Natural Language Engineering: page 1 of 28. © Cambridge University Press 2012
doi:10.1017/S1351324912000289

Modeling human newspaper readers: The Fuzzy Believer approach

RALF KRESTEL[1], SABINE BERGLER[2] and RENÉ WITTE[2]

[1]*Department of Computer Science*
University of California, Irvine, CA, USA
e-mail: `krestel@uci.edu`
[2]*Department of Computer Science and Software Engineering*
Concordia University, Montréal, Canada
e-mail: `bergler|rwitte@cse.concordia.ca`

(*Received 6 February 2012; revised 6 September 2012; accepted 7 September 2012*)

Abstract

The growing number of publicly available information sources makes it impossible for individuals to keep track of all the various opinions on one topic. The goal of our *Fuzzy Believer* system presented in this paper is to extract and analyze statements of opinion from newspaper articles. Beliefs are modeled using the fuzzy set theory, applied after Natural Language Processing-based information extraction. The Fuzzy Believer models a human agent, deciding what statements to believe or reject based on a range of configurable strategies.

1 Introduction

News articles are everywhere: in print, on-line, and in social media. Available from different publishers and in many languages, they convey information on developing events worldwide, politics, economics, cultural happenings, and social developments. While on-line access to other information sources provides easy alternatives for certain information needs, the continued importance of newspaper articles stems in large part from the established coding of the source of the information and its (resulting) trustworthiness. Especially explicit in the North American tradition, most information that is not absolutely certain is attributed to a source, usually in form of *reported speech* accompanied by further information about the circumstances of the communication of the reported information (Bergler 1992), as in *'We think this is the bottom year,' a Nissan official said*. This means that most information in newspaper articles is not reported as a plain fact, but embedded in a layer of information that each reader can interpret differently. The practical importance of distinguishing the factual from the embedding material in all genres of text has recently been underscored with several shared tasks in the biomedical domain (BioNLP Shared Tasks; Kim *et al.* 2011a, 2011b) and more generally in the QA4MRE pilot task.[1] Reported speech has traditionally made such embedding contexts explicit and is thus an important beacon in this field. As thus it has been explicitly annotated in the European NewsExplorer in a ‘Quotes from’ section (Pouliquen, Steinberger and Best 2007), for instance, http://emm.newsexplorer.eu/NewsExplorer/entities/en/381.html.

With the proliferation of news sources, readers are required to select useful information from the abundance of content. Current tools for navigating on-line content[2] include traditional information retrieval (search with Google, Bing, etc.), but increasingly include more targeted, deeper text analysis for special purposes. Thus, BioNLP tools include linking of search terms with ontologies (Doms and Schroeder 2005) and automatic term expansion using MeSH (Nelson, Johnston and Humphreys 2001). For newspaper content, automatic news aggregation is illustrated by Columbia’s Newsblaster system, which compiles similar newspaper articles from different sources and presents *multi document summaries* grouped into standard newspaper categories and dynamically created topic subcategories (McKeown *et al.* 2002). More recent enhancements include browsing news from multiple languages from multiple sites on the Internet with synthetic English summaries (Evans, Klavans and McKeown 2004). The effectiveness of automatically generated news summaries for user task performance was evaluated and confirmed in McKeown *et al.* (2005). Competitor systems are providing different extensions on the same type of services, see Google News’ personalized recommender system, which uses the click history of multiple users over different items together with a specific user’s click history set to recommend stories to the user (Das *et al.* 2007), or the European NewsExplorer http://emm.newsexplorer.eu/NewsExplorer/home/en/latest.html.

These services provide tools for general browsing and information retrieval, designed to give a quick overview over topics and articles with links to the actual documents. Incremental summarization of multiple articles on a developing story with emphasis on avoiding repetition of already presented material has been emphasized in National Institute of Standards and Technology’s Text Analysis Conference summarization update task since 2008 (Dang and Owczarzak 2008).

This level of personalization is of even greater importance in the increasing market of reduced screen electronics (cell phones, iPods, etc.), where browsing becomes more cumbersome. Recommender systems exist in many different areas to automatically forward content on topics marked as interesting. Combining these ideas of news summaries, recommender systems, and adding the (optional) notion of a personal profile of already held beliefs, we present here the idea of a personalized news reader, which updates a belief base with information from newspapers in accordance with different *belief strategies*.

The notion of an artificial believer was introduced by Ballim and Wilks (1991) in the context of constructive dialog. More recently, Prabhakaran, Rambow and

[1] QA4MRE pilot task, http://celct.fbk.eu/ResPubliQA/index.php?page=Pages/modalityTask.html

[2] Most print newspapers are also available online and we will consider only online access for the remainder of this paper.

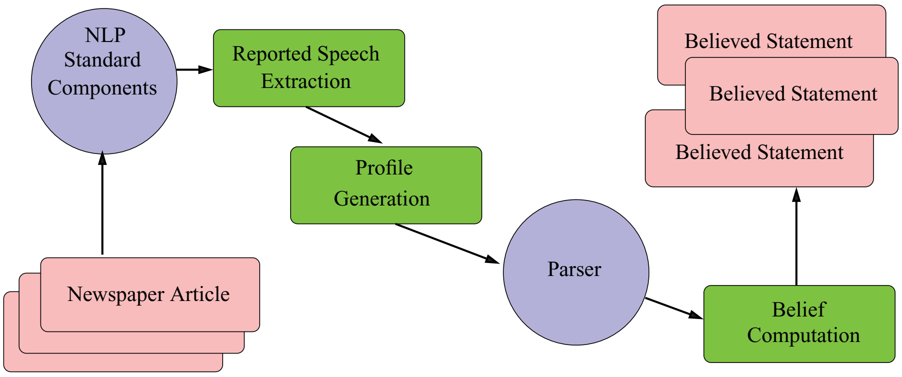


Fig. 1. (Colour online) Fuzzy Believer system structure.

Diab (2010) distinguish belief from intentions and desires but focus on automatically annotating the intentions of authors by distinguishing committed belief from non-committed belief, based on a corpus developed in Diab *et al.* (2009).

Our *Fuzzy Believer* system models human newspaper readers who develop their own point of view for current events described in newspaper articles. Most notably, Fuzzy Believer is not gullible, the system requires information about the source and other circumstances around a news statement to form its own beliefs. The system relies only on information stated within the grammatical construct of *reported speech*, a clear assignment of statements to sources[3], thus enabling the Fuzzy Believer to judge according to different degrees of reliability in a source. This focus on explicitly modeling all information journalists give us goes beyond the capabilities of the most current systems. It is our belief that an accurate model of interpreting source information in newspaper articles serves as a useful default model for combining information from multiple sources in general, be it for customer reports regarding different vendors (product reviews) or for advice from question-answer bulletin boards, like OnStartups (http://answers.onstartups.com/). We propose to advance this matter by providing an implemented case study here.

Our approach illuminates the penalty incurred in the added complexity when addressing two different problems at the same time, which are usually dealt with in isolation: opinion extraction/mining and textual entailment. The first step identifies and normalizes the different types of information conveyed in reported speech sentences to extract information that has to be assessed for reliability. The second step compares this new text with information already in the belief base or with text from other sources: the textual entailment component can do both tasks. Solving both tasks is necessary to implement an artificial believer, and we present our model as an exploration into the added complexity required.

An overview of the system's structure is shown in Figure 1, indicating the three main components constituting our system:

[3] A version of our reported speech tagger is available for download at http://www.semanticsoftware.info/reported-speech-tagger.

(1) Reported speech handling (extraction and profile generation).
(2) Predicate–argument structure extraction from different parsers.
(3) Belief computation using various strategies.

Evaluating such a complex system has to be done in several steps, in order to compare to the state of the art. We present piecewise evaluations of subcomponents on the most closely related comparable datasets.

To summarize, the system we present in this paper addresses various problems within the Natural Language Processing (NLP) domain. Our main contributions are as follows: (1) Developing rules to identify and extract reported speech from newspaper articles; (2) personalizing this information by applying the fuzzy set theory to NLP; and (3) creating a working implementation of these ideas together with an evaluation environment.

2 Background

To address the problem of information overload head on, our system presents an attempt to select information based on a user-specific assessment of the expected reliability of different information. The task of the system can be described as the simulation of a human newspaper reader, modeling the different ways humans approach newspaper articles – or more precisely, what a human *believes* after reading a newspaper article: Not every person believes everything he or she has read. Each person has a different background, different knowledge, and different preferences.

Usually the conscious belief process does not start with reading a newspaper article, but already by choosing the newspaper to read. Most newspapers are known to hold certain views on political, economical, or social aspects, reflected in the way they report events. The reporter can introduce specific subjective opinions about a topic into the article by choosing different styles of writing and using special language. A very common way in anglophone newspapers is to use reported speech to express opinion while at the same time attributing them to a third party. This ensures a more objective style and limits the implicit influence of the reporter on the article.

The reporter has to decide whom to cite, thereby highlighting specific views. We define as *basic belief* a simple proposition that does not have another belief as an argument. We decided to limit the possible basic beliefs here to the complements of reported speech in newspaper articles. This reflects the role of reported speech to express peculiar or distinctive opinions or beliefs while clearly ascribing the source holding the opinion expressed in the statement. *Evidential analysis* (Bergler 1992) maps this linguistic coding into a degree of reliability for the relayed opinions.

Previous Work. Bergler *et al.* (2004, 2005) presented a precursor system extracting reported speech from newspaper articles together with its source and reporting verb. The system passed the extracted information through evidential analysis and separated the results into different *profiles*. A *basic profile* consists of a statement and its source. Intermediate, *merged profiles* exploit coreference information – from a noun phrase (NP) coreferencer (Witte and Bergler 2003) – to gather statements of the same source in one (merged) profile.

Gerard (2000) used the idea of profiles and fleshed out the *percolation* algorithm, which was first presented in Ballim and Wilks (1991) and Ballim, Wilks and Barnden (1991) to attribute nested beliefs to their sources. Modeling different degrees of confidence an artificial reader attributes to a certain newspaper, a certain reporter, and a certain source, a dichotomy of held beliefs and *potential* beliefs was introduced. This dichotomy is related to the recently introduced notion of *committed belief* (Diab *et al.* 2009).

The Fuzzy Believer presented here extends these models with state of the art content analysis of the reported statement itself, thus presenting a complete system while also improving the *extraction of reported speech* itself. The system presented in this paper is capable of *identifying the topic* of the reported speech and for each topic the *polarity of individual statements* concerning the topic. In addition, an *artificial believer* model is implemented, simulating knowledge acquisition through different *trust strategies.*

3 System overview

Our central approach is to use fuzzy set theory in NLP for an explicit modeling of fuzziness inherent to natural languages. This also enables the user to control the system's behavior by varying various runtime parameters responsible for the fuzzy processing. Reported speech statements present the basic set of beliefs for our system. These kinds of statements usually express a belief held by the source of the statement, providing a clear attribution of the statement to this source. The extracted reported speech structures are parsed and dependency structures are transformed into predicate–argument structures (PAS). Each predicate–argument structure defines a basic belief, which the system eventually either accepts or rejects. The reported speech structures form the foundation for our heuristic fuzzy belief processing.

3.1 Architecture

Mirroring the different processing steps, our Fuzzy Believer system consists of a set of components running consecutively. It is implemented using the General Architecture for Text Engineering (GATE) (Cunningham *et al.* 2011), which offers an open source framework for developing NLP applications. For preprocessing, we use a number of standard components shipped with GATE, e.g. ANNIE to do entity recognition, which comes with a set of different gazetteer lists to detect named entities. For high-level processing we developed our own components. The input to our system is a selection of newspaper articles. Different components are used to realize specific tasks within the system to process the input documents, as can be seen in Figure 2.

After preprocessing an input document, a first important step is to identify noun phrases. These structures are important for our task to identify acting entities, e.g. persons within a text. We do full noun phrase coreference resolution, making use of

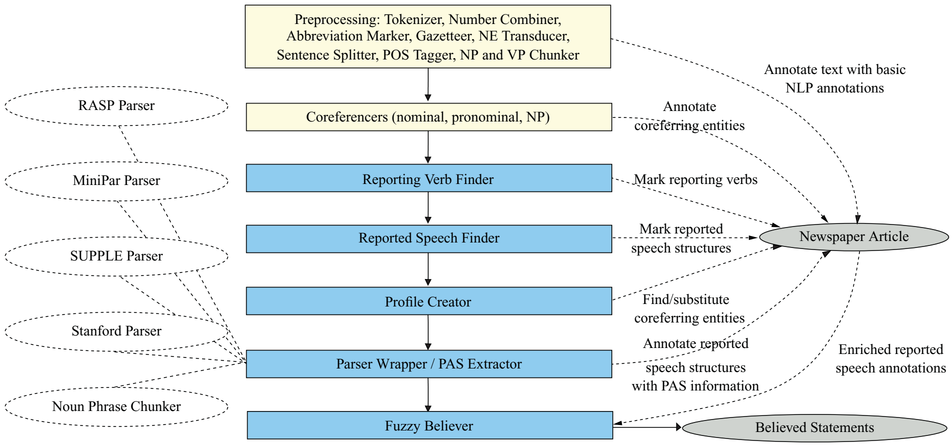


Fig. 2. (Colour online) Overview of the Fuzzy Believer system components.

an existing coreferencer (Witte and Bergler 2003). The next step is to identify and *extract reported speech* within the document, combining the results of the last two steps (see Section 4). The coreference component can identify the same source of two different reported speech utterances enabling us to *build profiles*.[4]

The core of our system is the processing of the information encoded in the profiles. We use external parsers to extract PAS as a basis for further processing (Krestel, Witte and Bergler 2010). Our focus lies thereby on the analysis of the extracted PAS of the reported speech utterance and the generation of held beliefs from it in the last step: statements from newspaper stories are *potential beliefs* that will only be made into *held beliefs* if they do not contradict previously held beliefs.

Determining whether two statements are paraphrases of each other, whether they contradict each other, and whether they are partially related or wholly unrelated is an area of great interest for many applications and it has been formalized for the task of determining *textual entailment* (Dagan, Glickman and Magnini 2005; Bar-Haim *et al.* 2006). Determining the *supporting group structure*, that is the grouping of statements by different sources that support each other and contrasting those that are opposed, is a subtask of the general textual entailment task and we have tested our methods in the textual entailment challenges of 2008 and 2009 (Krestel, Bergler and Witte 2008a, 2009). Our approach uses the fuzzy set theory and WordNet (Fellbaum 1998) to tackle this question.

The final step, after trying to 'understand' what has been said and by whom, is to define what the system should actually retain as potential or held beliefs. The Fuzzy Believer has to evaluate the created belief structures (previously held as well as all that were derived from texts). To model different human 'believers,' the Fuzzy Believer component uses different *believe strategies*. The result of the system is a set of propositions the system 'believes,' and a set of propositions the system rejected.

[4] We are referring here to the notion of *profile* introduced in Bergler (1992, 1995a) and Krestel, Witte and Bergler (2007b).

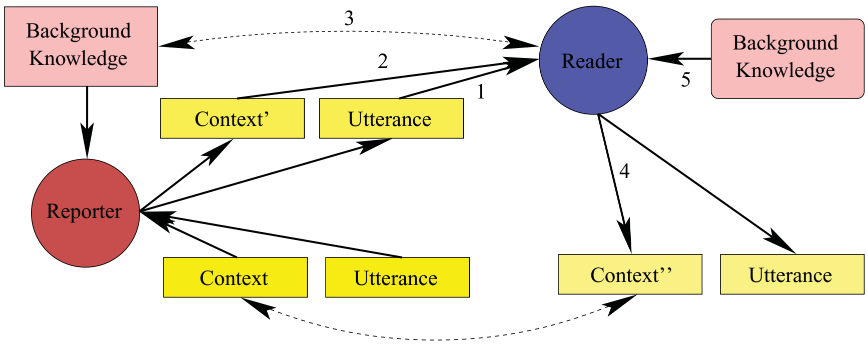


Fig. 3. (Colour online) Steps in reported speech analysis.

The next sections describe the main components of our system, together with conducted experiments and evaluation results.

4 Reported speech extraction

Newspaper articles are the result of a two-step process: information is conveyed from a *source* to a *reporter*, and from the *reporter* to a *reader*. A source's utterance and its context will be interpreted by the reporter, encoded by the reporter for the reader, and decoded by the reader. The reporter can use the mechanism of reported speech to not only reproduce the content of the utterance but also to put it into context. This process involves mutual assumptions about the three actors' beliefs and background knowledge.

Consequently, from the reader's point of view, reading a newspaper article is a multilevel process, as illustrated in Figure 3. In order to interpret reported speech, the reader has to

(1) understand the content of what is expressed in the article (the reported clause);
(2) evaluate the additional information given by the reporter to reconstruct not only the original utterance but also the original context (the reporting clause);
(3) interpret the reporter's presentation;
(4) reconstruct the original situation; and
(5) interpret the assumed original situation.

This encoding typically takes the form of *reported speech*. The use of *direct* and *indirect* speech is similar, with the distinction that in direct speech, the reporter commits to a literal transcription of the original utterance, given in quotes, whereas he gives a summary interpretation when using indirect speech.

4.1 Reported speech

Reported speech usually consists of the *reporting clause* and the *reported clause* (Quirk 1985). The reporting clause contains information about the source of the utterance, the circumstances in which it was made, and possibly a characterization of the manner or force with which it was made. Example (1)[5] shows a sentence in indirect speech, detailing its constituents (RV = reporting verb):

(1) $$\underbrace{\overbrace{\textit{Last October,}}^{\text{circumstances}}\ \overbrace{\textit{his brother Hubert}}^{\text{source}}\ \overbrace{\textit{told}}^{\text{RV}}\ \overbrace{\textit{the bankruptcy court}}^{\text{addressee}}}_{\text{reporting clause}}\ \textit{that}\ \underbrace{\textit{Paul was very ill}}_{\text{reported clause}}.$$

The reported clause can consist of direct or indirect speech. Paraphrasing Example (1) now with direct speech yields Example (2):

(2) *Last October, his brother Hubert said: 'Paul is very ill.'*

There is also a form of *free direct* and *indirect speech*. It is used, for example, to express a stream of consciousness in fictional writing. The reporting clause is omitted in that kind of reported speech. Free form is more widely used in German newspapers, for instance, but is virtually absent in the North American newspaper tradition (Bergler 1995b) and here we will concentrate on direct and indirect speech only.

Reported speech obeys a few, consistent patterns; quotation marks, for instance, indicate *direct speech* in newspaper articles:

(3) *To strong applause from the demographically diverse Arizona State University audience, Mr. Jackson declared: 'Let's change basic and fundamental direction.'*

The position of the reporting clause can vary: from the beginning of a sentence, as in (3); the middle, as in (4); to the end of the sentence, as in (5). When the reporting clause is located within or after the reported clause, subject and verb positions may be reversed, as in Example (4).

(4) *'If you want managers to gamble,'*
Michael Jensen said | said Michael Jensen,
'you've got to give them a chance to become really rich.'

In Example (5), the reporting clause is at the end:

(5) *'We think this is the bottom year,' a Nissan official said.*

Direct speech can span over more than one sentence. In that case, the reporting clause is usually found within the first sentence. As a grammatical relation, the direct speech can function as a subordinate clause, for example in (3). But it can also be a subject complement, an apposition to a direct object, or an adverbial construct. A special case is the mixture of direct and indirect speech, where the direct speech forms only part of the reported clause:

(6) *In a televised address, the president concluded that the initiative 'was a mistake.'*

[5] All examples are taken from the *Wall Street Journal* (WSJ) (1986–1989).

Table 1. *Six patterns for finding reported speech in newspaper articles*

Source	Verb		Content
Verb	Source		Content
Content	Source		Verb
Content	Verb		Source
Content	Source	Verb	Content
Content	Verb	Source	Content

In newspaper articles, *indirect speech* is ubiquitous. Like direct speech, it relates information from a source, but in a summary form, designed to convey the essence of a larger discourse. It can also contain *circumstantial information* that indicates additional features of the context of the original utterance. Consider this example, where 'as a result' puts the information of the reported clause in the necessary context:

(7) *As a result, the company said that it will restate its 1986 earnings.*

4.2 Extracting reported speech

It is crucial not only to identify reported speech sentences but also to mark the different elements for further analysis. We focus on reported clauses in the shape of declarative sentences, excluding here certain reported speech structures with reported clauses that do not form a grammatically correct statement on their own. This work is based on Krestel Bergler and Witte (2008b) with a focus on automatic tagging of reported speech in newspaper articles.

Identifying reporting verbs is done by the reporting verb marker, which tags verbs used to express reported speech using a finite state transducer. This component was first developed and implemented by Doandes (2003) to extract information for an evidential analysis of reported speech (Bergler 1995a). The reporting verb marker is implemented using GATE's *Java Annotation Patterns Engine* (JAPE) (Cunningham, Maynard and Tablan 2000). It works with the chunker notion of *verb groups*, contiguous sequences of auxiliaries and verbs. When one of the listed verbs is detected as the head of a verb group, the reporting verb finder marks it as a reported speech verb by adding a corresponding annotation containing the lemma of the reported speech verb.

To identify reported speech in newspaper articles, we extract six general patterns. These differ in the position of the reporting verb, the source, and the reporting clause. An overview of these six patterns is shown in Table 1. Identifying reported speech sentences enables us to label different elements for further analysis. Our components have been designed to allow extracting statements in form of declarative sentences. Currently, we exclude structures where the reported clause is not a grammatical sentence, since infinitival and other omitted constructs no longer report the speech of others, but interpret their actions or utterances, which requires a different treatment, for example:

(8) *The President denied signing the bill.*

The six patterns are not exhaustive, for example, they will ignore the second source and reporting verb in Example (9):

(9) `Mr. Coen` *predicted* that a weak sector in 1987 will be national print – newspapers and magazines – which `he` *said* will see only a 4.8% increase in advertising expenditures.

Constructs that do not fit into our six basic patterns are rare – numbers depend on the literary style of the newspaper, but are around 3 percent – and additional patterns can easily be added.

4.3 Generating profiles

Reported speech structure analysis extracts source noun phrases for the reported speech sentences, and we have to resolve whether two source noun phrases in different sentences describe the same entity at different degrees of abstraction. For example, we could put all 'officials' into one group or treat 'administration officials' and 'law enforcement officials' separately. This is done using coreference chains computed by a fuzzy system (Witte and Bergler 2003). For instance, 'The president,' 'Mr. Reagan,' and 'he' may refer to the same entity in some article, licensing a merging of the three reported statements into one profile. Our profile generation builds upon Gerard (2000).

4.4 Evaluating the reported speech extraction component

In order to evaluate the reported speech extraction component, we randomly picked seven newspaper articles (∼6,100 words) from the WSJ corpus and created a gold standard containing the reported speech elements: *source*, *reporting verb*, and *reported clause* (that is we did not evaluate the detection of circumstantial information). The articles contain about 400 sentences and among them 133 reported speech constructs.

Apart from correct and incorrect identification of reported speech, we also measure partial correctness: If the system annotates a reported speech sentence nearly correctly with the exception of one or two terms of circumstantial information, for instance, we speak of partially correct detection as long as the meaning of the reported speech in general is maintained.[6]

For the detection of reporting verb and source, i.e. partial correctness, our system achieves a recall value of 0.79 and a precision value of 1.00, thus an F-measure of 0.88. The results for the reported clause, together with a detailed overview of the results obtained for the different test documents, can be seen in Table 2. The results for the extraction of the reported clause (content) suffers mostly from the misinterpretation of parts of reported clauses as circumstantial information.

[6] This is akin to the textual entailment task in National Institute of Standards and Technology's Recognizing Textual Entailment challenges, or the task of determining whether a sentence from an article is novel and should be included in an incremental extractive news summary as defined in National Institute of Standards and Technology's Text Analysis Conference
challenges.

Table 2. *Reported speech extraction results on inhouse gold standard*

	Reported clause			Source/verb		
WSJ article	Precision	Recall	F-measure	Precision	Recall	F-measure
861203-0054	1.00	0.50	0.67	1.00	0.63	0.77
861209-0078	1.00	0.77	0.87	1.00	0.79	0.88
861211-0015	0.97	0.88	0.96	1.00	0.89	0.94
870129-0051	1.00	0.71	0.83	1.00	0.71	0.83
870220-0006	0.96	0.74	0.84	1.00	0.93	0.96
870226-0033	1.00	0.58	0.74	1.00	0.58	0.74
870409-0026	1.00	1.00	1.00	1.00	1.00	1.00

5 Predicate–argument extraction

Predicate–argument structures seem to be the core of the semantic structure of all human languages (Jurafsky and Martin 2008). The arguments of a predicate are not arbitrary. A verb restricts the arguments that it can predicate over grammatically and semantically. These small semantic units are well suited for our purpose. We extract PAS from the output of different parsers: RASP (Briscoe, Carroll and Watson 2006), SUPPLE (Gaizauskas *et al.* 2005), MiniPar (Lin 1998), and Stanford Parser (Klein and Manning 2003a). Although the number of arguments can vary between different PAS, we limit the number to two (in the simplest case, these are the subject and object in a sentence). If more than two arguments are present in a statement, we use heuristics to identify the two most important ones. In the following, we show how the extracted reported speech statements are parsed and subject/verb/object triples (PAS) are subsequently extracted from the parser output.

5.1 Predicate–argument structures

Predicate–argument structures can be seen as a representation between the raw syntactic level and semantic role representations. Most verbs in English require a subject and a complement to be specified in a grammatical sentence. For simple sentences, this subject–verb–complement structure constitutes a complete analysis; for more complex sentences the task is to identify the predicate–argument structure, to assign the correct arguments to all verbs, and to identify adjuncts, i.e. prepositional phrases (PP) or noun phrases that are not in argument position (Merlo and Ferrer 2006).

Dependency parsers have been addressing this as a major issue for some time and some prioritize correct dependencies over achieving a complete parse for a sentence. Even full-fledged constituent parsers have lately offered a conversion module that transforms a parse tree into dependency notation, because these notations have been the most useful notations for different applications. Dependency relations are like severed components of PAS or adjunct specifications, but they do not make the complete event structure explicit and it is surprisingly complex to extract the underlying PAS from dependency parser output.

Table 3. *Predicate–argument structures generated by PAX for three different parsers*

RASP	MiniPar	Stanford	SUPPLE	MuNPEx	
Obama	Obama	Obama	–	trip	Dalai
meet	meet	meet	meet	be	be
trip	Lama	Lama	Lama	five-day	Lama

5.2 Extraction from parsers' output

We use the output of different parsers to identify PAS. In addition, we make use of a noun phrase chunker[7] to extract PAS. As an example, consider the following sentence:

(10) *President Barack Obama will not meet the Dalai Lama during his five-day trip to the U.S. capital.*

The different output formats of the parsers make it difficult to compare the results with each other, but the fact that 'someone met the Dalai Lama' is covered by all parsers. Our Predicate–Argument eXtraction (PAX) component normalizes the different outputs into PAS as shown in Table 3. The parsers demonstrate different representations for the input sentence. This is not an exceptional, or special case, but typical for this task. Note that we chose a rather simple sentence to demonstrate the different outputs. For more complex sentence structures, the difference in output is even greater and the extracted PAS look quite different.

Our PAS extractor is based on a set of rules for each of the four parsers and the noun phrase chunker. These rules determine which part of the parser output is considered the subject, verb, and object. Because of the different nomenclature and relations scheme of the parsers, this has to be done individually for each parser. For our task at hand, partial PAS structures focusing on subject–verb–complement are sufficient and we do not elaborate on complete PAS extraction.

SUPPLE. For SUPPLE (Gaizauskas *et al.* 2005), the extraction process is quite straightforward. The parser outputs *semantic* relations, which comprise a logical subject and verb, and sometimes also a logical object. The PAS extractor therefore only selects these elements from the output of SUPPLE. The coverage of SUPPLE is lower in comparison with other parsers. This is due to the philosophy of the parser (Gaizauskas *et al.* 2005): *'Rather than producing all possible analyses or using probabilities to generate the most likely analysis, the preference is not to offer a single analysis that spans the input sentence unless it can be relied on to be correct. This means that in many cases only partial analyses are produced, but the philosophy is that it is more useful to produce partial analyses that are correct than full analyses which may well be wrong or highly disjunctive.'*

[7] Multi-lingual Noun Phrase Extractor (MuNPEx), http://www.semanticsoftware.info/munpex

MiniPar. To obtain PAS that represent the underlying sentence as closely as possible, we often have to choose between multiple candidates for the *object*. We employ a decision tree to select the grammatical structure to fill the *object* slot from the parser's output. If it exists and relates to the subject–verb pair, we choose in this order: 'obj,' 'obj1,' 'pred,' and 'pcomp-n.'

Sometimes the *object* does not have a direct relation to the *verb* but an indirect link through another element in common, like a 'mod' construct. In this case we have to track down and identify this relation to find a representative object. A complex sentence can contain more than one subject and our extractor has to be able to handle them reasonably. Besides dealing with more than one 's' (subject) in one sentence, it also handles simple conjunctions.

RASP. For RASP's version 3 (Briscoe *et al.* 2006) we developed a wrapper to be able to use it from within GATE. It calls the appropriate script and delivers the parser's output for further processing. The strategy to find subject, verb, and object relations is to look for 'ncsubj' occurrences in the parser output. They describe a subject together with the corresponding verb. To find a suitable object, we often have to choose between different elements like 'dobj,' 'iobj,' 'obj,' or 'xcomp.' To obtain PAS that accurately represent the underlying sentence, we use the following decision tree on what grammatical structure to use as *object*. If it exists and is related to the verb of the subject, we choose in this order: 'obj,' 'dobj' if dependent of an 'iobj,' which itself relates to the relevant *verb*, 'iobj,' 'dobj,' and last 'xcomp.' Besides dealing with more than one 'ncsubj' in one sentence, we can also handle conjunctions.

Stanford Parser. The Stanford Parser (Klein and Manning 2003a, 2003b) extracts dependency relations. We take all 'nsubj' and 'nsubjpass' elements for subjects and the associated predicates as verbs. For the object, we consider in this order: 'dobj,' 'prepPobj,' and 'dep.' Conjunctions are already considered by the parser and there is no further processing needed from our side.

MuNPEx Noun Phrases. Each noun phrase that contains a modifier is a candidate for a predicate–argument structure. For example, the noun phrase 'the rich king' contains the same information as the PAS 'king – be – rich.' Adding the noun phrase predications generates additional PAS that can be especially useful for certain tasks, like comparing documents' content based on PAS or for recognizing textual entailment between statements, as in the Recognizing Textual Entailment tasks (Bar-Haim *et al.* 2006).

5.3 Evaluating the predicate–argument extraction component

To evaluate our Predicate–Argument EXtraction component, we selected an article from the WSJ and annotated it manually with PAS. The structure of the sentences was particularly complex, with three or more PAS per sentence in most cases. For simple sentences of the shape 'subject, verb, object' all parsers perform well and we

Table 4. *Results for the four parsers C = correct, P = partially correct, F = false*

Sent	No. of PAS	SUPPLE			MiniPar			RASP			Stanford		
		C	P	F	C	P	F	C	P	F	C	P	F
1	4	–	2	–	–	2	–	1	2	1	1	2	1
2	4	1	–	–	–	2	–	2	2	–	2	2	–
3	3	–	–	–	–	2	–	2	1	–	1	1	–
4	3	–	–	–	–	–	1	2	–	–	–	–	–
5	4	–	1	–	–	2	–	–	2	1	–	2	–
6	1	–	1	–	–	1	–	–	1	1	–	1	–
7	4	1	–	–	–	2	–	–	3	–	–	4	–
8	5	2	1	–	3	1	–	–	4	1	3	1	–
9	3	–	–	–	–	1	2	1	2	1	1	1	–
10	3	–	1	–	–	1	–	–	–	–	–	1	1
11	6	1	2	–	3	1	–	2	4	–	1	1	2
12	3	1	–	–	–	2	–	–	3	1	–	–	–
13	5	1	1	–	2	1	–	1	2	–	1	1	–
14	2	–	–	–	–	1	–	–	2	–	–	1	–
15	3	–	1	–	–	1	–	–	2	–	–	2	–
16	2	–	1	–	–	–	–	–	1	2	–	1	1
17	4	1	1	–	–	3	–	–	4	–	1	1	1
18	3	–	–	–	–	2	–	–	3	–	–	3	–
19	3	1	–	–	–	1	–	2	–	–	1	1	–
20	3	–	1	–	–	2	–	1	–	2	1	2	–
21	2	–	1	–	–	2	–	–	2	–	–	2	–
22	1	–	1	–	–	1	–	–	1	–	–	1	–
23	4	–	–	–	2	–	1	2	1	1	–	1	1
24	0	–	–	–	–	–	–	–	–	–	–	–	1
$\sum$	75	9	15	–	8	31	3	16	42	11	13	32	8
Recall			0.32			0.52			0.77			0.60	
precision			1.0			0.93			0.84			0.85	

can extract PAS reliably from the parsers' output. Therefore, we are interested in the most difficult cases only. We excluded the noun phrase PAS extraction from this evaluation, since it is a special case also yielding different types of errors. Table 4 gives an overview of the performance of different parsers with correctly extracted PAS (column C), wrong PAS (column F), and partially correct PAS (column P), where 'partially' means for example that the object was not found or an indirect object instead of a direct one was selected. This evaluation is not meant as a general performance assessment of the parsers but only mirrors the convenience of these parsers for the task of PAS extraction for one particular newspaper article.

Some errors such as unresolved pronouns, e.g. 'that,' 'he,' 'who,' or 'myself' were not considered errors of the PAS extraction but need to be dealt with in the future, although for some parsers we are already able to resolve these constructs. Another possible source of errors are verb phrases (VP) like 'declare unconstitutional,' or 'prevent s.o. from doing s.th.' If we insist on having only one term as a predicate, we need to decide which verb reflects the intended meaning of the PAS best.

Noun phrases with modifiers cannot always be converted to PAS. For example, it works fine with 'the elected President' $\longrightarrow$ 'President – be – elected'; but not for 'last year's President' $\not\longrightarrow$ 'President – be – year.'

6 Finding topics and identifying the polarity of statements

The next step in our processing pipeline is to gather the generated profiles according to their topic in an attempt to determine the argumentative structure of the article. In articles with a single source, this will merely delineate topic boundaries, but in articles based on information from several sources (that might possibly be contradictory), this is an important grouping to determine the scope of disagreement.

Because multiple quoted statements can be offered as possible beliefs to the reader, it is not appropriate to simply select one. Their respective content and credibility have to be compared. Since topics do not have clear boundaries and one statement may address multiple topics, we formally represent the statement/topic relationship using fuzzy sets (Zadeh 1965), i.e. we employ a soft computing approach as discussed in Witte and Bergler (2003).

The topic-finding component uses the generated PAS to group the statements of a newspaper article according to common topics. In order to build a fuzzy representation, we need to set a few constraints: a single (atomic) fuzzy set represents all statements for a single topic. Processing these fuzzy sets takes place in the following four steps:

- Grouping statements into topics.
- Finding a fuzzy representation for the statements.
- Identifying the polarity of statements.
- Computing beliefs according to a strategy.

Due to the very small amount of text for the topic-detection phase, this component uses two simple heuristics that compare the extracted PAS of two statements: one is syntactic, based on string similarity, and the second is semantic, based on WordNet. Before we describe in detail how statements are represented and processed, we briefly review foundations from the fuzzy set theory.

6.1 Fuzzy set theory background

Natural language is intrinsically vague and fuzzy. Rather than attempting to artificially represent such imprecise information with classical crips sets, the idea of soft computing is to explicitly model them, for example, based on the fuzzy set theory (Zadeh 1965). The most basic concept of the fuzzy set theory is that set members can have a *degree* of membership, rather than a binary relation: formally, a fuzzy set is characterized by a membership function μ_A, which maps the members of the universe Ω into the unit interval [0,1]:

$$\mu_A : \Omega \rightarrow [0, 1].$$

The cases where $\mu_A(x) = 1$ and $\mu_A(x) = 0$ correspond to classical set membership ($x \in A$) and non-membership ($x \notin A$).

In our artificial believer setting, we model the fact that a statement belongs to a certain topic: Instead of making a (possibly incorrect) decision early on, fuzzy sets allow us to explicitly model the degree of *certainty* that a statement belongs to a given topic (represented by a fuzzy set); and one statement can also belong to multiple topics, with a different membership degree for each.

In this model, a single fuzzy set, a fuzzy *atom*, only captures a single statement, obtained from a predicate–argument structure. To model complex information, we can combine these to a *clause* using disjunctions, which in turn can be joined by a logical *and* to form a (fuzzy) formula in conjunctive normal form. This hierarchy constitutes a very flexible way to attribute different propositions to entities.

Logical *and* and *or* can be expressed in terms of fuzzy set intersection and union, respectively. However, when applied to vague and possibly inconsistent input data, as in our case, these operations will eventually lead to an empty fuzzy set (membership of zero for all elements) in case of intersection, or a meaningless set where all members have a degree of one in case of union. Hence, in our approach, we employ fuzzy *belief revision* operators (Witte 2002) that help us maintain a consistent belief base.

Stated briefly, fuzzy *expansion* allows to add new information to an existing formula, but only if the resulting formula achieves a minimum consistency degree γ – otherwise the new information is rejected. Thus, expansion is a monotonous operation, as the belief base can only grow or remain unchanged. Belief *revision*, on the other hand, will always add the new information, but if necessary restore consistency by removing conflicting information from the belief base. To decide which of the conflicting information should be removed first, we can supply a preference order to the belief revision operator (Witte 2002). Revision is a nonmonotonic operation, as the belief base can shrink when a new piece of information is added.

In our system, we leverage this fuzzy representation for finding the topic of a predicate–argument structure and also within each topic for managing the belief base according to a belief strategy.

6.2 Grouping statements into topics

Grouping statements into topics is a necessary step to identify unanimous or contradicting statements dealing with the same topic. Having extracted PAS in the previous step, we can now decide whether two statements deal with the same topic by looking at their PAS representations. For example, we want to determine whether the PAS 'Mr. Preisig – work – consultant' deals with the same topic as 'Mr. Preisig – do – consult.' Obviously, requiring PAS to match 1-to-1 would be too restrictive in this context. Therefore, we employ the fuzzy set model introduced above to provide for fuzzy matching between two PAS. Our framework allows to easily plug in different heuristics to compute the similarity between PAS, which is then interpreted as the fuzzy set membership degree. We have currently implemented

Table 5. *Two heuristics to group similar predicate–argument structures into topics*

Predicate–argument structures		String similarity	WordNet similarity
Mr. Preisig	Mr. Preisig	1.0	0.0
work	do	0.0	0.6
consultant	consult	0.7	0.0

two heuristics, one based on string similarity (syntactic matching) and another based on WordNet (semantic matching). The string similarity takes into account character overlap and matches, e.g. plural and singular terms. To capture synonyms, hypernyms, etc. we use WordNet and its graph structure to compute a similarity distance. If the heuristics recognize a fuzzy similarity degree higher than a given threshold between two statements, then they are grouped into one topic.

WordNet Similarity Heuristic (WNH). The WordNet similarity computes a similarity score between 0.0 and 1.0, based on semantic similarity. For two terms, the minimal length in the WordNet hierarchy is computed as shown in Algorithm 1.

The *maxSemanticDistance* parameter can be set by the user at run-time to establish a minimum semantic relatedness for two terms to match.

Algorithm 1 Fuzzy WordNet Heuristic

$maxSemanticDistance \leftarrow 5.0$
for all $wordNetSense1 \in getWordNetSenses(word1, PosType)$ **do**
  for all $wordNetSense2 \in getWordNetSenses(word2, PosType)$ **do**
    $distances.add(getPathLength(wordNetSense1, wordNetSense2))$
  end for
end for
$distance \leftarrow Min(distances)$
if $distance >= 0$ and $distance < maxSemanticDistance$ **then**
  $level \leftarrow (maxSemanticDistance - distance)/maxSemanticDistance;$
end if
return $level$

String Similarity Heuristic (STH). The string similarity computes a similarity score between 0.0 and 1.0, based on substring matching (Algorithm 2). This heuristic is particularly useful for proper nouns that do not occur within the WordNet database. The score of this heuristic depends on the character overlap of two words, thus a perfect match is not necessary to gain a fuzzy score.

These two heuristics are used to define the membership degree in the corresponding fuzzy set. An example showing the two heuristics at work is shown in Table 5. We combine the two heuristics by taking the maximum value: $max(level_{STH}, level_{WNH})$.

Algorithm 2 Fuzzy String Heuristic (STH)

$i = word2.indexOf(word1)$
$j = word1.indexOf(word2)$
if $i >= 0$ **then**
  $level \leftarrow word1.length/word2.length$
end if
if $j >= 0$ **then**
  $level = word2.length/word1.length$
end if
if $word1.equals(word2)$ **then**
  $level \leftarrow 1.0$
end if
return $level$

Grouping Process. With these heuristics at hand, topic classification becomes a matching process where we look for similar PAS among our extracted sentences. We can plug these heuristics into our fuzzy set framework to group the statements into topics by computing matching scores for each pair of PAS. The detailed algorithm can be found in Algorithm 3. To ensure that we compare the appropriate predicate–argument structure components, passive and active voice have to be distinguished by analyzing the verb group and exchanging syntactic subject and object for passives.

Requirements for two PAS to match are that at least two of the three element pairs have a matching score of at least the defined fuzzy threshold. This threshold can be set as a run-time parameter, which allows for more strict or more lenient topic classification as needed. In addition, the overall matching process is controlled by two alternative modes, strict or non-strict.

Strict topic classification demands that a predicate–argument structure is only added to a topic if it matches all other PAS within that topic. A run-time parameter allows to switch to *non-strict* mode, resulting in larger and broader topics. This more lenient topic classification strategy requires only one predicate–argument structure from the topic to match. This way, we allow transitive relations between the elements of one topic: Consider the following situation, where we have statement S_1 with the extracted PAS u, v, and w (S_1:u-v-w); a second statement (S_2:u-v-x); a third statement (S_3:u-y-x); and a fourth one (S_4:z-y-x). S_1 and S_4 have no PAS element in common, but with the lenient topic classification and S_2 and S_3 they will be placed in the same topic because the similarity of S_4 and S_3 is sufficiently high, the same holds for S_3 and S_2, and finally for S_2 and S_1. For our running example, the topic assignment results after using the heuristics can be found in Figure 4, step 4 (the fuzzy sets for this step are not shown).

Topic Representation and Polarity Detection. By splitting up the topic classification and the polarity-detection process, we can use different thresholds for the fuzzy assignment of statements to different topics and better discover supporting and

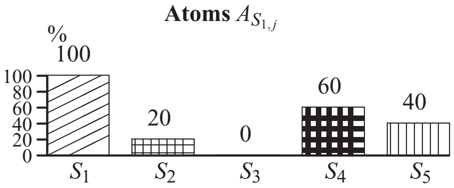
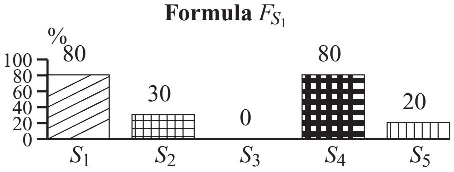
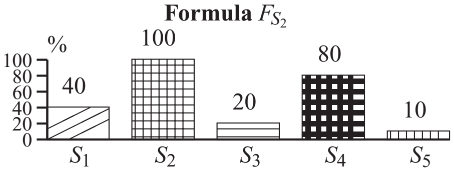
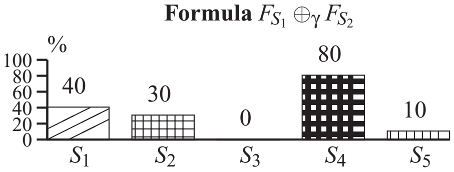
Step	Example	Description
1.	*'Preisig worked as a consultant,' one of the employees said.*	Sentence in a newspaper article.
2.	[*Preisig worked as a consultant*](content) [*one of the employees*](source) [*said*](reported verb)	Reported speech structure identified.
3.	[*Preisig*](subject) [*work*](verb) [*consultant*](object)	Predicate–argument structure extracted from parser output.
4.	Preisig – work – consultant (S_1, $topic_{38}$)	Predicate–argument structure assigned to a topic.
5.	**Atoms** $A_{S_{1,j}}$ 	Atoms $A_{S_{1,j}}$ for predicate–argument structure S_1 with correlation grades for all statements in $topic_{38}$ ($S_1, \ldots, S_5$). $(S_i, S_j) = X$ indicates X percent possibility that S_i has the same meaning as S_j. Thus, here S_1 and S_2 probably have different meanings.
6.	**Formula** F_{S_1}  **Formula** F_{S_2}  **Formula** $F_{S_1} \oplus_\gamma F_{S_2}$ 	Fuzzy belief revision: Result of γ-revision with $\gamma = 0.8$ for the top two formulas on the left. The first formula represents the existing statements within a topic by combining the different atoms to form literals, then clauses, and then formulas of each statement. The second formula represents the new statement added to the topic. The resulting formula contains only clauses that do not contradict the new one, i.e. have a similarity degree ≥ 0.8. The result means that the system believes the new statement and all those older statements about the same topic that do not contradict it.

Fig. 4. Example for all processing steps.

conflicting statements. One statement can belong to more than one topic, which presents no problem for the fuzzy set representation.

After all statements have been assigned to topics using the fuzzy topic classification described above, we are now interested in whether the statements support or oppose each other within a topic. Therefore, we again make use of fuzzy modeling and represent every predicate–argument structure as a fuzzy set containing its degrees of similarity with other PAS within the same topic. Thus, each topic fuzzy set contains all PAS assigned to this topic.

Algorithm 3 Grouping Algorithm

```
TOPICS = {}
for all pas_i ∈ PAS do
    foundTopic ← false
    for all topic_j ∈ TOPICS do
        if match(pas_i, topic_j) then
            topic_j.add(pas_i)
            foundTopic ← true
        end if
    end for
    if foundTopic = false then
        createNewTopic(j + 1)
        topic_{j+1}.add(pas_i)
        TOPICS.add(topic_{j+1})
    end if
end for
return TOPICS
```

To identify opposing statements, the fuzzy representation of the PAS is evaluated. To compute the degree of opposition, we currently use two heuristics: A WordNet heuristic to identify antonyms, and a syntactic heuristic looking for negations. If these heuristics yield small values for the degree of similarity, the meaning of the two statements are considered opposing. A threshold makes it possible to decide whether two statements are similar enough to be considered as expressing the same polarity or are likely to contain opposing views. The polarity detection is currently implemented based on finding antonyms in WordNet or identifying simple negation patterns in the vicinity of a predicate–argument structure. For our example, Figure 4, step 5 shows the representation of one predicate–argument structure (S_1) that is an element of a topic containing five PAS ($S_1, \dots, S_5$). S_1 is likely to contradict S_2 and S_3, whereas S_4 and S_5 are more likely to have the same meaning as S_1. Note that step 6 in Figure 4 will be explained in Section 7.

6.3 Evaluating the topic-finding component

The evaluation of the topic-finding component includes the comparison of the results obtained with RASP, MiniPar, and manually annotated PAS. The test data we use is taken from the MSR Paraphrase Corpus (Dolan, Brockett and Quirk 2005) and comprises 116 paraphrase pairs. This corpus contains pairs of sentences which deal with the same topic but use different wording. This makes it a suitable test corpus for our topic-finding component. We treat all sentences as content of a reported speech construct. With two different configurations we obtain a recall of 81 percent and a precision of 52 percent. Detailed results, including manual PAS annotated test data, can be found in Krestel, Witte and Bergler (2007a). Table 6 gives an overview of the results. The configuration settings in the table mean, from left to right, Maximum WordNet Distance between (1) subjects, (2) verbs, (3) objects of two statements, and

Table 6. *Topic classification: recall and precision for different parsers*

	Recall			Precision		
Average	RASP	Minipar	Manual	RASP	Minipar	Manual
3-3-3-lenient	0.59	0.54	0.56	0.57	0.63	0.78
3-3-3-strict	0.59	0.50	0.55	0.63	0.75	0.85
5-5-5-lenient	0.70	0.60	0.62	0.29	0.39	0.29
5-5-5-strict	0.52	0.52	0.52	0.41	0.53	0.54
5-3-5-lenient	0.65	0.51	0.59	0.31	0.57	0.45
5-3-5-strict	0.59	0.58	0.52	0.56	0.41	0.61

(4) indicates whether a new statement has to match with one (lenient) or all (strict) statements within one topic.

The rather shallow semantic approach sets a practical limit to the achievable results as indicated by the comparison of the numbers for manually parsed PAS with the numbers obtained from the parsers. There is space for improvement on the side of the parsers as well as on the side of the PAS extractor. But a recall of 55 percent with a precision of 85 percent, as obtained for the best configuration of the system using manually parsed PAS, is an indication for the requirement of more and better heuristics to improve recall for the topic-finding component.

7 Computing beliefs

When presented with opposing opinions, human readers can either take one of the two sides or believe neither. Our Fuzzy Believer system has to model these different behaviors as different strategies when deciding which statements to believe. In each case, the result of processing newspaper articles is a set of held beliefs and a set of rejected beliefs. Our strategies used to model different human behavior are as follows:

(1) Believe everything.
(2) Believe old news.
(3) Believe new news.
(4) Believe majority.
(5) Believe certain source/reporter/newspaper.
(6) Believe weighted majority – a combination of strategies (4) and (5).

Let us take a closer look at one of the strategies: The ‘believe new news’ strategy uses the fuzzy belief *revision* operation introduced in Section 6.1. As explained above, revision can remove information from a knowledge base in case of conflicts to restore consistency. In this case, a fuzzy revision of two formulas depends on their order, as the semantics of fuzzy belief revision is that the new information must be added. In case of conflicts, when there is more than one possibility to restore consistency, the result also depends on the preference order of the clauses in the formula as mentioned previously. In strategy (3), we order the statements chronologically,

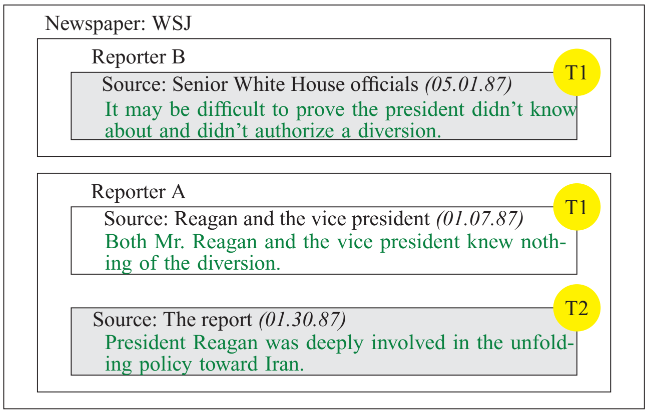


Fig. 5. (Colour online) System's beliefs based on majority strategy.

depending on the timestamp of the news source. The fuzzy belief revision process compares statement sets, formally represented by the fuzzy formulas in conjunctive normal form, with each other. If the two statements sets are compatible, the revision process results in a new set containing the fuzzy union of both sets. However, in case some of the statements differ to a degree exceeding the prescribed minimal consistency γ, the revision operator will remove individual, inconsistent statements from the first set, according to the preference ordering (Witte 2002). In the example in Figure 4 at step 6, we can see the formula generated in previous steps containing two clauses, and below it, the new formula, with which we start the revision. The result shown at the bottom in Figure 4 is a new formula containing two clauses. Here the ordering of the clauses, which determines the sequence of processing, is defined by the date of the statements, which models a reader giving up older beliefs in favor of newer ones; however, as mentioned above, we can easily model different revision strategies by supplying a different preference ordering.

Figure 5 shows an example output generated by our system for newspaper articles. The sentences in the inner boxes show the extracted reported speech statements that are grouped (nested) according to their source, the reporter, and the newspaper. The circles in the top right of each box show the *id* of the corresponding *topic*. Here the statements with gray background are believed by the system using the *majority* strategy: The Fuzzy Believer rejected the belief offered by one article stating that 'Both Mr. Reagan and the vice president knew nothing of the diversion,' and instead opted to believe that 'President Reagan was deeply involved in the unfolding policy toward Iran.'

7.1 Evaluating the polarity-detection component

Detecting opposing statements requires polarity detection. We evaluate this aspect of our system on the entailment pairs of the PASCAL challenge corpus (Bar-Haim *et al.* 2006). We tested different configurations and computed accuracy for two settings. For one experiment, we included all results in the evaluation counting the entailment pairs that were not grouped into the same topic by the topic classification

Table 7. *Polarity identification: accuracy values for different parse methods*

Configuration					Polarity & Topic		Polarity only	
1	2	3	4	5	RASP	MiniPar	RASP	MiniPar
3	3	3	strict	0.7	0.52	0.55	0.53	0.58
5	5	5	lenient	0.7	0.51	0.53	0.51	0.53
5	5	5	strict	0.3	0.52	0.53	0.55	0.51
5	5	5	strict	0.7	0.51	0.54	0.50	0.56
7	7	7	strict	0.7	0.51	0.52	0.51	0.52

as non-entailing ('Polarity & Topic' in Table 7). Here the best results were around 55 percent accuracy. The other test setting only considered the sentence pairs that were actually grouped into the same topic by the topic classification component ('Polarity only') yielding an accuracy of 58 percent using the MiniPar-extracted PAS.

Table 7 gives an overview of the obtained results with different configuration settings: From left to right: Maximum WordNet distance between (1) subjects, (2) verbs, (3) objects of two statements, (4) indicates whether a new statement has to match with one (lenient) or all (strict) statements within one topic, and (5) is the threshold for assigning the same polarity to a statement. When judging the topic identification and the polarity determination, MiniPar gives better results than RASP. Judging only the polarity, there is one configuration where RASP outperforms MiniPar.

8 Discussion

The extraction of opinions from newspaper articles (Pang and Lee 2008; Balahur *et al.* 2009; Somasundaran *et al.* 2009) or customer reviews (Gamon *et al.* 2005; Kim and Hovy 2006) has become an active research field. These approaches are usually only concerned with the identification and extraction of information without processing it further, except for binary classification within a clearly specified domain.

In the wake of the PASCAL challenge (Dagan *et al.* 2005; Bar-Haim *et al.* 2006), systems have been developed to deal with the relation of sentences to each other. Different approaches focus on the recognition of false entailment (Snow, Vanderwende and Menezes 2006) or on learning entailment (MacCartney *et al.* 2006). Others are concerned with relatedness between words and how to measure it (Klebanov 2006). We were not interested in concentrating on one of these areas but rather to develop an all-encompassing system, incorporating many different aspects.

The automatic detection and extraction of quotations was done, for example, by Pouliquen *et al.* (2007) in a multilingual setting. Related work, such as Ruppenhofer, Sporleder and Shirokov (2010), deals with the sub-problem of speaker attribution. Our system achieved highly competitive results for extracting reported speech. Doandes (2003) used a different subset of the WSJ corpus and reports a recall of 0.44 and a precision of 0.92, compared with 0.83 and 0.98 obtained by our system.

For the topic classification, our best results for 300 paraphrase pairs from the MSR Paraphrase Corpus are, for recall 81 percent (with a precision of 38 percent), and for precision 52 percent (with a recall of 58 percent). These values can probably be improved by using more sophisticated heuristics, although there will be a ceiling set by the parser and by the use of language in general. The same meaning can be expressed by different sentences whose words are not in close relations to each other, and therefore hard to detect by current NLP tools. Keeping these facts in mind, the obtained numbers are rather satisfactory and promising for future development.

The rather shallow semantic approach sets a practical limit to the achievable results. This can be inferred from the discrepancy of performance using manually annotated structures compared with using automatic parsers. Combining the results of different parsers could lead to better results, but a precision of 55 percent and a recall of 85 percent, as obtained for the best configuration of the system using manually parsed PAS, shows that our PAS extraction requires better heuristics for a significant improvement.

The polarity identification task was expectedly the hardest one. This is illustrated by the rather poor results that we obtained by trying to find different opinions within one topic. Best accuracy values were obtained using MiniPar and were around 58 percent. This task is very hard for computational systems. But with more elaborated heuristics it is possible to increase these numbers, comparable to the Pascal challenge (Dagan *et al.* 2005; Bar-Haim *et al.* 2006), where systems also started with around 50 percent accuracy and improved over time.

Evaluating the overall system is difficult, because it is a novel task for which no comprehensive data sets are available. We chose to evaluate components individually where possible, as reported above. To assess the overall usefulness of our complex system, however, we had to switch to a proxy task that shares enough characteristics to be informative. Textual entailment (Dagan *et al.* 2005), the task of inferring whether two texts produce inferences which overlap significantly, can be construed as a belief expansion test case. While the analogy between the tasks is strained, adequate performance on textual entailment tasks is a reasonable assessment of the interaction of different components previously evaluated individually only and thus gives a good indication of the penalty of the added complexity. An accuracy of up to 54 percent for the 2008 main task on recognizing textual entailment (Krestel *et al.* 2008a) and 56 percent in 2009 (Krestel *et al.* 2009) indicate the viability of our approach with our system being a little below average (Giampiccolo *et al.* 2008; Bentivogli *et al.* 2009). Still, not all entailments can be found relying only on the output of parsers and PAS. The inclusion of noun phrases to generate PAS showed promising results for the topic identification and allowed to capture some previously undetected entailment relations.

9 Conclusions

Recent efforts in computational linguistics have increasingly focused on shallow semantic tasks, such as extractive summarization and textual entailment. Tools have increased in performance, and we have dependency parsers of good accuracy and general lexical semantics encoded in WordNet. In this context, we have explored a complex system that builds on several of these current base techniques to model an artificial newsreader that can be personalized with respect to previously held beliefs and belief strategies. We combined dependency parser output and WordNet similarities with heuristics to process reported speech sentences into different topics (challenging because of the lack of redundancy). Within the topic, the basic statements are analyzed according to their source and whether they support or oppose each other. This representation was mapped into the fuzzy set theory. We implemented a few simple belief strategies (e.g. believe everything, believe new news, believe majority) to determine the variation this achieves.

The combined system cannot be meaningfully evaluated, thus we compared components on current shared task data. In general, our components are competitive, but not at the top of the state of the art. Interestingly, the combined and complex prototype system does not suffer inordinately for its complexity and thus serves as a viable proof of concept.

Our Fuzzy Believer system can be applied in different scenarios: (1) Companies evaluating product reviews on web sites or blogs, (2) governmental organizations interested in dispositions of people, or (3), as we demonstrated here, individuals requiring news analysis. While the overall goal of an artificial newspaper reader is rather ambitious, individual analysis steps performed by the system, such as predicate–argument extraction (Krestel *et al.* 2010) or reported speech analysis (Krestel *et al.* 2008b), have become essential building blocks for other NLP tasks, and our open source components[8] have been widely adopted by the community.

Apart from the evaluations described in this paper, tests of the system on actual newspaper articles showed accepted and rejected beliefs that reflect the desired results. Embedding the system within an Internet agent and measuring its effectiveness for a real user will be the next major step.

References

Balahur, A., Steinberger, R., der Goot, E. V., Pouliquen, B., and Kabadjov, M. A. 2009. Opinion mining on newspaper quotations. In *Proceedings of the Web Intelligence/IAT Workshops'2009*, pp. 523–6. Milan, Italy.

Ballim, A., and Wilks, Y. 1991. *Artificial Believers: The Ascription of Belief*. Mahwah, NJ, USA: Lawrence Erlbaum Associates.

Ballim, A., Wilks, Y., and Barnden, J. A. 1991. Belief ascription, metaphor, and intensional identification. *Cognitive Science* **15**: 133–71.

Bar-Haim, R., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B., and Szpektor, I. 2006. The second PASCAL recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, pp. 1–9. Venice, Italy.

Bentivogli, L., Dagan, I., Dang, H. T., Giampiccolo, D., and Magnini, B. 2009. The fifth PASCAL recognizing textual entailment challenge. In *Proceedings of TAC'2009*, pp. 1–15. Gaithersburg, Maryland, USA.

[8] Reported Speech Tagger, http://www.semanticsoftware.info/reported-speech-tagger and Predicate–Argument Extractor, http://www.semanticsoftware.info/pax

Bergler, S. 1992. *The Evidential Analysis of Reported Speech.* PhD thesis, Brandeis University, Massachusetts, USA.

Bergler, S. 1995a. From lexical semantics to text analysis. In P. Saint-Dizier and E. Viegas (eds.), *Computational Lexical Semantics*, pp. 98–124. Cambridge, UK: Cambridge University Press.

Bergler, S. 1995b. Generative lexicon principles for machine translation: a case for meta-lexical structure. *Journal of Machine Translation* **9(3)**: 4–155.

Bergler, S. 2005. Conveying attitude with reported speech. In J. C. Shanahan, Y. Qu, and J. Wiebe (eds.), *Computing Attitude and Affect in Text: Theory and Applications*, pp. 11–23. New York, USA: Springer-Verlag.

Bergler, S, Doandes, M, Gerard, C., and Witte, R. 2004. Attributions. In Y. Qu, J. Shanahan, and J. Wiebe (eds.), *Exploring Attitude and Affect in Text: Theories and Applications*, pp. 16–19. Technical Report SS-04-07. Stanford, CA, USA: AAAI Press.

Briscoe, T., Carroll, J., and Watson, R. 2006. The second release of the RASP system. In *Proceedings of the COLING/ACL on Interactive Presentation Sessions.* (COLING-ACL '06), pp. 77–80. Stroudsburg, PA, USA: Association for Computational Linguistics.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M. A., Saggion, H., Petrak, J., Li, Y., and Peters, W. 2011. *Text Processing with GATE (Version 6).* South Yorkshire, UK: Department of Computer Science, University of Sheffield.

Cunningham, H., Maynard, D., and Tablan, V. 2000 (November). *JAPE: A Java Annotation Patterns Engine*, 2nd ed. Research Memorandum CS–00–10. South Yorkshire, UK: Department of Computer Science, University of Sheffield.

Dagan, I., Glickman, O., and Magnini, B. 2005. The PASCAL recognising textual entailment challenge. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment.*

Dang, H. T., and Owczarzak, K. 2008. Overview of the TAC 2008 update summarization task. In *Proceedings of the Text Analysis Conference 2008.*

Das, A. S., Datar, M., Garg, A., and Rajaram, S. 2007. Google news personalization: scalable online collaborative filtering. *Proceedings of the 16th International Conference on World Wide Web* (WWW '07), pp. 271–80. New York, NY, USA: ACM.

Diab, M. T., Levin, L., Mitamura, T., Rambow, O., Prabhakaran, V., and Guo, W. 2009. Committed belief annotation and tagging. In *Proceedings of the Third Linguistic Annotation Workshop at ACL-IJCNLP09*, Singapore, pp. 68–73. Stroudsburg, PA, USA: Association for Computational Linguistics.

Doandes, M. 2003. *Profiling For Belief Acquisition From Reported Speech.* Master's thesis, Concordia University, Montreal, Quebec, Canada.

Dolan, B., Brockett, C., and Quirk, C. 2005 (March). *Mirosoft Research Paraphrase Corpus.* http://research.microsoft.com/en-us/downloads/607d14d9-20cd-47e3-85bc-a2f65cd28042/ (Accessed 3 Oct 2012).

Doms, A., and Schroeder, M. 2005. GoPubMed: exploring PubMed with the gene ontology. *Nucleic Acids Research* **33**(suppl 2): W783–6.

Evans, D. K., Klavans, J. L., and McKeown, K. R. 2004. Columbia Newsblaster: multilingual news summarization on the web. In *Demonstration Papers at the Proceedings of HLT-NAACL 2004*(HLT-NAACL–Demonstrations '04), pp. 1–4. Stroudsburg, PA, USA: Association for Computational Linguistics.

Fellbaum, C. (ed). 1998. *WordNet: An Electronic Lexical Database.* Cambridge, MA: MIT Press.

Gaizauskas, R., Hepple, M., Saggion, H., Greenwood, M. A., and Humphreys, K. 2005. SUPPLE: a practical parser for natural language engineering applications. In *Proceedings of the Ninth International Workshop on Parsing Technology* (Parsing '05), pp. 200–1. Stroudsburg, PA, USA: Association for Computational Linguistics.

Gamon, M., Aue, A., Corston-Oliver, S., and Ringger, E. K. 2005. Pulse: mining customer opinions from free text. In A. F. Famili, J. N. Kok, J. M. Peña, A. Siebes, and A. J. Feelders (eds.), *Advances in Intelligent Data Analysis VI, Proceedings of the 6th International Symposium on Intelligent Data Analysis, IDA 2005*, Madrid, Spain, September 8–10, pp. 121–32, LNCS vol. 3646. New York, USA: Springer.

Gerard, C. 2000. *Modelling Readers of News Articles Using Nested Beliefs*. Master's thesis, Concordia University, Montreal, Quebec, Canada.

Giampiccolo, D., Dang, H. T., Magnini, B., Dagan, I., Cabrio, E., and Dolan, B. 2008. The fourth PASCAL recognizing textual entailment challenge. In *Proceedings of TAC'2008*.

Jurafsky, D., and Martin, J. H. 2008. *Speech and Language Processing*, 2nd ed. Upper Saddle River, New Jersey, USA: Pearson-Prentice Hall.

Kim, S.-M., and Hovy, E. 2006. Identifying and analyzing judgment opinions. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pp. 200–7. New York City, NY, USA: Association for Computational Linguistics.

Kim, J. D., Pyysalo, S., Ohta, T., Bossy, R., Nguyen, N., and Tsujii, J. 2011a. Overview of BioNLP shared task 2011. In *Proceedings of BioNLP Shared Task 2011, a Workshop at ACL HLT 2011*.

Kim, J.-D., Pyysalo, S., Ohta, T., Bossy, R., Nguyen, N., and Tsujii, J. 2011b. Overview of BioNLP shared task 2011. In *Proceedings of BioNLP Shared Task 2011 Workshop*.

Klebanov, B. B. 2006. Measuring semantic relatedness using people and WordNet. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pp. 13–16. New York City, USA: Association for Computational Linguistics.

Klein, D., and Manning, C. D. 2003a. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics* (ACL '03), pp. 423–30. Morristown, NJ, USA: Association for Computational Linguistics.

Klein, D., and Manning, C. D. 2003b. Fast exact inference with a factored model for natural language parsing. In *Advances in Neural Information Processing Systems*, vol. 15. Cambridge, MA: MIT Press.

Krestel, R., Bergler, S., and Witte, R. 2008a. A belief revision approach to textual entailment recognition. In *Proceedings of Text Analysis Conference (TAC 2008)*. Gaithersburg, MD, USA: National Institute of Standards and Technology (NIST).

Krestel, R., Bergler, S., and Witte, R. 2008b. Minding the source: automatic tagging of reported speech in newspaper articles. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC 2008)*, pp. 2823–28. Paris, France: European Language Resources Association (ELRA).

Krestel, R., Bergler, S., and Witte, R. 2009. Believe it or not: solving the TAC 2009 textual entailment tasks through an Artificial Believer System. In *Proceedings of Text Analysis Conference (TAC 2009)*. Gaithersburg, MD, USA: National Institute of Standards and Technology (NIST).

Krestel, R., Witte, R., and Bergler, S. 2007a. Creating a fuzzy believer to model human newspaper readers. In Z. Kobti and D. Wu (eds), *Proceedings of the 20th Canadian Conference on Artificial Intelligence (Canadian A.I. 2007)*, pp. 489–501, LNAI vol. 4509. Montréal, Québec, Canada: Springer.

Krestel, R., Witte, R., and Bergler, S. 2007b. Processing of beliefs extracted from reported speech in newspaper articles. In *Proceedings of Recent Advances in Natural Language Processing* (RANLP-2007), September 27–29.

Krestel, R., Witte, R., and Bergler, S. 2010. Predicate-Argument EXtractor (PAX). In *Proceedings of the First Workshop on New Challenges for NLP Frameworks Co-located with LREC 2010*, pp. 51–4. Paris, France: European Language Resources Association (ELRA).

Lin, D. 1998. Dependency based evaluation of MINIPAR. In *Proceedings of the Workshop on the Evaluation of Parsing Systems, First International Conference on Language Resources and Evaluation*.

MacCartney, B., Grenager, T., de Marneffe, M.-C., Cer, D., and Manning, C. D. 2006. Learning to recognize features of valid textual entailments. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pp. 41–8. New York City, NY, USA: Association for Computational Linguistics.

McKeown, K. R., Barzilay, R., Evans, D., Hatzivassiloglou, V., Klavans, J. L., Nenkova, A., Sable, C., Schiffman, B., and Sigelman, S. 2002. Tracking and summarizing news on a daily basis with Columbia's Newsblaster. In *Proceedings of the Second International Conference on Human Language Technology Research* (HLT '02), pp. 280–5. San Francisco, CA, USA: Morgan Kaufmann.

McKeown, K., Passonneau, R. J., Elson, D. K., Nenkova, A., and Hirschberg, J. 2005. Do summaries help? In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR '05), pp. 210–17. New York, NY, USA: ACM.

Merlo, P., and Ferrer, E. E. 2006. The notion of argument in prepositional phrase attachment. *Computational Linguistics* **32**(3): 341–78.

Nelson, S. J., Johnston, W. D., and Humphreys, B. L. 2001. Relationships in medical subject headings. In Carol A. Bean, and Rebecca Green (eds.), *Relationships in the Organization of Knowledge*, pp. 171–84. New York: Kluwer.

Pang, B., and Lee, L. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval* **2**(12). doi:10.1561/1500000011.

Pouliquen, B., Steinberger, R., and Best, C. 2007. Automatic detection of quotations in multilingual news. In *Proceedings of Recent Advances in Natural Language Processing* (RANLP-2007), September 27–29.

Prabhakaran, V., Rambow, O., and Diab, M. T. 2010. Automatic committed belief tagging. In C.-R. Huang and D. Jurafsky (eds.), *23rd International Conference on Computational Linguistics, Posters Volume* (COLING 2010), August 23–27, Beijing, China, pp. 1014–22. Beijing, China: Chinese Information Processing Society of China.

Quirk, R. 1985. *A Comprehensive Grammar of the English Language*. London, UK: Longman.

Ruppenhofer, J., Sporleder, C., and Shirokov, F. 2010. Speaker attribution in cabinet protocols. In *Proceedings of the International Conference on Language Resources and Evaluation* (LREC 2010), May 17–23, Valletta, Malta. Paris, France: European Language Resources Association.

Snow, R., Vanderwende, L., and Menezes, A. 2006. Effectively using syntax for recognizing false entailment. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pp. 33–40. New York City, USA: Association for Computational Linguistics.

Somasundaran, S., Namata, G., Getoor, L., and Wiebe, J. 2009. Opinion graphs for polarity and discourse classification. In *Proceedings of TextGraphs-4: Graph-based Methods for Natural Language Processing*, Singapore, August 7, 2009.

Witte, R. 2002. Fuzzy belief revision. In *Proceedings of the 9th International Workshop on Non-Monotonic Reasoning (NMR'02)*, April 19–21, pp. 311–20.

Witte, R., and Bergler, S. 2003. Fuzzy coreference resolution for summarization. In *Proceedings of 2003 International Symposium on Reference Resolution and Its Applications to Question Answering and Summarization (ARQAS)*, June 23–24, pp. 43–50. Venice, Italy: Università Ca' Foscari.

Zadeh, L. A. 1965. Fuzzy sets. *Information and Control* **8**(3): 338–53.