the search engine, with simple manual rules or sophisticated machine learning algorithms. Deeper natural language processing (NLP) analyses such as syntactic parsing can also help determine term importance and estimate document relevance.

We present a corpus consisting of tail queries issued to the Bing search engine,[1] together with manual annotation of term importance. In addition, our corpus contains annotation of named entities, part-of-speech (POS) tags, and syntactic analyses, generated using state-of-the-art NLP models. Along with the above, we include manual annotation of head-words for a small subset of the query set. Based on this manual annotation, we estimate the automatic parser accuracy and estimate the extent of correspondence between syntactic heads and most important terms.

A linguistically annotated corpus of tail queries is needed because queries are linguistically different from typical documents. This difference results in low quality of current NLP tools' output. We argue this quality can be improved with a resource such as ours, as has been the case with other task-specific annotations.

Our corpus can be used as a benchmark for new models of term importance in web search. It can also be used to study the impact of linguistic analysis on this task-driven problem, and to understand and improve the performance of NLP models in the domain of search queries.

2 Data Collection and Annotation

2.1 Query Set

We used English queries issued to the Bing search engine in the USA. Of these queries, we randomly selected tail queries, i.e., queries for which there was no sufficient user click data to determine the most relevant documents that the search engine should return. We further biased selection towards longer queries (longer than three words).

2.2 Term Importance Annotation

We used the services of English-speaking contractor crowd workers ("judges"), who annotated the queries using an in-house crowdsourcing application, similar to Amazon Mechanical Turk.[2] Judges were instructed to mark the most important term, the second most important term, and the third, if any.

The guidelines defined the most important term as the term that, if issued to the search engine alone, would return results of highest quality. The second most important term was defined as the term that if used together with the most important term, would help return results with highest quality. The third most important term was defined analogously. It is expected that as more terms are added to the query, the quality of the search results improves; the most important term is the strongest single-term indicator of the search intent. A term may span more than a single

[1] Bing.com

[2] https://www.mturk.com

word, if denoting a named entity or a multiword expression (defined as a sequence of words that is highly likely to occur in the returned document in the exact same order).

The application displayed a search engine window on part of the screen, with the query and results page. Judges were encouraged to alter the query to help themselves determine the most important terms. A summary of the guidelines was displayed on another part of the screen.

In order to maintain quality, we created a gold set, in which two experts (the authors) annotated a few hundred queries independently. There is a large number of valid (but not necessarily correct) annotation choices for each query, and the number of possible analyses grows quickly with the length of a query. For example, for a three-word query there are four possible ways to annotate the segmentation of the query words into terms: {1,2,3}, {1-2,3}, {1,2-3}, {1-3}, annotated by term position. There are eleven ways to annotate term importance for all possible segmentations, annotated as relative order of only top three terms: {1,2,3}, {1,3,2}, {2,1,3}, {2,3,1}, {3,1,2}, {3,2,1}, {1-2,3}, {3,1-2}, {1,2-3}, {2-3,1}, {1-3}. Because of this and the inherent ambiguity of this task (i.e., number of annotations that "feel correct", a number which also grows with the query length), a low inter-annotator agreement rate is expected. Indeed, the inter-annotator agreement rate turned out to be around 50%. Therefore, the only queries that were used for the gold set were those for which both experts agreed on the top three terms *and* their relative importance. The gold set was then used to train judges. For judgment, the same query was presented to up to four judges, or until two of them independently agreed. A single judge was limited to no more than 800 queries.

2.3 Entity, POS and Parsing Annotation

We used state-of-the-art tools: version 3.4.1 of the Stanford parser (Dan Klein and Christopher D. Manning, 2003) was used to generate part-of-speech, dependency, and constituency parses of queries. The dependency analyses were obtained by mapping the constituency parses from the model described in Klein and Manning (2003) to typed dependencies using the method from de Marneffe et al. (2006).

The named entity tags are more fine-grained than the basic Person, Organization, Location, and Miscellaneous distinction, and are generated from a perceptron-based in-house (Microsoft-internal) named entity recognizer built by Aitao Chen.

3 Corpus Description

Out of 5,000 queries, non-disqualified judges judged 4,719 queries, out of which 4,627 (93%) had two or more judgments. From the latter set, 3,542 (76.6%) had two or more judges agree. Looking at the markup of individual terms for each of the three positions, 75% of the annotations for the most important term had another annotator agree on that markup. However, for the second and third most important term annotations, this number dropped to 67% each.

For ease of use, we present the following subsets, fondly named along the corpus' acronym theme:

- *Sober* Subset: annotations that two or more crowd workers agreed upon, with a suggested division to training and test subsets, and
- *Full Glass*: all annotations.

4 Analysis

The correlation between syntactic head and most important term -- i.e., the percent of cases in which the most important term contained the syntactic head, as determined by the automatic parse, was strikingly low: 46%. This percentage was measured over the *Sober* subset. This low agreement could stem from the differences between syntax and semantics -- but could also stem from low quality parses on the query set, which is often quite far from the domain on which the parser was trained (Wall Street Journal). To tease these two factors apart, we manually labeled the head of each query in a small subset (217 random queries, after filtering out a few queries with adult or unclear intent). Head annotation was performed independently by each of the same two experts. Disagreements were then resolved by discussion. We annotated two gold variants:

(1) Fair comparison: attempt to follow the Stanford Parser's own conversion rules from constituency to dependency.

(2) Harsher comparison: head markup as we see fit (details below).

The parser's head accuracy relative to (1) and (2) was 44% and 42%, respectively. The correlation between the gold syntactic heads and the most important terms was 35% and 33% relative to (1) and (2), respectively. Crossing the syntax-semantics factor (most important term contains the syntactic head) with the parser head accuracy, we see in Table 1 that of the cases where gold syntax correlates with the IR semantics ('y' in column 1), the parser misses the correct head in a bit over half the cases (58%) – and almost two-thirds (64%) if using the harsher gold set. The largest category is where there is both syntax-semantics mismatch and the parser is wrong (last row). Having observed that, we take these numbers qualitatively only, due to the small sample size (due to the intersection between our gold head annotation and the *Sober* subset).

Term 1 has head (gold)	Parser correct head	# (fair)	# (harsh)	% (fair)	% (harsh)
y	y	16	13	15%	12%
y	n	22	23	20%	21%
n	y	33	31	31%	29%
n	n	37	41	34%	38%

Table 1: Most important term containing the syntactic head crossed with parser head accuracy

Query	pred. head word	gold head word (fair)	gold head MWE (fair)	gold head word (harsh)	gold head MWE (harsh)	interpretation / comments
276 smithtown blvd	blvd	blvd	smithtown blvd	276	276	house # 276 on smithtown blvd
tuscon to ft stockton	tuscon	tuscon	tuscon	ROOT	ROOT	how to get from tuscon to ft stockton (head "get" is missing)
xmas sugar sugar	xmas	xmass	xmas sugar sugar	Xmass	xmas sugar sugar	Sugar Sugar Christmas (Xmas) Edition (entity reference by inexact name)
starr st vallejo ca	ca	st	starr st	st	starr st	St / ave / blvd head the street name (and the city/state)
what is a bushel and a peck	is	what	what	bushel	a bushel and a peck	Our convention is Z in "what's Z?"; Stanford's is on "what"
which side of the body is the appendix on	is	is	is	on	on	Stanford Parser convention: is=main verb (root) with PP. Our convention is the P

Table 2: Query examples with predicted and gold head annotations, including multi-word expressions (MWE)

Our "harsher" gold variant differs from the Stanford dependency conversion rules in our head choice for the following:

1. Person names: first name is the head, following our interpretation that last names usually modify the first name, e.g., Joan of Arc.
2. Approximate names of entities: are still treated as named entities, and the head is assigned according to the interpretation of what the canonical form for that entity is. See "xmas sugar sugar" in Table 2.
3. Addresses: house number is head, following our interpretation "house 10 in Downing Street (which is in London (which is in the UK)) for "10 Downing St". If no house number, "st" and "ave" would be the heads of "starr st" and "fifth ave", respectively (and similar to the Stanford Parser), even though "starr" and "fifth" would be more informative (most important words).
4. Business names and geolocations ("the hilton hotel", "hotel deca", "mount everest", "the mississippi river"): the unique name is typically the most important word, but we decided to interpret such names similarly to Stanford, namely the head is the last word. Some motivating examples are "the barefoot contessa cookbook", "joe 's pub", and "comet net bus station", where the most important word is likely "contessa", "joe", and "comet", respectively, but it is much more natural to think that the syntactic head is "cookbook", "pub", and "station", respectively. The price we pay for this decision is some loss of generalization, e.g., "hilton" is no longer the head of both "hilton hotel" and "hotel hilton" (it is now only the head of the latter phrase).
5. Copula, equation declaratives and interrogatives: In the declarative form "X is Y", Y typically carries the new information, being more predicative than X, therefore Y is the head. Examples: "Star" in "Venus is the Morning Star"; "blue" in "the sky is blue" (this part is same as Stanford). In questions of the form "what's Z?" it is often ambiguous whether Z is X or Y (subject or object). For practical search-related reasons, we set Z to be the head. Example: "amendment" in "What is an amendment".
6. Auxiliary verbs and modals (e.g., "is being written", "has written", "can write", "should write"): the head is the main verb (write / written; same as in the Stanford Parser, mentioned it here for completeness).
7. *BE* as a main verb (e.g., "the book is on the table", "is there hotel near the airport"): the preposition following *BE* is the head (*on* and *near*, in the recent examples). In other words, we treat *BE* here as a copula.
8. Topicalization and non-typical / "marked" word order: head is assigned according to our

interpretation of the typical / "unmarked" word order, e.g., "fix" in "nexus 7 fix broken screen".

9. Omitted predicates (e.g., "tuscon to ft Stockton", omitting "how to get from"…): the head is ROOT.

Table 2 contains examples of issues we encountered, and our annotation choices. Often there is no single correct way to annotate syntactic dependency relations. For example, one can choose the main verb or (the head of) the subject to be the root. In equation declaratives and interrogatives ("X is Y" / "What's Z?"), one can choose X (subject), Y, or the verb *BE*. And as mentioned above, in "the Hilton hotel" one can choose *hotel* as the head (viewing *Hilton* as a modifier of *hotel*), or conversely, choose *Hilton* as the head (viewing *hotel* as a disambiguating designation: "which is a hotel"). The convention choice is therefore often guided by task needs and practicality. Our differences from the Stanford dependency conversion rules arise from our search-oriented perspective, and the under-specificity of the Penn treebank annotation regarding complex nouns and proper names.

5 Related Work

The different linguistic nature of search queries has been noticed before. Barr et al. (2008) analyzed the POS tag distribution of queries, reporting 40% proper nouns, and 30% (other) nouns in queries, with the majority of queries (70%) being noun phrases (NPs). NP query properties were also explored by Li (2010). Even though this linguistic nature has changed in recent years (e.g., a higher number of sentential, non-fragment queries), qualitatively these observations still hold.

As for Query POS tagging, segmentation and parsing: Bergsma et al. (2007) were one of the earliest published attempts to automatically segment queries. They focused on NP segmentation. Manshadi et al. (2009) attempted parsing the query as a bag of chunks. Bendersky et al. (2010) tagged and segmented the query by learning hidden information need variables. Yu et al. (2010) did so using deep-structured CRFs. Ganchev et al. (2012) learned useful transfer of POS tags from documents to queries, using search click logs. They released a search query corpus annotated with these POS tags. Carmel et al. (2014) used document-side syntactic parses to improve query-side term weighting. Gao et al. (2004), Cui et al. (2005), and many others, used query-side dependency parsing for document ranking and passage retrieval. Li (2010) analyzed NP query structure using semantic terms of intent head and intent modifiers.

Query term importance prediction has a rich history in IR literature, using vector-space similarity models, language models, supervised machine learning (Kim et al., 2010), and so on. Perhaps the most known method is TF-IDF (Salton, 1971; Sparck Jones, 1972). But the field is still poor regarding having human annotated resources for term importance, especially together with syntactic information. Having said that, the relation between term importance and syntactic structure has also been explored previously, e.g., with shallow syntax and POS tags, mainly for (weighted) query expansion (Grefenstette, 1992).

Our resource is the first to provide manual annotation of term importance for the web search task. This encourages application-driven studies of linguistic analysis for the important domain of tail queries.

6 Potential usage and Future Work

We see the contribution of this corpus on several levels:

1. Linguistic: Help better understand the relation between form/structure (syntax) and meaning (with term importance as proxy)
2. Applied: Improve search engine results, by using the corpus as a benchmark for term importance methods, or for better using the interaction between syntax and term importance.

We encourage other researchers to add other knowledge layers to this corpus, such as gold annotations, query-adapted parses, etc. In the future we plan to increase the corpus size and improve the associated automatic annotations.

The E-TIPSY corpus is publicly available at http://www1.ccls.columbia.edu/~ymarton/pub/lrec16/data.

7 Bibliographical References

Cory Barr, Rosie Jones, and Moira Regelson. "The linguistic structure of English web-search queries." Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2008.

Michael Bendersky, W. Bruce Croft, and David A. Smith. "Structural annotation of search queries using pseudo-relevance feedback." Proceedings of the 19th ACM international conference on Information and knowledge management. ACM, 2010.

Shane Bergsma, and Qin Iris Wang. "Learning Noun Phrase Query Segmentation." EMNLP-CoNLL. Vol. 7. 2007.

David Carmel, Avihai Mejer, Yuval Pinter, and Idan Szpektor. "Improving Term Weighting for Community Question Answering Search Using Syntactic Analysis." Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. ACM, 2014.

Hang Cui, Renxu Sun, Keya Li, Min-Yen Kan, and Tat-Seng Chua. "Question answering passage retrieval using dependency relations". In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '05). ACM, New York, NY, USA, pp. 400-407. 2005.

Marie-Catherine de Marneffe, Bill MacCartney and Christopher D. Manning. 2006. Generating Typed

Dependency Parses from Phrase Structure Parses. In LREC 2006.

Kuzman Ganchev, Keith Hall, Ryan McDonald, and Slav Petrov. "Using search-logs to improve query tagging." Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2. Association for Computational Linguistics, 2012.

Jianfeng Gao, Jian-Yun Nie, Guangyuan Wu, and Guihong Cao. "Dependence language model for information retrieval". In Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '04). ACM, New York, NY, USA, pp. 170-177. 2004.

Gregory Grefenstette. "Use of syntactic context to produce term association lists for text retrieval." Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval. New York, NY, USA, 1992.

Jae Dong Kim, Hema Raghavan, Rukmini Iyer and Chris Leggetter. "Predicting Term Importance in Queries for Improved Query-Ad Relevance Prediction." Proceedings of ADKDD'10, July 25, 2010, Washington D.C., USA.

Dan Klein and Christopher D. Manning. Fast Exact Inference with a Factored Model for Natural Language Parsing. In Advances in Neural Information Processing Systems 15 (NIPS 2002), Cambridge, MA: MIT Press, pp. 3-10. 2003.

Xiao Li. "Understanding the semantic structure of noun phrase queries." Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010.

Mehdi Manshadi and Xiao Li. "Semantic tagging of web search queries." Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2. Association for Computational Linguistics, 2009.

Gerard Salton. "The SMART retrieval system—experiments in automatic document processing." Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1971.

Karen Sparck Jones. "A statistical interpretation of term specificity and its application in retrieval." Journal of documentation 28.1 (1972): 11-21.

Dong Yu, Shizhen Wang, and Li Deng. "Sequential labeling using deep-structured conditional random fields." Selected Topics in Signal Processing, IEEE Journal of 4.6 (2010): 965-973.

8 Language Resource References

Dan Klein and Christopher D. Manning. "Fast Exact Inference with a Factored Model for Natural Language Parsing". In Advances in Neural Information Processing Systems 15 (NIPS 2002), Cambridge, MA: MIT Press, pp. 3-10. 2003.

9 Acknowledgements

The authors would like to thank Owen Rambow and Chris Quirk for useful discussions. Thanks also to Grady Simon for his help with the early version of the crowdsource app.