Semantics and Question Answering: an approach to "*why*" questions

Svetlana Stenchikova

Semantics course

Stony Brook University

December 18, 2006

The internet brings to our fingertips unlimited information resources. We need tools to access these resources. Search engines are the most widely used tools for finding information on the web today. They crawl the web, index information and build large dynamic dictionaries. Researchers and engineers are exploring question answering (QA) as another tool for extracting information from the web or from a set of documents. S*earch* result is a set of text snippets and links to the related documents, while a *question answering* result is a sentence - a concise answer to the posed question. Current question answering technology is based on matching a string from a question to strings in documents. This simple technique works surprisingly well for open-domain (factoid) questions: *who, what, when, and where* finding an answer in approximately 30% or cases (Vorhees 2002 – 2005). This paper investigates an approach to answering *why* questions and the role of semantics in answering them. *Why* questions are inherently more difficult than factoid questions. In most cases finding an answer to a *why* question requires a semantic analysis and domain knowledge. In this paper, I look at several examples of *why* question-answer pairs and derive an automatic approach for detecting an answer from text for the chosen examples.

Current Approaches to Automatic Question Answering

Despite a seeming similarity to search, question answering is a significantly more complex. It requires linguistic knowledge and extra processing steps. Automatic Question Answering uses search as one of its components. A general QA system design is illustrated on Figure 1.

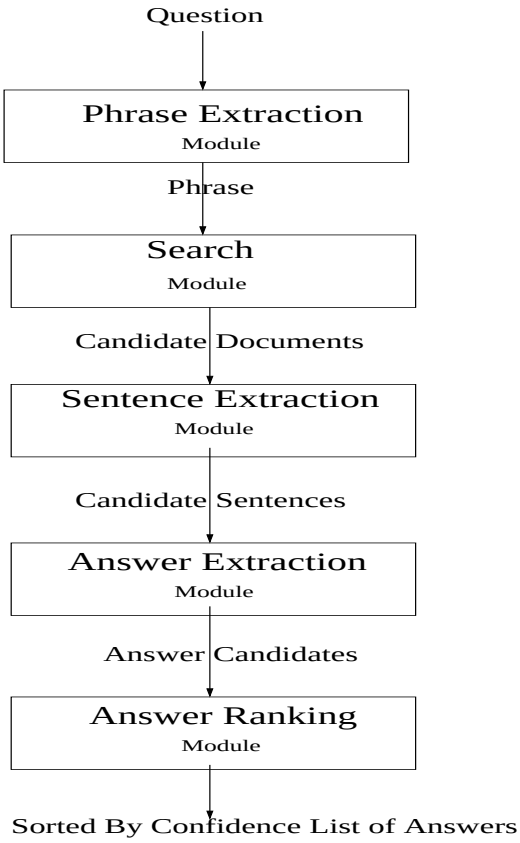


Figure 1: Automatic question answering architecture

Question answering takes a natural language question as input. A phrase extraction module automatically retrieves a search phrase from a sentence. Next, QA passes the automatically extracted phrase to a search engine. For example, consider the following question:

Who won the Nobel prize in literature in 1988?

The Phrase Extraction step identifies the search phrase for this question as "*won the Nobel prize in literature in 1988"*. A search engine returns a set of documents containing the phrase and *sentence extraction* returns a set of candidate sentences. Then *answer extraction* gets the actual answer from a set of candidate sentences. The basic version of *answer extraction* expects the answer to the question occurs next to the search phrase, as in this answer:

Naguib Mahfouz *won the Nobel prize in literature in 1988*

Using the Web as a data source for answering questions has both advantages and disadvantages. The main advantage is that the Web is highly redundant - the same information may be repeated in different sources. A description of a news-worthy event may appear in multiple newspapers or may be discussed by many interested bloggers. A scientific or technical term may appear in various dictionaries, blogs, personal websites, etc. The disadvantage of the Web is that it is not a reliable resource, it may contain false or biased opinion information that is hard to detect for an automatic question answering system.

The approach to question answering described above is based on matching a phrase from a question to phrases in text. It assumes that the search string occurs in some of the source documents and that the correct answers prevail over the incorrect ones. This approach fails when there is no string match between a question and sentences in text, or if the answer extraction requires semantic knowledge. Consider this question from a Text Retrieval Conference competition:

What position did Moon play in professional football? (Correct answer: *quarterback*.)

One of the occurrences of the correct answer appears in a source document:

```
Free agent quarterback Warren Moon will visit
the Cleveland Browns on Tuesday and Wednesday, his agent said
Friday.
```

Although the sentence contains the correct answer, it is impossible to detect it by simple string matching since neither of *"play"*, *"position"*, or *"professional baseball"* appears in the sentence. Extracting an answer requires domain knowledge on playing, sports, and baseball, in particular, that "*quarterback*" is a name for a player position in baseball.

Semantic Role Labeling

Semantic role labeling (SRL) also known as *shallow semantic parsing* is a research area of the Natural Language Processing. SRL identifies predicates and their arguments in a sentence (Gildea & Jurawsky, 2002). Current systems consider only verb predicates. The argument types are: ARG0 for an agent, ARG1 for a direct object, ARG2 for an indirect object, ARG–TMP for a temporal argument, ARG-LOC for location argument etc. For example:

Q: *[**ARG0 Who**] [**TARGET created**] [ARG1 the comic strip Garfield]*

*A: Garfield is [ARG1 a popular comic strip] [**TARGET created**] [**ARG0 by Jim Davis**] featuring the cat Garfield*

It has been shown that the semantic information from SRL parsing benefits answering factoid questions. In (Stenchikova, et. al., 2006) we parsed the question and the candidate sentences using a semantic parser. Semantic information allowed the system to pick out the correct answers to factoid questions more accurately.

Modern semantic role labeling systems use a statistical approach to extract a partial semantic parse of a sentence. Statistical approach to a NLP task is opposite to a rule-based approach. In a rule-based method, experts come up with a set of rules (e.g. syntactic context free rules for the syntactic parsing). While in the statistical method, a data set is manually annotated by an expert and a system uses the dataset to train its prediction model. The prediction model is used to automatically annotate new sentences. For most NLP tasks (part-of-speech tagging, chunking – identifying phrases in a sentence, syntactic parsing, etc.) rule-based methods are developed first, and statistical methods are developed later on. It is intriguing that in the case of semantic role labeling, the statistical approach is developed first, while there is no attempt to use a rule-based

approach for this task. A possible reason for this could be that creating an exhaustive set of rules for semantic derivation is more challenging than creating an exhaustive set of syntactic rules. If we had an exhaustive set of rules for derivation available, a tool like Semantica (Larson, et al 1997) could be used for automatic derivation of semantic representation of a sentence.

Semantic analysis with its application to automatically answering *why* questions is further explored in the following sections.

Answering "Why" questions

In this section I investigate an approach for an automatic answer extraction for "*why*" questions from text. I consider two cases: one where the answer text contains a lexical causation cue (because, caused by, etc.) and the second, when it does not. In both cases I look at non-trivial examples with no exact lexical match between a question and an answer. My approach utilizes the modern semantic role labeling technology. I evaluate the strengths and limitations of my approach and show what additional knowledge and technology is required to extract answers for the considered examples.

I draw my examples of question-answer pairs from a dataset collected by Vebern (2006) where several annotators read short newspaper articles and constructed *why* questions about the events described in these articles. Other annotators constructed answers to these questions. Vebern reports that for the majority of questions there is a high syntactic and lexical difference between the questions and the answers in the candidate sentences.

Semantic relation between phrases

In the next sections I use *semantic relations* to compare frame attributes. The semantic relations are the domain knowledge stored in anthologies and dictionaries. Some of the semantic relations useful for a system are:

- Synonym, e.g. famous & well-known
- Hypernym(type-of/is-a), e.g. car & vehicle
- Meronym (part-of), e.g. wing & bird
- Implication, e.g. argue & "not oppose" (if A argues for B, then A does not oppose B)

The semantic relations can be either constructed manually or gathered from data automatically. WordNet (Fellbaum 2001) is an example of a manually constructed ontology which captures hypernym, meronymy and synonym relations. Girju (2003) automatically extracted causal relations from text. Designing and building a useful ontology is not trivial; it is one of the current research areas of the Natural Language Processing. In the following sections I assume that the semantic relation information can be accessed automatically by a system.

Text contains causation cues

This section describes an automatic approach to answering *why* question from a text with a causation cue. Consider a question from Verbene's dataset:

> Q1. Why *has Dixville Notch become famous?*

The passage from the text containing the answer:

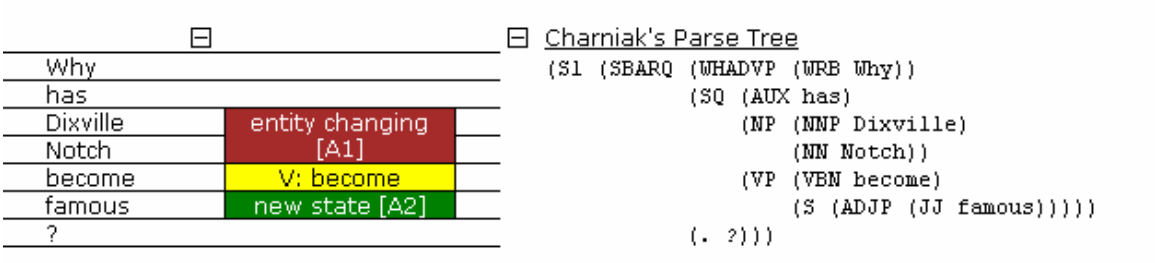
> Primary primacy is important to the 39 residents of Dixville Notch, a once obscure hamlet hidden away in the icy mountains of New Hampshire's North Country. Since 1964 it has **grown famous by being the first "precinct" to declare its election result.**

In order to automatically identify this passage as a candidate answer for the question Q1, the automatic analysis needs to:

- Find the synonymy between "X becomes famous" and "X has grown famous" corresponding to a semantic frame become(X, famous)
- Identify using co-reference resolution that the pronoun *it* refers to Dixville Notch.
- Identify *by being* as a lexical cue for the causation

The automatic semantic role labeling identifies a frame in the question FQ1:

FQ1 *become (Dixville Notch, famous).*



FQ1: become(A1/Dixville Notch, A2/famous)[1]

To capture the question word *"why"* in the frame convert the FQ1 to FQ1' by adding an unknown variable *REASON*:

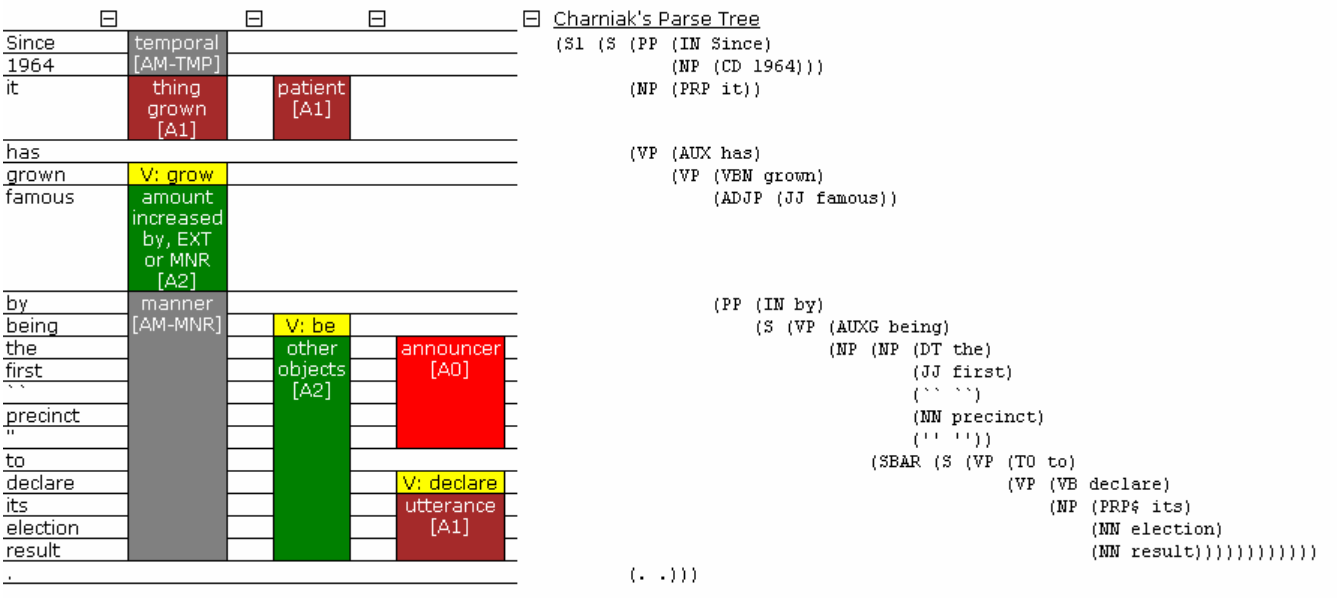
FQ1': *become (Dixville Notch, famous,* ***REASON****).*

This can be done automatically by triggering a rule for each question:

RULE-WHY: if "why" is present in the question frame, add an unknown variable *REASON* to the frame.

The automatic analysis of the candidate sentence text using the same program returns:

[1] This and all other semantic parses are done using http://l2r.cs.uiuc.edu/~cogcomp/srl-demo.php



A co-reference resolution is applied to each pronoun to identify its referent. It should identify that *it* refers to *Dixville Notch*. The modern technology for automatic co-reference resolution achieves reasonably good results for pronominal resolution (Baldwin, 1997)

Three frames are identified by the automatic semantic parse. One of them is FA1 (it will be used to extract an answer):

FA1: *grow (AM-TMP/since 196, A1/ Dixville Notch, A2/famous, AM-MNR).*

The automatic semantic role labeling treats the verb *grow* in FA1 literally, or synonymous to *increase*. It identifies its direct and indirect object attributes as "thing grown" and "amount increased by". However, in this case *grow* has a sense closer to *become* than to *increase.* A word sense disambiguation module should be utilized prior to the semantic role labeler to identify the correct sense of the verb. Word sense disambiguation (WSD) is a research area of the Natural Language Processing aimed at identifying a correct meaning of the word from the context (Ide, Veronis 1998). WSD captures rules based on syntactically related constituents from the context. For example,

one rule may state that if *grow* appears in a syntactic structure with an adjective sister node, e.g. *(VP (VBN grown) (ADJP (JJ famous))*, it is more likely to carry a meaning of "become" or "change state" than "increase" .

Assuming that the word-sense disambiguation and co-reference resolution succeed, we can identify a semantically relation between the frames for a question (QFrame) and an answer (AFrame). To match the question and the answer frames using RULE-MATCH: for each attribute in the question frame, find a corresponding attribute in the answer frame. In the given example we get exact word match for the frame strings: *Dixville Notch* and *famous*. ARG-MNR attribute in the answer is matched to the unknown attribute REASON in the question. The answer to the *why* question is the text of the ARG-MNR argument in the answer frame:

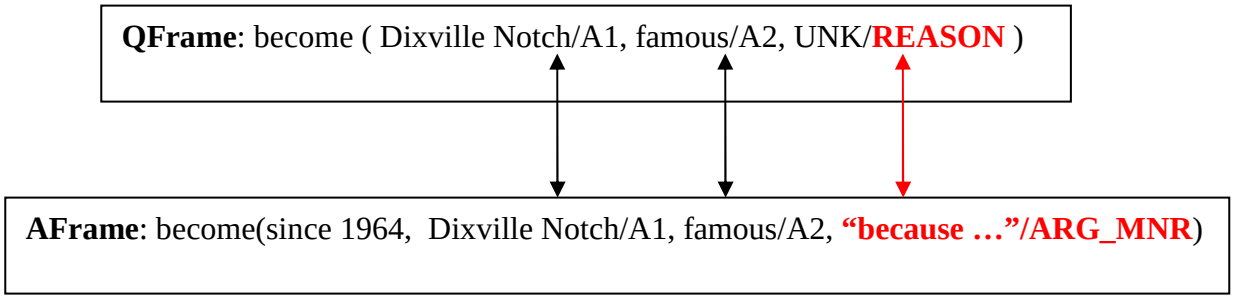


Figure 2A. Matching question and answer frames

Consider a hypothetical example where the question phrase is Q':

Q': *Why did* ***the small town*** *become* ***well-known***.

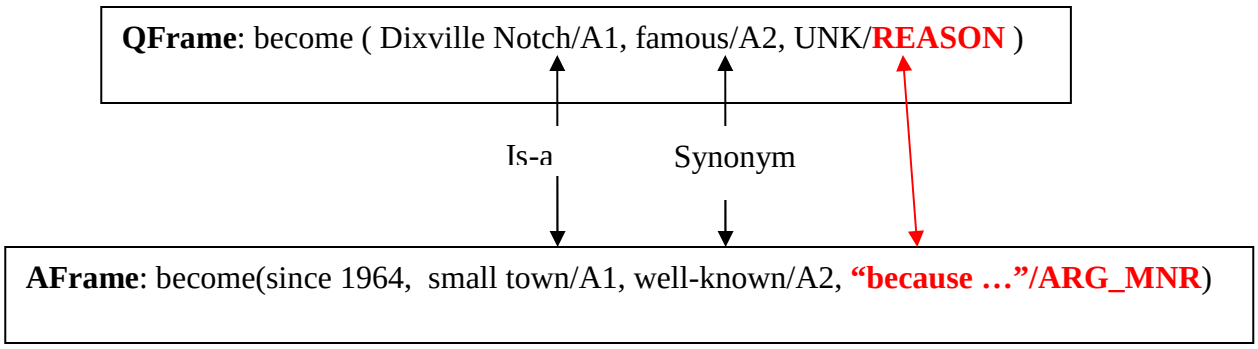


Figure 2B. Matching question and answer frames without lexical match

By identifying a hypernym relation (*Dixville Notch* **is a** *small town)* and synonym relation (*famous* is **synonymous** to *well-known)* the frames may be matched (Figure 2B). This example shows how ontology for identifying synonyms and hypernyms can be used to match the question and a candidate answer frames when there is no exact lexical match between the attributes.

To summarize, finding an answer in a sentence with causation cue involves:

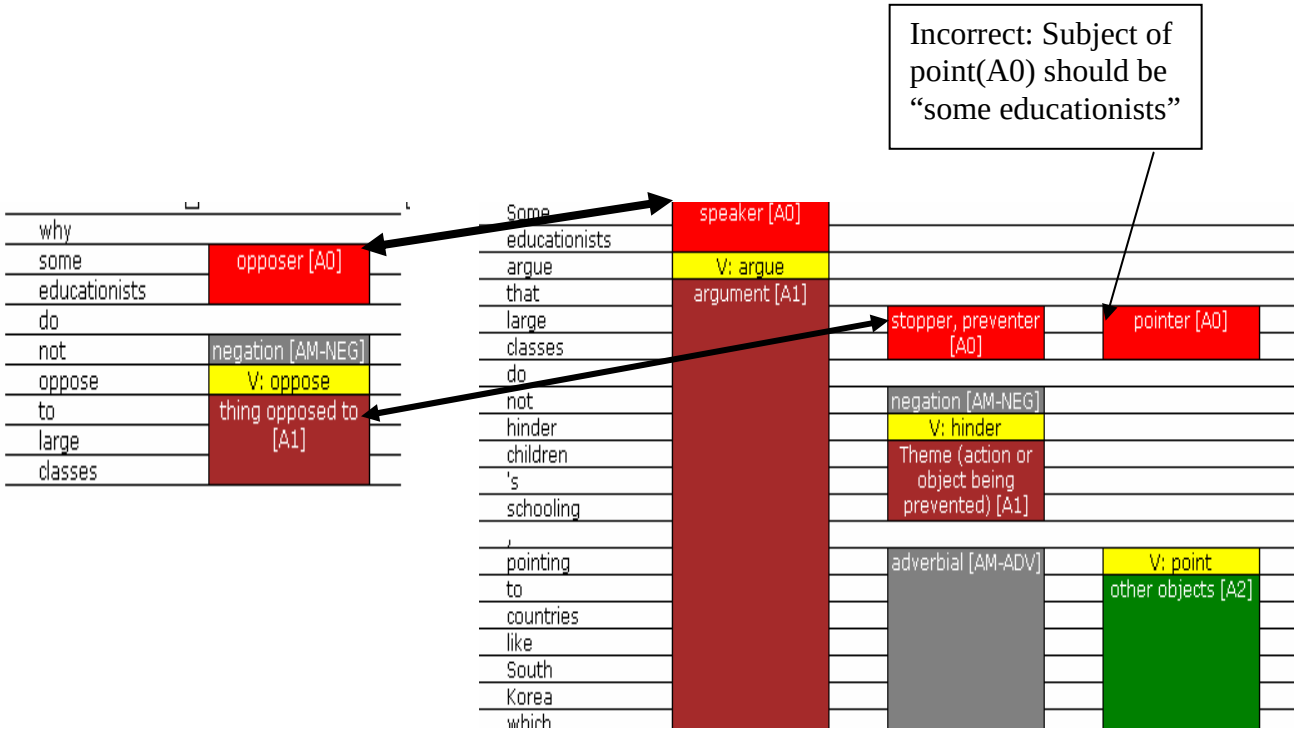
1) Semantic parse of a question and of text
2) Disambiguate the verbs
3) Apply RULE-WHY to the question frame
4) Identify referents of the pronouns using co-reference.
5) Apply RULE-MATCH to match question and the answer
6) Use ARG_MNR argument from the answer frame as an answer to the *why* question

Text does not contain a causation cue

This section describes an automatic approach for finding an answer to a *why* question from text without a causation cues. This approach uses heuristics to extract a frame (or a part of a frame) from the candidate sentence as an answer to the question. A natural language generation module is necessary to convert the answer semantic frame to the answer text. Consider a Q2 and its candidate answer text:

Q2: why some educationists do not oppose to large classes?[2]

"Some educationists argue that large classes do not hinder children's schooling, pointing to countries like South Korea which appear to achieve better academic results despite having groups of up to 60."



The partial semantic parse of the question:

FQ = Not_Oppose(A0="some educationalists", A1="large classes", REASON = ?)[3]

The partial semantic parse for the answer identifies several nested semantic frames:
F0 = Argue(who="some educationalists", argument= F1)
F1 = Not_Hinder(A0/preventer="large classes", A1/theme="children's schooling", adv=F2"))
F2 = point(A0/pointer = "some educationists", A2 = "other countries …")

[2] This question was simplified. The original question was: *why is it that not everyone opposes to large classes?* In my analysis I ignore the quantification problems: "some educationalists" is treated as a simple noun phrase and not analyzed further.

[3] For the simplification I treat the negation as a part of the predicate.

A valid answer to the question may be: a) the whole candidate sentence, b) a part of a sentence, e.g. "*because some educators argue that large classes do not hinder children's schooling*", or c) a paraphrase of the sentence, e.g. "*some educationalists point to countries like South Korea with large classes and still good academic results"*

Algorithm for extracting the complete sentence as an answer

1) Find a frame *F* in the candidate sentence where the subject (A0) matches (either exact lexical match, hypernym, or synonym relation) the subject (A0) of the question

In the above example, the subject of the frames F0 (*Argue)* and *F2 (Point)* A0="some educators" matches the A0 of the question.

2) Identify the relation between the verbs of the question frame and the matched answer frame (not_oppose and argue). If this is an *implication* relation, use the frame *F* as an answer.

Semantic knowledge is required to identify a relationship between *argue* and *not oppose:* an ontology entry: *argue for X* implies *not oppose X*. The relation needs to be established not just between the verbs, but between the verb-attribute pairs: "*not_oppose large classes*" and "*argue that large classes … "*. We assume that the *argue* frame in the candidate answer has a meaning of "*argue **for** large classes*", however this is only a guess because we are not analyzing the meaning of the *argue* frame. If the statement was "*argue that large classes hinder children's schooling*", its meaning would be "*argue **against** large classes*" and the sentence would not be a valid answer to the posed question.

The frame F0 (*Argue*) is selected as an answer, while the frame F2 (*point)* is not selected because *argue X* does not imply *point to X*. The frame F0 (Argue) corresponds to the complete candidate sentence.

Conciseness of an answer is important if the system has a limiting user interface (a PDA) or runs over a voice interface.

Algorithm for extracting a more concise answer: "*because some educators argue that large classes do not hinder children's schooling"*

Steps 1 & 2 are the same as above

3) If an argument of A1 is a frame *F1(A0, A1, A2 …)* and A0="large classes" in *F1* matches an argument of the question, simplify F1 to F1' (A0, A1), by keeping only the subject and direct object arguments.

The motivation behind the third step is to remove irrelevant information from the answer. An assumption is that the direct argument of F1 contains a meaninful answer, while other arguments are irrelevant. The third step applied to the example identifies the frame F1 = not_hinder(A0, A1, AM_ADV), where A0="the large classes" matches the argument of the question. The AM_ADV is removed from the answer sentence, so the answer frames are:

F0 = Argue(who="some educationalists", argument= F1')
F1' = not_hinder(A0=large classes, A1=children's schooling)

The lexical representation for these frames can be derived by a natural language generator: "some educationists argue that large classes do not hinder children's schooling".

Rephrasing the answer

In the previous example, the answer was part of the whole candidate sentence. The human annotator who answered this question came up with slightly different answer:

H1 *some educationalists point to countries like South Korea with large classes and still good academic results*

H1 is paraphrased on several levels. First, this answer makes several generalizations by paraphrasing the frame arguments:

"groups of up to 60" is rephrased as "large class"

And "better" is replaced by "good" in the phrase "good academic results".

Let **A** stand for the statement "large class/group of 60", **B -** for "good/better academic results", and **E** – for "some educationists". Then statement H1 and the phrase from the text (T) are:

H1: *E point to countries like SK with A and still B*

T: *E point to countries like SK which appear to achieve A despite having* B

The second level of a paraphrase uses constituents A and B within different strings:

"*which appear to achieve **A** despite having **B***" ~ "*with **A** and still **B***"

Paraphrasing an answer requires the knowledge of paraphrase strings, or different lexical representations for the same semantic knowledge. Flexible and correct paraphrasing is a difficult problem in natural language processing research. Choosing an answer from a set of possible paraphrases by adapting it to a particular user is an interesting research problem.

Case of Conflicting Candidate Sentences

The shallow semantic analysis described in the previous sections avoids understanding a question. Consider a scenario where we had two candidate sentences.

1) *Some educationists argue that large classes **do not hinder** children's schooling*

2) *Some educationists argue that large classes **hinder** children's schooling*

This would give us two conflicting candidate answers. In order to choose a correct answer (1) we must have semantic knowledge about educationists and their goals toward schooling and to be able to derive a correct answer.

Educationists aim at teaching children, so hindering children's schooling is not in their interest. Ontology with an entry for "Educationist" and the knowledge that *hinder* is an antonym of *improve* enable the system to pick a correct answer.

A sample ontology entry for **Educationist**:

Type	Person
Type	Job description
Job location	School, university…
Goals	Teach kids, Improve schooling,

Assume domain knowledge:

- K1. "If X does not improve children's schooling, then educationist opposes X" or not_improve(X, children's schooling) -> oppose (E, X)
- K2. Hinder is opposite of improving: hinder(X,Y) -> NOT_improve(X,Y)

In order to pick a correct answer we need to automatically perform a logical proof which matches the answer 1 or contradicts the answer 2. The following propositional logic proof derives a contradiction for the second (incorrect) answer A2.

A2: E argue that hinder(A, B)
Q: NOT_oppose("educationist", "large classes") *question frame with REASON_WHY removed*
K1: NOT_improve(X, "children schooling") -> oppose("educationist", X)
K2: hinder(X, Y) -> NOT improve (X,Y)
Step1: *From A2, K2* : E argue that NOT_improve("large class", "children schooling")
Step2: *From K1 & Step1*: oppose("educationist", "large class")
Step3: *From Q and Step2:* NOT_oppose("educationist", "large classes") AND oppose("educationist", "large class")
CONTRADICTION

Figure 2: Logical proof derives a contradiction
A2 is the semantic representation of the second (incorrect) answers
K1, K2 are derived from the knowledge base

The logical proof finds contradiction between the incorrect answer and the statement of the question and allow the system automatically reject an incorrect candidate.

Conclusion

In this paper I presented an approach to automatic question answering of *why* questions from text . The approach identified and matched the semantic frames in a question and candidate sentence detecting semantic relations between the elements of the frames. Question answering is a complex task that benefits from utilizing many other technologies. I identify the technologies necessary to answer *why* questions:

- Word sense disambiguation for the frame verbs
- Co-reference resolution for pronouns
- Paraphrasing is beneficial to achieve more human-like answers.
- Natural language generation module converts the semantic frames to the lexical surface representation.
- Logical provers allow the system to validate the potential answers.

The approach utilizes the following domain knowledge for answering *why* questions:

- Ontology for the identification of semantic relations between the frame attributes (*synonym, meronym, hypernym* relations) allows matching attributes that are not exactly lexically equivalent.
- Ontology of *implication* semantic relations for the verb predicates is necessary when extracting an answer from text without causation cues.
- Paraphrase dictionaries could be used for deriving paraphrasees

In this paper I analyzed several sample question-answer pairs. This analysis is not exhaustive, similar analysis should be applied to identify more types of question/answer pairs. The next step is to build an automatic question answering system utilizing existing technologies (word sense disambiguation, ontology, logical provers, co-reference resolution, natural language generation, and paraphrasing) and test this approach automatically on the data. Some of these technologies are available off-the-shelve and achieve reasonably high performance, while others are still in the research stage.

Semantic role labeling technology is used to extract partial semantic parse of a sentence. Currently it has limitations, as it only considers verb predicates. I think that full semantic parsing is the next step in the NLP technology that would help to improve many applications including question answering.

References:

S. Stenchikova, et. al. *QASR:Question answering using Semantic Roles* 2005
S. Verberne *Data for question answering: the case of why.* LREC 2006
C. Fellbaum *WordNet* ***An Electronic Lexical Database*** MIT Press 2001
E. Vorhees 2002, 2003, 2004, 2005, Overview of the Text Retrieval Conference
R. Girju Automatic detection of causal relations for Question Answering ACL 2003
M. Collins Dissertation, 1999
M. Marcus *Building a large corpus of English: the Penn Treebank* 1993 Computational Linguistics

D. Gildea, D. Jurawsky *Automatic labeling of semantic roles* 2002 Senseval
Framenet *http://framenet.icsi.berkeley.edu/*
B. Baldwin *CogNIAC: High Precision Co-reference with Limited Knowledge and Linguitic Resources* 1997 ACL/EACL
P. Kingsbury, M. Palmer *PropBank: the Next Level of TreeBank*
D. Roth Semantic Role Labeling Demonstration http://l2r.cs.uiuc.edu/~cogcomp/srl-demo.php
N. Ide, J. Veronis *Word Sense Disambiguation: The State of the Art* Computational Linguistics 1998