SemQuest: University of Houston's Semantics-based Question Answering System

Araly Barrera
University of Houston
abarrera7@uh.edu

Rakesh M. Verma
University of Houston
rmverma@cs.uh.edu

Ryan Vincent
McKendree University
revincent@mckendree.edu

Abstract

This work presents, *SemQuest*, a question-answering system used in the TAC 2011 guided summarization task based on semantics and extensions of a previous-developed single-document extractor. Our overall methodology includes: a data cleaning step, linguistic preprocessing among category articles, and a sentence extraction phase. A maximal marginal relevance technique, proposed by Carbonell et al., is also used in *SemQuest* to reduce redundancy and save space to answer as many category aspects as possible within a short summary extract.

1 Introduction

Automatic question-answering based on natural language content is one of the most challenging tasks confronting natural language researchers in the information-driven world of today. Work in this area has been propelled by a great desire to condense massive information loads into shorter, indicative or informative summaries, for everyday readers.

This year's overall Guided Summarization task consisted of a total of 44 different topics containing 20 relevant newswire articles, divided between part A and part B tasks. Each topic is mapped into one of five total categories. Categories included, "Accidents and Natural Disasters", "Attacks", "Health and Safety", "Endangered Resources", and "Investigations and Trials". For each part, participating systems were to automatically generate 100-word summaries for the 10 newswire articles falling under each topic. Topic summary effectiveness is based upon how well the summary answers a predefined list of aspects for the topic's category. A summary for a topic set in the "Endangered Resources" category, for instance, should cover aspects such as: *what* the resource was, the *importance* of the resource, the *threats*, and the *countermeasures*.

Each topic set additionally required the construction of two individual summaries: one for 10 documents composing set A and the second for 10 subsequent documents composing set B, used for what is also known as the update task. The update task is more challenging because it requires a summarizer to omit information contained in the articles of set A and thus provide a more-focused *update* of new information contained in the articles of set B again addressing the aspects for its topic category.

The work presented here reflects the approach used in the 2011 Guided Summarization task, our team's second year participating in TAC, and the improvements made to our past information extraction engine. In (BV10), we proposed a ranking-based information extraction method inspired by the importance of named entities and document date for sentence extraction decisions. Document sentences for sets A and B were essentially sorted and ranked based on four levels of prioritization and then selected for summary based on tie-breaking mechanisms. The crucial role played by named entities in answering questions (such as *who*, *where*, *when*, and *what*) for topic categories was the crux of that method.

This year's approach also exploits the use of named entities, among other important aspects, but in a new way. The overall methodology consists of

the following three major components:

1. **Data cleaning** This step was used to remove unwanted information and format article sets for input specifications required by our extraction engine.

2. **Category and sentence preprocessing** These steps were used to determine specific linguistic aspects of both a category and a topic set, including its individual sentences. Overall, the preprocessing involved the assignment of scores computed from a series of linguistic modules. We utilized tools such as SenseLearner (MC05), WordNet (Fel98), and Jigsaw (SCZ08). Computed sentence scores included semantic relationship to category aspects (a *WordNet* score), sentence pronoun count, sentence named entity count, and an *M-SynSem* score. The *M-SynSem* module is based on *SynSem* (BV11; BV12), our work on single document summarization. *SynSem* has shown successful results in past DUC and non-DUC datasets. *M-SynSem* is essentially a multiple-document version of the single-document extracting counterpart and was first implemented for this year's TAC competition.

3. **Sentence extraction** This final component of *SemQuest* involved the computation of preprocessed scores gathered from step 2 and the mechanism used to produce a final summary extraction. Some of the key aspects involved determining a summary *Named Entity Box* to represent a word threshold, computing a final sentence score, and using a maximal marginal relevance technique (MMR) to avoid redundant sentences the final summary.

Figure 1 gives an overview of the sentence extraction methodology. The following sections describe each component of *SemQuest* and the purposes behind the mechanisms involved.

2 Data Cleaning and Preparation

The following measures were taken as a means of organizing and preparing TAC 2011 guided summarization test data[1] as input to our system.

[1]We utilized the *cleaned data* version that was first released this year.

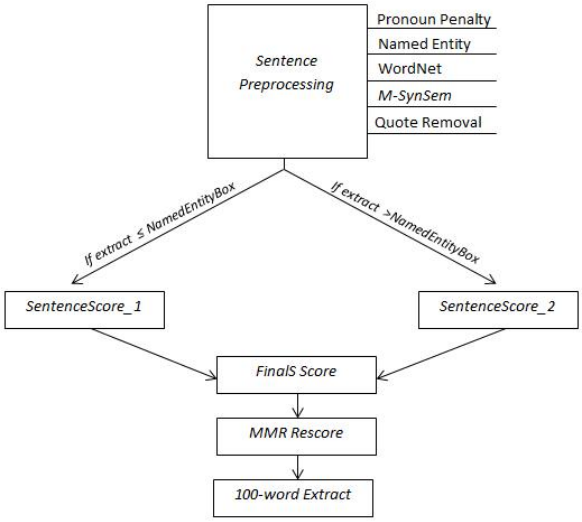

Figure 1: Figure showing *SemQuest*'s sentence extraction process

2.1 Noise Removal

This procedure involved the removal of unwanted content such as HTML tags and short, headline-like sentences that may have persisted in the text portions of articles in the cleaned data we used. The former issue was accomplished using an automated tag cleaner and the latter was performed by executing the Stanford Part-of-Speech tagger (Sta04) on all articles in efforts to eliminate those sentences lacking a verb.

2.2 Redundancy Removal

The approach taken to handle the update task for set B articles was to eliminate sentences containing mostly redundant information found in set A articles in order to avoid unnecessary and time-consuming processing of those sentences later. Set B article sentences having a fifty-percent overlap of stemmed words with those of set A were eliminated from the input data and from extraction consideration.

2.3 Article Set Concatenation

This component involved the concatenation of individual articles that would comprise sets A and sets B for all 44 total topics into single entities. All ten articles comprising set A for topic D1101, for instance, were combined into a single combined document,

"D1101A-Combined". The same was performed on the other 10 documents comprising set B data and all resulting documents were used as input for the next major steps described in Section 3.

2.4 Linguistic Identification

All data was also pre-tagged with a series of vital linguistic aspects required by later modules. These were accomplished using tools such as Jigsaw (SCZ08), SenseLearner (MC05), and WordNet (Fel98). The Jigsaw application is an interactive document analyzer that incorporates a named entity recognizer. This was used to identify and tag the presence of *named entities*, proper names classified under people, places, or things. "John Doe", for instance, is considered a *named person*, "U.S" is considered a *named location*, and "Federal Aviation Authority" is considered an *organization*. Named entity tagging was performed for topic relevancy computations later made under a *Named Entity* scoring module described in Section 3.

Five basic named entities identified by Jigsaw are *people*, *organizations*, *locations*, *date*, and *money*. Test data sentences were similarly pre-tagged with word sense identification using SenseLearner. This information was primarily used for the *M-SynSem* and *WordNet* scoring modules involving semantic analysis.

The final preparatory component involved exploiting the categories associated with this task and semantic relevancy to the aspects required in answering each. WordNet was therefore used to explore five levels of possible synonyms considered to answer specific category aspects which were then to be stored and used in the *WordNet* scoring module described in Section 3 as well. We associate, for instance, the words in Figure 1 to the *what*, *who affected*, *how*, *why*, and *countermeasures* aspects of the "Health and Safety" category.

affect, prevention, vaccination, illness, disease, virus, demographic

Figure 2: Sample Level 0 words considered to answer aspects for "Health and Safety" aspects

Synonyms of Level 0 words, as in the sample of Figure 1, generated words comprising Level 1 and the synonyms of the hyponyms of this level generated those for Level 2. The procedure of descending in the WordNet hierarchy and taking synonyms continued until reaching a fifth level for each category. The linguistic aspects gathered this component were stored and treated independently for use in modules described in the next section.

3 Sentence Score Preprocessing

Before sentence extractions were performed, a series of individual sentence scores highlighting various linguistic aspects were pre-computed on the organized documents resulting from the Data Cleaning and Preparation stage. Each sentence of each 10-document article set is assigned a pronoun penalty, a *Named Entity* score, a *WordNet* score, and an *M-SynSem* score. The goal of these scores is to make optimal extraction decisions in the final stage, presented in Section 4.

3.1 Pronoun Penalty

One issue typically encountered in sentence extraction involves the presence of sentences containing pronouns such as *we, he, she, they, that* in any portion, without direct reference. The following sentence has no place in a summary:

"*They* should be held accountable for *that*"

A reader would simply not comprehend *who* or *what* the sentence refers to if "they" or "that" are not directly identified in previously extracted sentences.

Our approach to preventing the extraction of such sentences was to assign the following pronoun score to all sentences, S, containing pronouns:

$$PronounScore(S) = \frac{TotalPronounCount}{|S|} \quad (1)$$

where $|S|$ is the total sentence length. The pronoun score acts as a penalty towards the final sentence score, as described in Section 4.

3.2 Named Entity Score

Data observations have emphasized the power of named entity presence in sentences. Our belief is that sentences with larger numbers of named entities are more likely to answer the majority of aspect questions required for the category. Consider the following sentence from TAC 2011 test data:

> "Prosecutors alleged Irkus Badillo and Gorka Vidal wanted to "sow panic" in Madrid after being caught in possession of 500 kilograms 1,100 pounds of explosives, and had called on the high court to hand down 29-year sentences."

This sentence is sourced from an article categorized under "Investigations and Trials." Summaries in this category are required to answer seven total aspects in a summary: *who*, *who involved* (prosecutors/investigators), *why*, *charges*, *defendant*, *plead*, and *sentence*. The aforementioned sentence, alone, manages to satisfy the questions of *who* the defendants are (Irkus Badillo and Gorka Vidal, *named people*), *why* they are on trial (to show "panic" in Madrid, a *named location*), and the *sentence* of 29 years sought out by prosecutors (on the high court, a judicial *named organization*). Three out of the seven total aspects have been answered at once and named entity identification had a major contribution.

In TAC 2010, our method of sentence extraction was to give highest priority to sentences containing larger numbers of *distinct* named entities (From the five basic: people, location, organization, date, and money). In 2011, however, we felt a stronger need to additionally reward those sentences that contained *most mentioned* named entities within each topic. That is, sentences were not only required to mention as many distinct named entities as possible for the aspects of its category, but higher weight would also be given to those that referred to most popular (and hence, most relevant) entities for a topic. Our approach was to compute a named entity weight score (*NEWeight*) per sentences, S. This score is defined as:

$$NEWeight(S) = \sum_{\substack{n \in S \\ n \in C(D)}} \frac{n\_Frequency\_Count(D)}{10} \quad (2)$$

where n is named entity present in S and $C(D)$ are the possible named entities required to answer specific category aspects of a document set D (composed of 10 documents) in which $S \in D$. $n\_Frequency\_Count(D)$ is the number of documents of set D in which n appears. To see the full list of named entity requirements per category, refer to Table 3. *The main purpose of the Named Entity weight score is to reward sentences that identify as many category aspects as possible.* Section 4 describes the role named entity recognition played in the sentence extraction phase of *SemQuest*.

3.3 WordNet Score

Named entity identification is a promising method of rewarding sentences that answer certain category aspects but it alone is not enough to answer those aspects that do not suggest any named entities at all. For instance, the *Plead* aspect from an "Investigations and Trials" category does not involve an "entity" but rather a reaction or an action of a defendant. An additional approach was therefore implemented through the assignment of a *WordNet* score, which is intended to address semantic information about categorical aspects and the relevancy that sentences have to these. Here, WordNet (Fel98) was used to determine five synonyms levels of aspect keywords described under Linguistic Preprocessing of Section 2. *The idea of providing a WordNet score was to reward sentences containing content most relevant to the aspects that must be answered through the identification of these synonyms.* The *WordNet* score assigned to a sentence, S, is defined as follows:

$$WordNetScore(S) = \sum_{\substack{w \in S \\ w \in L(C)}} \frac{1}{2^l} \quad (3)$$

where w is a word residing in both a sentence and a synonym list generated for its category topic (C), $L(C)$[2]. The value of l here indicates the level [0-4] in which w was found in $L(C)$. Note that stemming was performed to compare words in this procedure. This score is combined with the *NEWeight* and the *M-SynSem* score, as described next.

3.4 *M-SynSem* Score

In previous work, we developed an automatic, extractive, single-document summarization program called *SynSem* (BV11; BV12), which uses a fusion of syntactic and semantic techniques. Since *SynSem* performed well on our evaluations with both DUC 2002 news articles and a scientific article dataset, we have extended *SynSem* to multiple document datasets. This extension, called *M-SynSem*, is used to assign the *M-SynSem* Score in *SemQuest*.

[2] refer to Linguistic Preprocessing of Section 2 for how the list was constructed

In a nutshell, *SynSem* combines the syntactic and semantic qualities of a text for extraction decisions. It implements part-of-speech identification, named entity recognition, stopword removal, word popularity ranking, SenseLearner for word disambiguation, a parser for heading recognition and filtering, and WordNet for word analysis. Its architecture is based on the preprocessing and computation of a *weighted* sum of three individual sentence scores: a Keyword Score, a WordNet Score[3], and a Position Score. When executed on both DUC and non-DUC datasets, summary evaluation results have demonstrated the effectiveness of *SynSem*'s overall sentence position score, keyword score and WordNet scores.

We extended *SynSem* to multiple document summarization by computing the Position Score of a sentence relative to the document containing it and computing the Keyword Score and WordNet Score of the sentence by concatenating all the documents, i.e., these two scores are relative to the entire set of documents. For Keyword Score, we experimented with two approaches: TextRank (MT04) and Latent Dirichlet Allocation (BNJ03). Hence, there were two versions of the overall *M-SynSem* score. *The purpose of assigning an M-SynSem score to the sentences of a topic set was to exploit effectiveness of the SynSem approach to multiple documents.* The *M-SynSem* scores for a sentence $S \in D$ (where D is a 10-document set) used in this task are defined as follows:

$$\textit{M-SynSem}Score\_1(S) = \textit{M-SynSem\_TextRank}(S) \quad (4)$$

$$\textit{M-SynSem}Score\_2(S) = \textit{M-SynSem\_LDA}(S) \quad (5)$$

Here, *M-SynSem_TextRank* refers to the version using TextRank Keyword score and *M-SynSem_LDA* refers to the version using the LDA method in place of TextRank to compute a a sentence's Keyword Score.

Based on executions on TAC 2010 data and evaluations made on those, we determined that the *M-SynSemScore_1* method reached highest ROUGE-1 results and thus executed this version for the two submissions allowed to enter, leading to the rejection of the *M-SynSemScore_2*. In particular, one submission involved *M-SynSem* using the following weighted model combinations: 30 percent for the TextRank model, 20 percent for the WordNet model, and 50 percent for the Position model. The second submission involved *M-SynSem* using the weighted model combinations of 30 percent for the TextRank model, 70 percent for the WordNet model, and no Position model used. The WordNet model used here prioritizes sentences that are semantically closer to the headings and the article's title using synonyms to accomplish the task, while the Position model prioritizes of sentences closer to only top portions of a document.

[3]Not to be confused with the *WordNet* Score mentioned in this work

M-SynSem version	*ROUGE-1*	*ROUGE-2*	*ROUGE-SU4*
TextRank (.3,.2,.5)	0.33172	0.06753	0.10754
TextRank (.3,.7,0)	0.32855	0.06816	0.10721
LDA (0,.9,.1)	0.31792	0.07586	0.10706
LDA (.3,.7,0)	0.31975	0.07595	0.10881

Table 1: Recall evaluation scores for *SemQuest* using TextRank and LDA version of *M-SynSem* on TAC2011 Part A

M-SynSem version	*ROUGE-1*	*ROUGE-2*	*ROUGE-SU4*
TextRank (.3,.2,.5)	0.31792	0.06047	0.10043
TextRank (.3,.7,0)	0.31794	0.06038	0.10062
LDA (0,.9,.1)	0.29435	0.05907	0.09363
LDA (.3,.7,0)	0.30043	0.06055	0.09621

Table 2: Recall evaluation scores for *SemQuest* using TextRank and LDA versions of *M-SynSem* on TAC2011 Part B

When compared to TAC2011 models, *M-SynSem* with TextRank achieved higher evaluation scores than the version with LDA. Tables 1 and 2 show ROUGE-1, ROUGE-2, and ROUGE-SU4 Recall scores for the various versions of *SemQuest* experimented on Parts A and B, respectively. We submitted only TextRank for TAC 2011 since we were considering ROUGE-1 scores and our own judgment of overall responsiveness for a small random sample of article sets.

3.5 Quote Removal

We also note a high presence of quotes residing within the test data. In efforts to prevent the extraction of quotations, without actually removing these sentences for linguistic computations, we assign a negative infinity number to quoted sentences. Specifically, any quoted sentence, $Q \in S$ is given negative infinity as a value and the same for all other scores described in this section (*Named Entity*, *WordNet*, and *M-SynSem Score*). These quote penalties were performed by identifying sentences either containing quotation marks at the beginning and ending portions of a sentence or containing quotation marks in any portion including references to words such as "said", "told", "replied", and "testified".

Now that all major scores have been defined, the final phase of sentence extraction mechanisms of *SemQuest* is provided next.

4 Sentence Extraction

The final phase of *SemQuest* involves the sentence extraction decisions made based on the individual sentence scores pre-computed for each topic set, as previously defined. The entire process, described next, was performed on topic sets for part A and part B separately. A final sentence score, along with fulfillment of a *Named Entity Box* and an MMR re-scoring procedure are the final components of *SemQuest* to produce 100-word summaries for TAC 2011.

4.1 Named Entity Box

Given the 100-word limit and a list of required category aspects, we carefully consider summary space requirements to answer as many aspects as possible within the limited space. We first consider the named entity requirements for each category aspect (as mentioned in Section 3.2) and now construct a *Named Entity Box* in efforts to provide enough room to satisfy those aspects requiring certain named entities for a category. The *Named Entity Box* is essentially a threshold, or "reserved" word space that we give to summaries originating from the five different categories so that sentences with relevant named entities are preferred within this space. For instance, we find that taking a quarter of the 100-word summary for an "Endangered Species" topic is sufficient to answer the "countermeasures" aspect of that topic, where countermeasures is the only aspect within this category that may be answered with a named entity relating to *money*. The summary would therefore be given guaranteed space for answers to topic questions so that non-named entity aspects are answered within the 100-word space. Table 4.1 shows *Named Entity Box* assignments given to all categories, based on their aspects and the possible named entities these may refer to.

4.2 Final Sentence Score

To exploit our previously computed linguistic scores and our *Named Entity Box* fulfillment, we take the product of two combinations to compute separate sentence scores for sentence S residing in a document set, D.

$$SentenceScore\_1(S) = (WN(S) \times NE(S)) - P(S) \quad (6)$$

$$SentenceScore\_2(S) = (WN(S) \times MSynSem(S)) - P(S) \quad (7)$$

where WN refers to the $WordNetScore$ described in Section 3.3 (equation (3)), P refers to the $PronounPenalty$ described in Section 3.1 (equation (1)), NE refers to the $NEWeight$ score described in Section 3.2 (equation (2)), and *M-SynSem(S)* refers to the *M-SynSem* score utilized and described in Section 3.4 (equation (4)) for sentence $S \in D$.

The idea behind the $SentenceScore\_1$ score is to prioritize named entity presence in sentences and is only used to select best candidate sentences to fulfill the $NamedEntityBox$ threshold. Once the box is fulfilled, the $SentenceScore\_2$ prioritizes the next best candidate sentences resulting from the combination of the *M-SynSem* and *WordNet* score for the rest of the summary.

The very last component before final sentence extraction involves the selection of either a $SentenceScore\_1$ or $SentenceScore\_2$ based on the following criteria:

$$FinalS = \begin{cases} SentenceScore\_1 & \text{if } |E| \leq NEBox \\ SentenceScore\_2 & \text{if } |E| > NEBox \end{cases} \quad (8)$$

Topic Category	*Aspects*	*Named Entity Possibilities*	*Named Entity Box*
1. Accidents and Natural Disasters	what	–	$\frac{5}{7}$
	when	date	
	where	location	
	why	–	
	who affected	person	
		organization	
	damages	–	
	countermeasures	money	
2. Attacks	what	–	$\frac{5}{8}$
	when	date	
	where	location	
	perpetrators	person	
	why	–	
	who affected	person	
		organization	
	damages	–	
	countermeasures	money	
3. Health and Safety	what	–	$\frac{3}{5}$
	who affected	person	
		organization	
	how	–	
	why	–	
	countermeasures	money	
4. Endangered Resources	what	–	$\frac{1}{4}$
	importance	–	
	threats	–	
	countermeasures	money	
5. Investigations and Trials	who/who involved	person	$\frac{2}{6}$
		organization	
	why	–	
	charges	–	
	plead	–	
	sentence	–	

Table 3: Named Entity Box determination based on categories and aspects these require. This is the space required within a 100-word extract. Numerator is the number of distinct named entity categories and denominator is the number of aspects for the category.

The maximum $FinalS$ scores were computed in combination with the MMR procedure described next for the extraction decisions made for extract E.

4.3 MMR for Redundancy Removal

A big challenge in multiple document summarization is detecting and eliminating redundancy in the summary since similar sentences will appear in multiple articles. It is highly undesirable to repeat the same or partly same information in the summary. For this purpose, we implemented the maximal marginal relevance (MMR) technique with lambda annealing, as proposed in (CGG97), to re-score $FinalS$ scores, mentioned previously. The role of this technique was to preserve high marginal relevancy and to include relevant, novel sentences in an extract.

Originally used for document reordering, the MMR procedure involves a linear combination of relevancy and novelty measures. It is used in *SemQuest* as a way to re-order extract candidate sentences determined from the $FinalS$ score defined in Section 4.2. A final MMR score in *SemQuest* is computed for all candidate sentences, $S_i \in R$, as R represents the set of $FinalS$ scores and $R_i = max(FinalS)$. The MMR score used in the selection of the set of sentences for an extract, E, is as follows:

$$MMR = \max_{S_i \notin E} \lambda Sim_1 - (1-\lambda) \max_{S_E \in E} Sim_2(S_i, S_E) \quad (9)$$

where $Sim_1 = R_i$, a candidate sentence score R_i (whose sentence $S_i \notin E$), and Sim_2 is the stemmed word-overlap between S_i and $S_E \in E$ (an extract sentence). Here, both Sim_1 and Sim_2 represent the relevancy metric and λ represents the novelty parameter, where $\lambda = 1$ signifies no novelty and $\lambda = 0$ signifies high novelty. The MMR computation performed in *SemQuest* involved decrementing λ, whose initial value was set to $\lambda = .7$ (high relevancy, low novelty), to $\lambda = .3$ (high novelty, low relevancy). The candidate sentence obtaining a maximum MMR would be selected and added to E. This procedure was performed on all other candidate sentences until reaching $|E|$ = 100 words.

5 Performance

Compared to last year, all scores are higher for both submissions. This includes the manual evaluations, the Rouge 2 and SU4 score and the BE scores. In particular, for the overall responsiveness measure, we have improved our rankings by 17 percent in the A category and by 7 percent in the B category for overall responsiveness. We also beat both baselines for the B category in overall responsiveness score and one baseline for the A category. Our best run is better than 70 percent of participating systems for the linguistic score.

6 Conclusions

We have made significant progress since last year when we participated in TAC's guided summarization task for the first time. Of course, there is still room for improvement. Apart from improvements to *M-SynSem*, we plan to research sentence compression for future TAC competitions, which is an area we have not yet explored.

References

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 2:993–1022, 2003.

Araly Barrera and Rakesh Verma. A Ranking-based Approach for Multiple-document Information Extraction. In *TAC 2010 Proceedings*, 2010.

Araly Barrera and Rakesh Verma. Automatic Extractive Single-document Summarization: Beating the Baselines with a New Approach. In *Proceedings of the Symposium on Applied Computing*. ACM, 2011.

Araly Barrera and Rakesh Verma. Combining Syntax and Sematics for Automatic Extractive Single-document Summarization. In *Proceedings of the* 13^{th} *Int'l Conf. On Intelligent Text Processing and Computational Linguistics*, volume LNCS 7182. Springer-Verlag, 2012.

J.G. Carbonell, Y. Geng, and J. Goldstein. Automated Query-relevant Summarization and Diversity-based Reranking. In *15th International Joint Conference on Artificial Intelligence, Workshop: AI in Digital Libraries*, 1997.

Christine Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

R. Mihalcea and A. Csomani. SenseLearner: Word Sense Disambiguation for all Words in Unrestricted Text. *ACL*, 2005.

R. Mihalcea and P. Tarau. TextRank: Bringing Order into Texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. (EMNLP, 2004), March 2004.

J. Stasko, G. Carsten, and L. Zhicheng. Jigsaw: Supporting Investigative Analysis through Interactive Visualization. *Information Visualization*, 7(2):118–132, 2008.

Stanford. Stanford Log-linear Part-Of-Speech Tagger. `http://nlp.stanford.edu/software/tagger.shtml`, 2004.