Available online at www.sciencedirect.com

ScienceDirect

Computer Speech and Language 35 (2016) 134–160

COMPUTER SPEECH AND LANGUAGE

www.elsevier.com/locate/csl

Coherent narrative summarization with a cognitive model

Renxian Zhang [a,*], Wenjie Li [b], Naishi Liu [c], Dehong Gao [d]

[a] *Department of Computer Science and Technology, Tongji University, China*
[b] *Department of Computing, The Hong Kong Polytechnic University, Hong Kong*
[c] *Department of English, School of Foreign Languages, Shanghai Jiaotong University, China*
[d] *1688 Search and Recommendation, Alibaba.Inc., China*

Received 21 April 2014; received in revised form 27 March 2015; accepted 8 July 2015
Available online 17 July 2015

Abstract

For summary readers, coherence is no less important than informativeness and is ultimately measured in human terms. Taking a human cognitive perspective, this paper is aimed to generate coherent summaries of narrative text by developing a cognitive model. To model coherence with a cognitive background, we simulate the long-term human memory by building a semantic network from a large corpus like Wiki and design algorithms to account for the information flow among different compartments of human memory. Proposition is the basic processing unit for the model. After processing a whole narrative in a cyclic way, our model supplies information to be used for extractive summarization on the proposition level. Experimental results on two kinds of narrative text, newswire articles and fairy tales, show the superiority of our proposed model to several representative and popular methods.
© 2015 Elsevier Ltd. All rights reserved.

Keywords: Cognitive modeling; Summarization; Coherence; Proposition extraction

1. Introduction

This paper is devoted to a special task in automatic text summarization: generating coherent as well as informative summaries for narrative text. Ever since Luhn (1958), summarization researchers have made great efforts to increase the information coverage, or **informativeness**, of a summary. But equally important is a summary's **coherence**, which is our current emphasis.

The concern with coherence is motivated by the ultimate purpose of automatic text summarization – to provide human readers, not machines, with a sufficiently abridged summary of a long document or document set to facilitate efficient information processing. In this sense, the summary serves as a surrogate for the original document(s) in terms of informativeness and expressiveness. Informatively, the summary is expected to maximally reproduce the original document's essential information in a reduced space. Expressively, it is expected to convey the information in an intelligible and coherent way to human readers.

[*] Corresponding author. Tel.: +86 13816793063.
E-mail addresses: rxzhang@tongji.edu.cn (R. Zhang), cswjli@comp.polyu.edu.hk (W. Li), rong@sjtu.edu.cn (N. Liu), dehong.gdh@alibaba-inc.com (D. Gao).

http://dx.doi.org/10.1016/j.csl.2015.07.004
0885-2308/© 2015 Elsevier Ltd. All rights reserved.

Many coherence-oriented or coherence-based approaches to summarization concentrate on textual content, such as word cohesion (Halliday and Hasan, 1976; Barzilay and Elhadad, 1997), sentence similarity (Hatzivassilogiou et al., 2001; Zhang, 2011), rhetorical structure (Marcu, 2000), etc. But since the ultimate consumers and judges of a summary are human readers, there is no reason why we cannot model coherence in *human terms*. But such attempts are surprisingly rare in the summarization community. To account for such human terms, we can resort to the theories and models developed by cognitive psychologists over decades.

We choose to summarize narrative text because compared with expository or argumentative text, a narrative text relies more on coherence for successful human understanding. When reading a typical expository article such as a biography, we can choose to read only the parts that interest us (e.g., birth place, education, marriage) and the lack of coherence between the chosen parts does not affect our understanding of the person. When reading a typical argumentative article such as a scientific thesis, we can focus on only particular sections to get the *method*, *result*, *conclusion*, etc. to understand the topic despite the lack of global coherence. What about reading a typical narrative article such as a story? Reading only parts of the story disrupts the development of plot and renders an incoherent representation of the characters, their relations, and events in our mind, which prevents us from understanding it. The situation is true for both the original text and the summary.

In this work, we will build a novel computational model based on a popular cognitive model (Kintsch, 1998) of narrative text comprehension, establishing its computational counterparts in the model's cognitive process. Coherence is an underlying constituent of the model, which is then used to summarize narrative text. Moreover, summary sentences extracted with this model are not only coherent but also important, a point that will be validated by experiments on event-centric news and fairy tales, both typical instances of narrative text. This is our major contribution to the summarization community.

We will discuss related work in the literature in Section 2. In Section 3, we will computerize a cognitive model of narrative text comprehension with all the technical details. In Section 4, the cognitive model-driven coherence will be used to summarize narrative text, where propositions instead of sentences will be taken as the basic processing units. Section 5 presents the experimental results on two kinds of narrative text. The highlights of our work are concluded in Section 6, where we also point out future directions.

2. Related work

Our work is informed by several sources of related work. The modeling of coherence has its root in cognitive accounts of text comprehension; the concern with coherence is generally preceded by many exemplar works; narrative summarization is not a new topic in the summarization community. We will briefly introduce works from those sources that jointly shape up the current endeavor.

2.1. Cognitive accounts of text comprehension and coherence

In cognitive psychology, a large body of research focuses on text comprehension, as many researchers relate the linguistic aspects and processes involved in reading to activities in the human memory. Coherence is, for cognitive psychologists, concomitant with text comprehension which is intensively studied to understand human cognition. According to many theories and models of cognitive psychology (Tapiero, 2000; van Dijk and Kintsch, 1983; Gernsbacher, 1996; Kintsch, 1988, 1998; van den Broek et al., 1996; Zwaan et al., 1995; Tapiero, 2007), a coherent representation is required for text comprehension. In order to make sense of a text, readers must establish coherent relations between textual units. Therefore, coherence and text comprehension are the two sides of the same coin. Guided by Centering Theory-based coherence, Cristea and Iftene (2010) empirically show that human cognition is near optimal and economical (stack-like).

To capture coherence in this flavor, many models have been developed, such as the Construction-Integration (CI) Model (Kintsch, 1998), the Structure Building Framework (Gernsbacher, 1990), the Landscape Model (van den Broek et al., 1996), the Event-Indexing Situation Model (Zwaan et al., 1995), and the Intentional Partial Order Causal Link Planning Model (Riedl and Young, 2010). The Landscape model, for example, captures the changing patterns of word activation guided by anaphoric clarity and clausal coherence. The CI model accounts for how propositions from input text are associated in a network with stored knowledge from the long-term memory. Its extended version, CI-2 (Kintsch and Mangalath, 2011), employs a dual-memory model that highlights the role of the explicit context of words.

Lemaire et al. (2006) describe an implementation of the well-known CI model and a comprehension model based on the information flow in human memory, a general structure that our model will adopt.

2.2. Coherence and cognitive modeling in summarization

Many researchers in text summarization have been trying to model coherence on two levels: global and local. Hybrid models that integrate both levels have also emerged in recent years.

Models of global coherence are often based on Rhetorical Structure Theory (RST) to capture the discourse-level coherence patterns. The extensive use of RST to text summarization is usually credited to Marcu (1997, 1999, 2000), who shows that guided by rhetorical relations between clauses, it is possible to parse a discourse. Wolf and Gibson (2004, 2006), however, find fault with the binary tree in RST, and advocate a "chain graph structure" that can represent crossed dependencies and multiple-parent nodes. Knott et al. (2001) argue against the (object-attribute) elaboration in RST and propose supplementing RST with entity-based coherence, a kind of local coherence. Using a content model based on HMM, Barzilay and Lee (2004) interpret global coherence as a domain-specific topical structure. According to their content model, each HMM state corresponds to a topic from which sentences are generated. In effect, the content model captures coherence pattern as shift between topic states. More recently, discourse-level coherence is also integrated into a graph model (Christensen et al., 2013) that accounts simultaneously for coherence, salience, and redundancy.

Local coherence models are mostly based on the Centering Theory (CT; Grosz et al., 1995) or the linguistic account of lexical cohesion to capture the relationship between adjacent textual units (usually sentences). As a direct application, CT's constraints and rules (Brennan et al., 1987) can be used to generate metrics for local coherence. Hasler (2004) directly applies the CT's transitions (Continue, Retain, Smooth Shift, Rough Shift) to text summarization. Orăsan (2003) develops a CT-based local coherence algorithm for sentence extraction by using evolutionary programming. Sentences are ranked and selected on the basis of content and context. The idea of entity coherence, which is related to CT transitions, gives rise to a wave of new research interests. Barzilay and Lapata (2005, 2008) propose an entity grid model to capture local coherence. In CLASSY, Conroy et al. (2006) rely on lexical overlap to order sentences that achieves local coherence, which instantiates a Traveling Salesman Problem (TSP)-style search method. For the same purpose, Lapata (2003) considers both lexical and syntactic features to calculate local coherence between neighboring sentences using a greedy algorithm.

An attempt to integrate lexical cohesion into a global coherence model is made by Alonso i Alemany and Fuentes (2003). They build a hybrid model of text summarization that combines rhetorical relations to account for coherence and lexical chains to account for cohesion. Soricut and Marcu (2006) develop "utility-trained coherence models" based on HMM. Their model integrates both local models (word-co-occurrence coherence and entity-based coherence) and global models (HMM-based content models), in a log-linear fashion. Similarly, Elsner et al. (2007) report on a method of coherence-based text generation that combines a local coherence model (Barzilay and Lapata, 2005) and a global coherence model (Barzilay and Lee, 2004). Cristea et al. (1998) establish the Veins Theory (VT) that combines both a global coherence-based model (like RST) and a local one (like CT). Kibble and Power (2004) present a CT-guided RST model based on the propositional representations and the established RST rhetorical structure of the text.

More recently, cognitive modeling has been applied to summarization. Pastor (2011, 2012) present the COMPENDIUM summarizer, which is based on van Dijk and Kintsch's (1983) theory about the process of text comprehension, which proposes the macrostructure as a result of macrorules. The author argues that the macrostructure is conceptually equivalent to summary and the macrorules can be converted to computational steps of summarization. It is noteworthy that van Dijk and Kintsch's (1983) theory does not address the psychological mechanism underlying macrostructures and macrorules, and consequently COMPENDIUM does not directly implement any cognitive constructs or processes.

Fang and Teufel (2014) propose a summarization approach that implements some of the key elements in the work of Kintsch and van Dijk (1978). Their most important contributions are automatic proposition extraction based on dependency parsing, concept matching based on lexical semantic computation, and proposition root shift. Their method also shares some technical details with ours, such as dependency parsing-based proposition extraction. In the experiments, we will compare their results with ours.

But Kintsch and van Dijk (1978) and van Dijk and Kintsch (1983) are early works in this line of research, which emphasizes concept matching and proposition attachment. Ideas like macrostructure and proposition attachment are later evolved into Kintsch's (1998, 2001) CI model, which relates concept, proposition, macrostructure, etc. to human memory operations and integrate everything into a more cogent cognitive account. It is this account that our summarization model builds on.

2.3. *Narrative summarization*

In the text summarization community, the usual target is the newswire article. Narrative or story summarization is rarely reported in early days (Lehnert, 1999) but sees a burgeoning growth in recent years (Kazantseva, 2006; Mihalcea and Ceylan, 2007; Kazantseva and Szpakowicz, 2010). Kazantseva and her colleague's work on short stories (Kazantseva, 2006; Kazantseva and Szpakowicz, 2010) focuses on finding summary-worthy sentences using rule-based and machine learning approaches. For that purpose, they extract features about character, aspect, and location. The most novel part is "aspect", which is a linguistic characteristic of a clause that gives an idea about the temporal flow of an event or state being described. Mihalcea and Ceylan (2007) apply well-known single-document summarization techniques to book summarization. Starting from the initial summarizer based on MEAD (Radev et al., 2004), the authors make steady performance gains by accommodating the document length: exclusion of positional scores, text segmentation and segment ranking, and a segment-based weighting scheme.

A distinct feature of narrative text is the plot organization. Plot-based narrative summarization is championed by Lehnert (1981), who focus on the affect states of characters. The summarization generation is based on the identification of plot units, which are composed of affect states and causal links (motivation, actualization, termination, equivalence). The narrative structure is in turn represented as a graph that reflects the relations between plot units. More recently, Goyal et al. (2010) discuss new corpus-based techniques to automatically extract plot units. Such methods usually require considerable knowledge engineering or external resources. None of them, however, makes use of the cognitive modeling of narrative comprehension.

3. Cognitive model of narrative comprehension and coherence

Our comprehension of a narrative changes constantly at different stages of reading. A new event or the appearance of a new character will alter our mental representation of the whole story as we seek to establish **meaningful or coherent links** between the new elements and the old ones. That is why many cognitive psychologists regard the process of text comprehension as guided by the mechanism for establishing and preserving coherence, such as those described in Section 2.1.

Those models are similar in that they model coherence establishment for a narrative as a dynamic process. A new textual unit (e.g., word) activates related information in the long term memory, and then a cognitive mechanism selects information that is most relevant to the current mental representation of the narrative. The textual units are linearly processed so that the mental representation is constantly updated. As narratives are typically about characters, their relations, and happenings around them, the propositional representation with a predicate-argument structure is often adopted by such models to capture such narrative elements that make up a whole plot. Among those models, Kintsch's (1998) CI model is the best developed in both cognitive and computational terms. It lays emphasis on proposition-based word activation from the long term memory (**construction**) and a spreading activation process of strengthening or inhibiting the activated words (**integration**). Our cognitive model to account for narrative text comprehension and coherence is built on those previous works.

3.1. *An overview of the model*

Fig. 1 illustrates the overall architecture of our model. The three blocks – **Long Term Memory**, **Working Memory**, and **Episodic Memory** – are based on the popular theory about human memory composition. The solid-line arrows mark major operations between and within the various text representations in the memory parts, among which **Association** and **Spreading Activation** are derived from the CI model. The dashed-line arrows represent influences from contextual (input text) or general semantic (long term memory) sources.

The model starts with the narrative text, indicated by the icon in the upper-left corner of Fig. 1. The whole text is segmented into sentences, from which propositions are extracted. We read a story sentence by sentence and understand

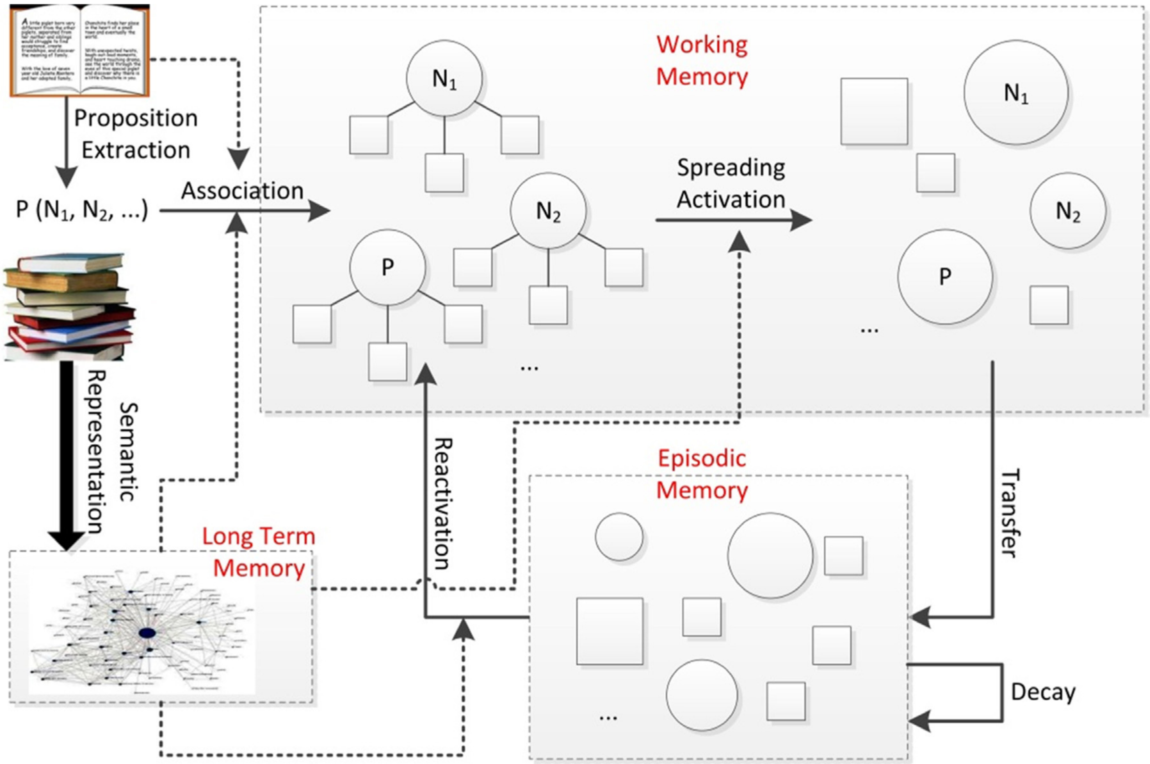


Fig. 1. Architecture of the narrative text comprehension/coherence model.

the plot proposition by proposition, which is modeled as a cyclic process. In each reading cycle, the model receives the current proposition with all its elements as input.

Independent of the narrative text is our general knowledge about relations between words, i.e., a semantic network that results from years of language contact such as reading. Its computational analog is a word vectorial space computed from a corpus. Stored in the long term memory, the semantic network is closely tied with the proposition-based comprehension.

At the beginning of each new reading cycle, the predicate (*P*) and nouns (*N*) in the current proposition is each associated with a number of closest words from the semantic network with cues from the narrative context. Each proposition element and associated word has an activation value and each pair of them has an association value, all computed from the semantic network. In the figure, circles represent words from the narrative text and squares represent associated words. Size indicates activation value.

In the next step, the proposition elements and associated words are reassigned activation values via a spreading activation algorithm. Its cognitive analog is the stabilization of the activation degrees of all related words in the working memory.

Since the working memory has a limited capacity (Just and Carpenter, 1992), the words with their stabilized activation values in the working memory are **transferred** to the episodic memory that stores all the activated words during a narrative's comprehension. It is also a cognitive tendency to attend to only the most recent information, or in a stack-like manner as Cristea and Iftene (2010) show in their experiments. As reading proceeds, a human reader tends to gradually forget narrative elements in earlier sentences/propositions, which is modeled by a **decay** process of all the stored words for each reading cycle.

If sufficiently activated (after decay) and sufficiently close (computed from the semantic network) to the elements in a proposition being processed, a word in the episodic memory will be **reactivated** back into the working memory. After all reading cycles are completed, the episodic memory contains all the narrative element words and their associates with their final activation values. This is also the mental representation of the narrative text according to our model.

Table 1
This is a comparison of different corpora. In our experiments we have used the Wiki, Reuters, and FT corpora.

	Wiki	TASA	Reuters	FT
# documents	3.6M	44k	21,578	453
# words	>2G	11M	3.5M	908k
Degree of Specialization	**Highly generic**	Moderately generic	**Highly specialized**	**Highly specialized**

In the following two sections, we will provide further details of the two main modules of the model: semantic network construction in the long term memory and the proposition-based cyclic comprehension.

3.2. Semantic network in long term memory

A semantic network is supposed to be built on a large corpus and used to decide how semantically close two words are. Kintsch (1998) first applied Latent Semantic Analysis (LSA; Landauer et al., 1998) to a specialized corpus and built a semantic space crucial to his CI model. In this section, we will discuss the use of different kinds of corpus and alternative ways to construct a semantic network, which have not been explored before.

3.2.1. Specialized corpus and Wiki corpus

To endow the computer with language experiences comparable to a human reader, we need to prepare a corpus as input to a semantic model. In the literature, a popular choice is the TASA corpus consisting of educational texts for American school students of different grade levels (Quesada, 2007), which contains over 44 thousand documents and 11 million word tokens.

In the current work, we experiment with two kinds of narrative text – event-centric news and fairy tales – and use two specialized corpora accordingly. The first is the Reuters-21578 benchmark (Reuters) corpus and the second is a freely available 453-story fairy tale (FT) corpus (Lobo and de Matos, 2010). In addition, we use a Wiki corpus from the English Wikipedia articles,[1] which is much larger and more generic than TASA.

The details of the above mentioned corpora are listed in Table 1. By using both a highly generic and two highly specialized corpora, we intend to study the influence of different kinds of corpus on a cognitive model, which has not been reported to the best of our knowledge.

3.2.2. Semantic modeling with standard LSA/LDA

When constructing a semantic network out of a corpus, we will essentially compute word similarities based on word distributional and co-occurrence patterns in documents. LSA and Latent Dirichlet Allocation (LDA) are two appropriate tools for this purpose.

LSA uses a term-by-document matrix A as word co-occurrence evidence and applies Singular Value Decomposition (SVD) to it so that $A \overset{SVD}{=} USV^T$. Then we take the k largest singular values of S to get a lower-rank approximation of A: $U_k S_k V_k^T$, a dense matrix representing a semantic space. For two words i and j in this space, we calculate the cosine similarity of their corresponding vectors:

$$Sim(i, j) = \text{Cosine}(u_{i,*} S_k, u_{j,*} S_k)$$

where $u_{i,*}$ is the ith row vector of U_k.

LDA (Blei et al., 2003) is an alternative model that introduces topic and probability distributions to the observed word co-occurrence pattern. It assumes multinomial distributions for both document-over-topic and topic-over-word distributions with Dirichlet priors. The model parameters can be learned from Bayesian inference such as variational Bayes (Blei et al., 2003) from which we can derive all the posterior topic distributions on word $P(z_n|w)$, $n=1, 2, \ldots, t$. These t probabilities make up a vector for w, based on which we can calculate word similarities as vector cosines.

The standard LSA and LDA described above share a common limitation. Once constructed, the LSA/LDA model is fixed. Updating with new documents would mean starting from scratch. This is computationally thwarting because

[1] We use the 20110317.bz2 dump for our experiment.

fitting millions of documents (for Wiki) in memory all at once is impractical. Instead, we would rather send smaller-sized batches of documents and update the trained model continuously. On the other hand, a "fixed" semantic network does not accord with the fact that the human long-term memory is constantly updated with new information from her cognitive environment.

For both computational and cognitive reasons, we will use the updatable variants of LSA/LDA.

3.2.3. Semantic modeling with updatable LSA/LDA

Distributed LSA (Řehůřek, 2011) is a solution to LSA updating. For the input matrix $A^{m\times n}$ with a large n (number of documents), we partition it into smaller submatrices $[A^{m\times c_1}, A^{m\times c_2}, \ldots, A^{m\times c_k}]$ where $\sum_{i=1}^{k} c_i = n$. Then for any two such submatrices A_1 and A_2, after SVD and k-dimensionality reduction, $A_1 \overset{SVD^k}{=} U_1 S_1 V_1^T = U_1 S_1^2 U_1^T$, $A_2 \overset{SVD^k}{=} U_2 S_2 V_2^T = U_2 S_2^2 U_2^T$. To merge (U_1, S_1) and (U_2, S_2) into (U, S) for $[A_1, A_2]$, we can apply QR decomposition on $[U_1 S_1, U_2 S_2]$ and get an orthonormal matrix Q with the same span of $[U_1, U_2]$. See Řehůřek (2011) for more technical details.

A successful updatable variant of LDA is the online LDA (Hoffman et al., 2010). It is based on batch variational Bayes to fit the parameters $\boldsymbol{\lambda}$ to the variational posterior over the topic distributions with an expectation-maximization (EM) algorithm. In the E-step, the algorithm holds $\boldsymbol{\lambda}$ fixed and fits the per-document variational parameters γ and θ with a new document. In the M-step, $\boldsymbol{\lambda}$ is updated by $\boldsymbol{\lambda}'$, an optimal setting if the whole corpus is a simple repetition of the new document. Hoffman et al. (2010) prove that online LDA converges fast and performs well.

Using distributed LSA and online LDA,[2] we can handle a large corpus like Wiki and build a semantic network with the potential of being updated with new knowledge sources.

3.3. Proposition-based cyclic text comprehension

Motivated by psychological theories of human memory (Anderson, 1976) and cognitive models of text comprehension (Kintsch, 1998), our computational model of story comprehension simulates the human reading process with coherence as an underlying theme. The whole reading process is a cyclic one and in each cycle, a new proposition is processed and the text representation updated in different parts of human memory. In the following, we provide the details of model components before showing a complete algorithm.

3.3.1. Proposition extraction

As mentioned in Section 3.1, propositions are the basic input units in our model, so the first step is to decompose an incoming sentence into propositions. Previous work on similar models (Kintsch, 2001; Lemaire et al., 2006) is equivocal on this issue or uses manually extracted propositions. We will fill the gap so that the model works fully automatically.

Essentially a proposition is represented as $Predicate(Argument_1, Argument_2, \ldots)$ where the predicate is a verb, noun, or adjective and an argument must be a noun. We extract propositions from the dependency tuples after parsing (Klein and Manning, 2003) because they contain information about governing verbs, subjects, objects, and modifiers, from which we can derive propositions.

A difficulty with this approach is that nominal and pronominal anaphora is frequently found in a narrative text. The following example is the first paragraph of the fairy tale *Beauty and the Beast*, where the nouns "merchant", "sons", and "daughters" appear only once and then referred to by 8 pronouns.

(1) *ONCE upon a time, in a very far-off country, there lived a merchant who had been so fortunate in all* ***his*** *undertakings that* ***he*** *was enormously rich.* (2) *As* ***he*** *had, however, six sons and six daughters,* ***he*** *found that* ***his*** *money was not too much to let* ***them*** *all have everything* ***they*** *fancied, as* ***they*** *were accustomed to do.*

If the pronouns are left as is in the dependency tuples, we have no way to tell that it is the same "merchant" who lived somewhere and was rich (sentence anaphora) and had twelve children (discourse anaphora). To extract high-quality

[2] In our experiment, we use the Python modules included in *Gensim*: http://radimrehurek.com/gensim/index.html.

Table 2
Associates of “kill”, using Wiki LSA.

Contextualized	revenge, disguise, terrified
Decontextualized	revenge, dead, steal

propositions, we apply coreference resolution to the dependency tuples first, using the state-of-the-art multi-pass sieve system[3] (Lee et al., 2011) that resolves both pronominal and nominal expressions to a head noun phrase.

Propositions are predicate-dominated and once the predicate is identified, all its attached nouns can be retrieved from a proper dependency tuple. For example, in the above Sentence (1), “lived” is a verb predicate and “merchant” is its attached noun from ***direct_object*** (*lived*, *merchant*).

Apart from extracting propositions based on the simple “Subject–Verb–Object” skeleton or its passive form, we also find predicates and arguments from modifier and complement structures in participles and clauses that characterize complex sentences. This helps to find, in Sentence (1), the proposition *fortunate*(*merchant*) from a clause-level dependency. It is possible that two verbs are found using the modifier or complement dependencies, but only one of them is chosen as the real predicate based on the dependency type. For example, ***complement*** (*accustomed*, *do*) in Sentence (2) gives us two verbs as possible predicates, but only *do* is chosen because it “complements” the meaning of *accustomed* and shifts the focus of the sentence.

The following shows all the propositions extracted from the first paragraph of *Beauty and the Beast*. (3) and (4) list all the propositions in (1) and (2), respectively. The propositions are ordered by the predicate position in the sentence.

(3) *lived(merchant)*; *fortunate(merchant*, *undertakings)*; *rich(merchant)*; (4) *had(merchant*, *sons*, *daughters)*; *found(merchant*, *money*, *sons*, *daughters)*; *let(money*, *sons*, *daughters)*; *have(everything)*; *fancied(sons*, *everything)*; *do(sons)*.

3.3.2. Contextualized word association

With an input proposition, all its words trigger their closest associates from the semantic network stored in the long term memory. However, word association partly depends on the explicit context for disambiguation, so that “bank” is associated with “money” in the context of “lend” but “river” in the context of “water”. In this sense, contextualized word association is related to the common NLP task of word sense disambiguation (Palmer et al., 2001). As we read a word in text, we understand it with reference to both its semantically related words in general and the explicit context it appears in, which is the underlying tenet for “gist-level and verbatim-level information” (Steyvers and Griffiths, 2008) or the “dual-memory model” (Kintsch and Mangalath, 2011).

For those reasons, we calculate the **contextualized association score** of word v with reference to word u and its context $C(u)$, noted as $CAS_u(v)$. We denote the similarity of u and v in the LSA/LDA space as $Sim(u, v)$ and then

$$CAS_u(v) = \frac{1}{|C(u)|+1} \sum_{w \in C(u) \cup \{u\}} Sim(w, v)$$

In our experiment, $C(u)$ consists of the left and right neighbors of u. Table 2 shows the difference between using the contextualized association score calculated by using u’s 3 most frequent neighbors in our experimental dataset and decontextualized association score ($C(u) = \emptyset$) to get the top 3 associates of “kill” according to $CAS_{kill}(v)$. The word similarities are from the Wiki-based LSA space. Obviously, the words “disguise” and “terrified” indicate some special context “kill” is found in.

3.3.3. Spreading activation in working memory

After we get the top n associates (in our experiment, $n = 3$) for each word, all the proposition words and their associates are now resident in the working memory, each with an **activation score *AS*** that denotes its degree of activation. Initially, $AS(w) = 1$ if w is from the proposition. Otherwise it is set to be w’s association score. If w appears more than once, the maximum $AS(w)$ is taken so that all scores fall in [0,1]. But the initial degrees of activation are unstable because of the relations between the words stored in the semantic network. Ultimately, some words may

[3] It is included in Stanford CoreNLP, which also includes the state-of-the-art Stanford Parser that we use for dependency parsing.

stabilize with higher scores because they are closely related to more activated words and some with lower scores because they are related to less activated words, which is supported by the reinforcement of relevant information and deactivation of irrelevant information (Tapiero, 2007:87).

This cognitive process can be modeled by a spreading activation algorithm, first introduced by Kintsch (1998). Let A be a vector of the activation scores of n words: $w_1, \ldots, w_n$: $A = (a_1, ..., a_n)^T$, $a_i = AS(w_i)$ and M be a similarity matrix for the n words: $M = [m_{ij}]_{n\times n}$, $m_{ij} = Sim(w_i, w_j)$. Let $A^{(t)}$ denote A at time t and define

$$A^{(t+1)} = MA^{(t)} / \max\{abs(MA^{(t)})\}$$

A is thus constantly updated by multiplying M and normalizing by the vector component with the largest absolute value: max $\{abs(MA^{(t)})\}$. We now prove its convergence.

Suppose $v_1, \ldots, v_n$ are the n eigenvectors of M, corresponding to the eigenvalues $\lambda_1, \ldots, \lambda_n$ in descending order of their absolute values. According to the definition, $A^{(t)}$ is also bounded ([0,1]). Suppose λ_1 is the single root of the characteristic polynomial, then using the eigenvectors,

$$\begin{aligned} A^{(0)} &= a_1 v_1 + \ldots + a_n v_n \\ A^{(t)} &= M^t A^{(0)}/\varphi_t = M^t(a_1 v_1 + \ldots + a_n v_n)/\varphi_t \\ &= (a_1 \lambda_1^t v_1 + \ldots + a_n \lambda_n^t v_n)/\varphi_t) \overset{t\to\infty}{\approx} a_1 \lambda_1^t v_1/\varphi_t \end{aligned}$$

where φ_t is the normalization coefficient at time t. This shows that φ_{t+1} is actually dependent on the component of v_1 with the largest absolute value. Therefore, $A^{(t)}$ converges to $v_1/\max\{abs(v_1)\}$, where $\max\{abs(v_1)\}$ is the component of v_1 with the largest absolute value.

3.3.4. Activation adjustment in episodic memory

After the current proposition is processed and before the next proposition comes, the activated words are transferred to the episodic memory with their activation scores copied if they did not exist. Otherwise, the activation scores are updated. If the activation score of w in the episodic memory after the nth proposition is $ES^n(w)$ and its activation score (after spreading activation) in the working memory is $AS(w)$, then

$$ES^n(w) = Min(1, ES^{n-1}(w) + AS(w) - ES^{n-1}(w)AS(w))$$

It is easy to see that $ES^n(w)$ is no less than $ES^{n-1}(w)$ or $AS(w)$ (Lemaire et al., 2006) and is still bounded by 1.

On the other hand, according to the **Decay Theory** (Berman, 2009), earlier processed words are gradually forgotten over time. To model this phenomenon, we follow Lemaire et al. (2006) by setting a decay coefficient ($\delta = 0.9$) as a multiplier to $ES^n(w)$ for all w in the episodic memory after proposition n is processed.

Stories typically mention major characters and happenings in different places, and each later mention makes us recall what was earlier said about them. So a word w can be **reactivated** back into the working memory if $ES^{n-1}(w) > \theta_1^{n-1}$ and $Sim(w, u) > \theta_2$ for some u in the nth proposition. Note that instead of taking a fixed value, the activation score threshold θ_1 is dependent on the current state of the episodic memory: $\theta_1^n = \sum_w ES^n(w)/|ES^n(w)|$. θ_2 is independent of the episodic memory and set to be 0.7.

3.3.5. Complete algorithm

To sum up this part, we provide the complete algorithm of the cyclic text comprehension in Fig. 2. *WM* and *EM* are mnemonic notations for sets of words with their activation scores in the working memory (*WM*) and episodic memory (*EM*). Note that when the algorithm terminates, *EM* contains all the activated words from both the text and the long-term memory, with their final activation scores.

The proposition is an important construct in Kintsch's (1998) CI model in that it provides the right words (predicates and arguments) for the model to work with. However, it is not indispensable to our implementation and the algorithm generalizes to any meaning blocks such as sentences, and that is because word association and spreading activation all apply to words. But as our experiments show, the proposition is a good unit of processing and extraction. In addition, they will also play an important role in the next step – summarization.

```
$WM = \{\}$;
$EM = \{\}$;

While exists next proposition $P^i = \{p_1^i, p_2^i, \ldots\}$

    $WM \leftarrow P^i \cup$ {top $n$ associates of $p_1^i, p_2^i, \ldots$ } $\cup$ {words reactivated from $EM$ by
                    $P^i$ };
    Apply spreading activation to $WM$;
    $EM \leftarrow EM$ updated with $WM$;
    Apply decay to $EM$;
```

Fig. 2. Complete algorithm of cyclic comprehension.

4. Coherent narrative summarization

After all propositions in a narrative text have been processed, the episodic memory contains all the activated words with their activation scores. This is the word-level representation of the text according to our cognitive model. Moreover, the highly activated words are relevant to each other because of the spreading activation mechanism. A passage based on such words is expected to be highly coherent. Therefore, a coherent summary of the narrative text can be constructed by focusing on the highly activated words in the episodic memory.

A summary, however, cannot be a mere collection of words. It is expected to be composed of well-formed sentences well connected to each other. A straightforward method is to interpret the highly activated word as the most salient words and select the original sentences containing such words, as most frequency-based extractive summarizers do. In a psychological study, Lemaire et al. (2005) show that selecting sentences based on the word values calculated from the CI model, which our cognitive model is built on, highly correlates with the human selection of sentences to make up a narrative summary.

Although selecting the original sentences may work in our case, it misses an important aspect of our model – propositions. According to Fig. 1, the model receives propositions as input in each reading cycle because proposition is the basic unit of human understanding. After reading the whole text, the propositions receive different degrees of salience in the reader's mind, and a summary should maximally cover all salient and non-redundant propositions. We judge some type of information as being less important and therefore the propositions do not capture each and every piece of information in the text. Doing so reduces the amount of potentially "noisy" propositions, though it is done at a cost of loss of potential and expressive power of the model. The flexible generation of propositions will be left to future work. The proposition-based summarization architecture is illustrated in Fig. 3.

As is shown in the above, the input propositions are first **ranked** according to the activation scores of their constituent words from the episodic memory after the whole text comprehension is completed. But since the summary cannot be composed of propositions like *killed*(*hunter*, *bear*), the propositions need to be **realized** as sentences, or **p-sentences**, which are not necessarily the original sentences from the text. From the ranked p-sentences we select those worthy of being included in a summary. In principle, the **selected** p-sentences need to be both salient (high-ranking) and non-redundant. Finally, the p-sentences are **ordered** to form the output summary.

A major challenge to apply the cognitive model to summarization is sentence realization. Generating sentences directly from our propositions is not feasible because much sentence-building information (verb tense, voice, mood, function words, etc.) cannot be found in the propositions. Our solution is to find sub-sentences corresponding to the propositions from the original text, a strategy to be elaborated in the following.

4.1. Proposition-based sentence extraction

Now we discuss the detailed algorithm of extracting p-sentences from an original sentence. As discussed above, they are the building blocks of the summary. For that purpose, a p-sentence is expected to be informationally compact

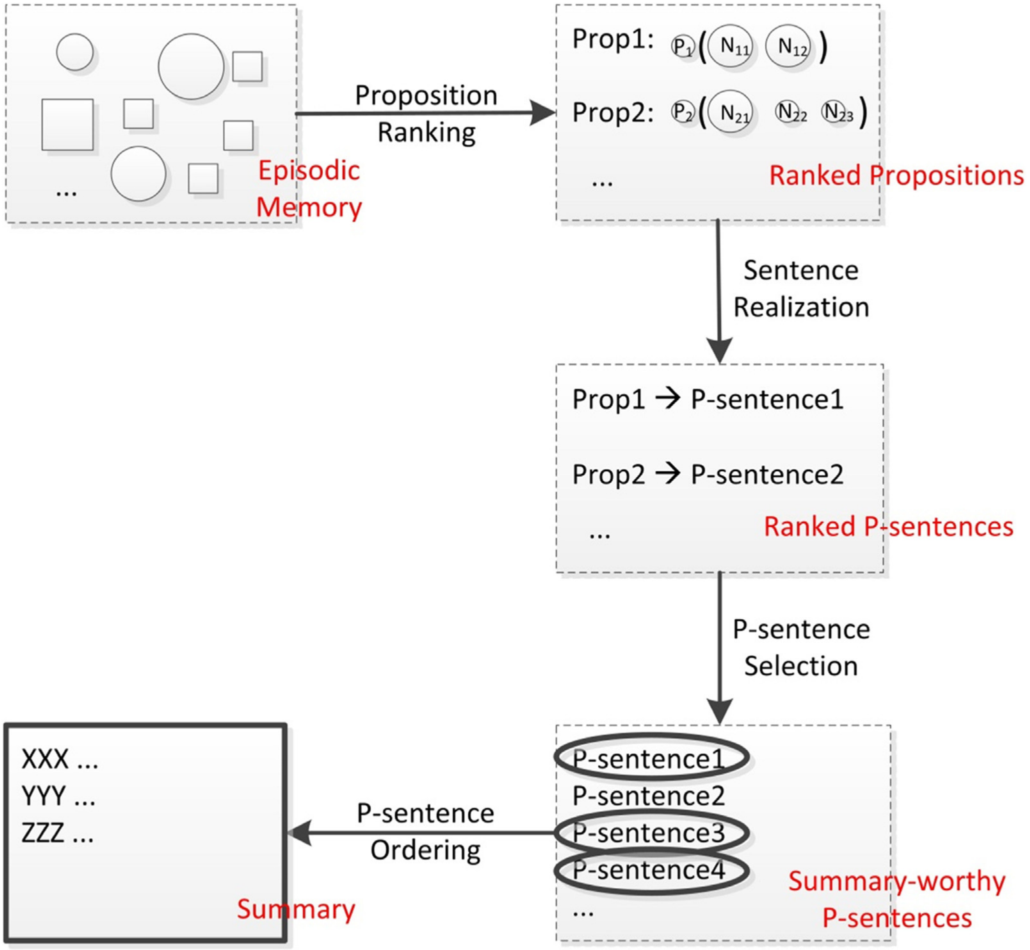


Fig. 3. Architecture of narrative summarization, based on the cognitive model.

(containing as little non-proposition material as possible) and grammatically acceptable. Such agenda can be met by operations on the parsing tree of the original sentence, which contains hierarchical relations between proposition elements as well as syntactical information about how they can be connected in a grammatical way.

Parsing-based methods and tree operations are commonly used in sentence revision (Mani et al., 1999), compression (Cohn and Lapata, 2008; Yousfi-Monod and Prince, 2008; Zajic et al., 2008), reduction (Jing, 2000; Jing and McKeown, 2000), or fusion (Barzilay and McKeown, 2005) to improve the summary quality. Our sub-tree deduction algorithm in the following has borrowed ideas, e.g., tree pruning and adjusting, from those previous works. But to the best of our knowledge, no attempt has been made to deduce sections of a tree to match propositions.

4.1.1. P-sentence extraction as sub-tree deduction

If a sentence contains *n* propositions, we can extract *n* p-sentences. Although the *n* p-sentences are all parts of the original sentence, they are not necessarily non-overlapping. Consider sentence (5) below, which is selected from our experimental dataset, and its automatically extracted propositions (Prop1 to Prop4) in (6)

(5) *THERE was once a young fellow who enlisted as a soldier, conducted himself bravely, and was always the foremost when it rained bullets.*

(6) Prop1: *fellow* (*THERE*)
Prop2: *enlisted* (*fellow*, *soldier*)
Prop3: *foremost* (*fellow*)
Prop4: *rained* (*bullets*)

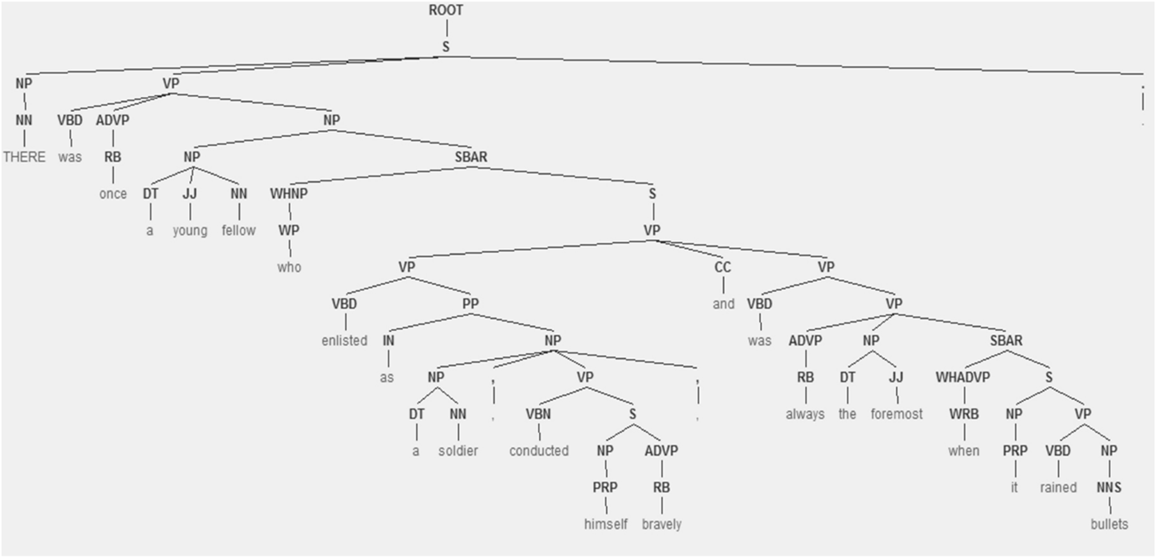


Fig. 4. Parse tree of example sentence (5).

Prop2 and Prop3 both have the word “fellow” as an argument, so their p-sentences must be overlapping. Thus, extracting p-sentences from the original sentence is not decomposing the sentence into non-overlapping parts. Rather, it is formulated as a sub-tree deduction process. Fig. 4 shows the parsing tree of sentence (5), the output of the state-of-the-art Stanford Parser.

Given such a parse tree and a proposition from the sentence, our goal is to deduce a sub-tree that minimally covers the proposition elements and preserves all the syntactically necessary constituents. The following is the top-level algorithm to attain this goal.

In the following, we will discuss the main steps of the algorithm.

- **Find the lowest common parent**

Given proposition elements in different places of the parse tree, we need to find a sub-tree that covers all those nodes. On the sub-tree, there is a path from the root node to all the proposition elements. To get a most specific sub-tree, its root should minimally cover all the proposition elements. In other words, we need to find the lowest common parent of the proposition elements.

For this purpose, we can simply compare the paths from the root to all leaf (element) nodes and take a common node that is the farthest away from the root. In Fig. 4, the lowest common parent of *fellow* (*THERE*) is S and that of *enlisted* (*fellow*, *soldier*) is NP.

- **Grow sub-trees**

After the lowest common parent (lcp) is determined, we grow a sub-tree for each proposition element with the lcp as the root and the element as a leaf node by “moving up” the tree. In order to make the sub-tree syntactically well-formed, we try to grow all branches by including all the sibling nodes and branches except where **pruning** is possible.

Pruning is applied to sub-trees decided to be subordinate or ancillary, whose absence does not affect the grammaticality of the resultant sentence. Using linguistic knowledge, we use two pruning rules:

- Prune the left or right sub-tree with the root node of SBAR or SBARQ and all its left or right siblings.
- Prune the left or right sub-tree with the root node of CC and all its left or right siblings.

The rules are aimed to eliminate detachable subordinate clauses and coordinate constituents. In Fig. 4, when growing *fellow* (*THERE*) by moving up the tree, we encounter a node SBAR as the sibling of (NP, (DT: *a*, JJ: *young*, NN: *fellow*)), so the whole sub-tree with SBAR as the root is pruned. Moving up one level, (NP, (DT: *a*, JJ: *young*, NN: *fellow*)) grows into (NP, (NP, (DT: *a*, JJ: *young*, NN: *fellow*))).

```
Input: parse tree $T$, propositions $Prop = P(N_1, N_2, \ldots)$
Output: sub-tree $ST(Prop)$ covering $Prop$
1. Find the lowest common parent, $CP$, of $P, N_1, N_2, \ldots$ in $T$;
2. For each element $e$ in $Prop$:
       Grow a sub-tree $ST(CP, e)$ with $CP$ as the root and $e$ as a leaf;
3. Merge all sub-trees $ST(CP, e)$ into one sub-tree $ST(Prop)$;
4. If the root node of $ST(Prop)$, $CP$, is NP
       Adjust $ST(Prop)$;
```

Fig. 5. Top-level algorithm of sub-tree deduction.

- **Merge sub-trees into one**

With the grown sub-trees sharing a common root, we next merge them into one sub-tree that represents the whole p-sentence. Essentially, the merging process is to adjoin same-root sub-trees as branches of a bigger sub-tree. In this process, redundant branches are eliminated.

In Fig. 4, we can grow two identical sub-trees for *fellow* (*THERE*): (S (NP (NN: *THERE*)) (VP (VBD: *was*, ADVP (RB: *once*), NP, (NP, (DT: *a*, JJ: *young*, NN: *fellow*))))), which are merged into one copy, corresponding to the p-sentence: *THERE was once a young fellow*.

- **Adjust the sub-tree**

The deduced sub-tree is expected to represent a complete sentence, which means its root must be S. On the other hand, the sub-tree should represent a proposition, which is backboned by NPs and VPs. We find that there are two major root nodes: S and NP. In the former case, we directly output the sub-tree; in the latter, we need to adjust the structure of the sub-tree.

In almost all cases, the NP-rooted sub-tree represents a noun phrase with a clause modifier. Functionally, the head NP plays a role in the clause and can be moved into the clause at an appropriate place, so that the root of the sub-tree becomes S. The following lists the major cases of an NP-rooted sub-tree and the adjusted result.

- (NP$_0$, (NP$_1$, SBAR (S$_0$ (. . .))))) → (S, (NP$_1$, (S$_0$ (. . .))))
- (NP$_0$, (NP$_1$, SBAR (WHNP, S$_0$ (. . .))))) → (S, (NP$_1$, (S$_0$ (. . .))))
- (NP$_0$, (NP$_1$, SBAR (WHNP, VP (. . .))))) → (S, (NP$_1$, VP (. . .)))
- (NP$_0$, (NP$_1$, SBAR (WHPP, S$_0$ (. . .))))) → (S, (NP$_1$, (S$_0$ (. . .))))

In Fig. 4, the merged sub-tree for *enlisted* (*fellow*, *soldier*) represents the sentence: *a young fellow who enlisted as a solider* . . . with the (NP, (NP, SBAR (WHNP, S (VP, . . .))))) structure. After *a young fellow* (NP) is moved to the inner sentence, we come up with *a young fellow enlisted as a soldier* . . . with the (S (NP, S (VP, . . .))) structure.

Fig. 5 shows the top-level algorithm of sub-tree deduction.

4.1.2. A complete example

Now we illustrate the algorithm of sub-tree deduction by walking through a complete example, sentence (5) with the four propositions shown in (5.6). We show an annotated parse tree in Fig. 6 to facilitate the discussion. Note that the boxed nodes are proposition elements, the shaded nodes are the lowest common parents, and the "X" indicates pruning places.

- **Find the lowest common parent (lcp)**

The lcp of *fellow* (*THERE*) is the top-level S. The lcp's of *enlisted* (*fellow*, *soldier*) and *foremost* (*fellow*) are both NP. The lcp of *rained* (*bullets*) is VP.

- **Grow sub-trees**

For *fellow* (*THERE*), starting from *fellow* and *THERE*, we grow the same sub-tree: (S (NP (NN: *THERE*)) (VP (VBD: *was*, ADVP (RB: *once*), NP, (NP, (DT: *a*, JJ: *young*, NN: *fellow*))))). Note that the SBAR branch is pruned, as indicated in the figure.

Similar operations apply to enlisted (fellow, soldier), foremost (fellow) and rained (bullets).

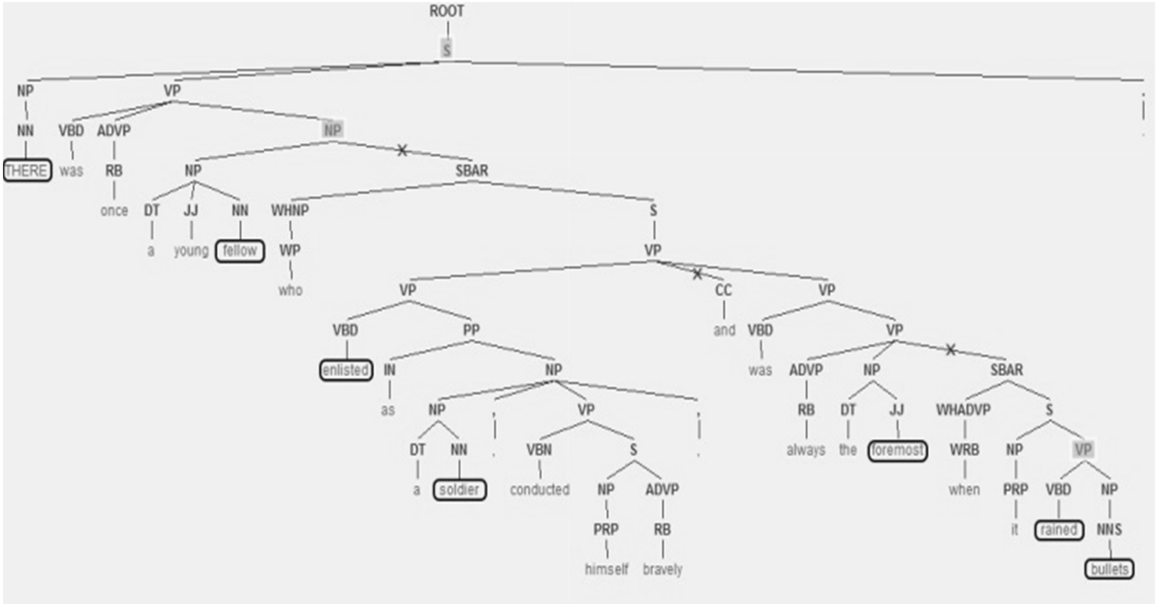


Fig. 6. Annotated parse tree of example sentence (5.5). The boxed nodes are proposition elements, the shaded nodes are the lowest common parents, and the “X” indicates pruning places.

- **Merge sub-trees into one**

For *fellow* (*THERE*), the two identical sub-trees merge into one: (S (NP (NN: *THERE*)) (VP (VBD: *was*, ADVP (RB: *once*), NP, (NP, (DT: *a*, JJ: *young*, NN: *fellow*))))).

For *enlisted* (*fellow*, *soldier*), the merged sub-tree is (NP (NP (DT: *a*, JJ: *young*, NN: *fellow*)), SBAR (WHNP (WP: *who*), S (VP (VP (VBD: *enlisted*, PP (IN: *as*, NP (NP (DT: *a*, NN: *soldier*),:, VP (VBN: *conducted*, S (NP (PRP: *himself*), ADVP (RB: *bravely*))),:,))))))).

For *foremost* (*fellow*), the merged tree is (NP (NP (DT: *a*, JJ: *young*, NN: *fellow*)), SBAR (WHNP (WP: *who*)), S (VP (VP (VBD: *was*, VP (ADVP (RB: *always*), NP (DT: *the*, JJ: *foremost*)))))).

For *rained* (*bullets*), the two identical sub-trees merge into one: (VP (VBD: *rained*, NP (NNS: *bullets*))).

- **Adjust the sub-tree and output the p-sentence**

For *fellow* (*THERE*), the root node is S and no adjustment is needed. The corresponding p-sentence is ***THERE was once a young fellow***.

Similar operations apply to *enlisted* (*fellow*, *soldier*), with the corresponding p-sentence of ***a young fellow enlisted as a soldier, conducted himself bravely***, *foremost* (*fellow*), with the p-sentence of ***a young fellow was always the foremost***, and *rained* (*bullets*), with the p-sentence of ***rained bullets***. Note that the last sentence is incomplete because we have not included the pronoun “it”, which cannot be resolved to a meaningful NP, as a proposition element.

4.1.3. P-sentence extraction evaluation

Because p-sentence extraction is central to proposition-based extraction and p-sentences are the building blocks of the summary, we evaluate their quality in this part. We assume that a desirable p-sentence should be grammatical and all the p-sentences of an original sentences together should convey all the information of their parent sentence.

Therefore, we recruited two human judges to assess the *grammaticality* and *information coverage* of the extracted p-sentences. To control the size of test data, we sampled 200 original sentences (about 5%) from all the sentences in a dataset of 50 fairy tales (see Section 5.2 for details), which were split into 1093 p-sentences in total using our algorithm.

Grammaticality is scored per p-sentence. Each of the two judges was asked to read all the 1093 p-sentences and gave a score on a scale of 3: 3 means grammatical, 1 means not grammatical, and 2 is the borderline case (e.g., fragments). Information coverage is scored per original sentence. After reading all the p-sentences of a corresponding original sentence, each of the two judges was to judge how well the p-sentences as a whole covers the original information by

Table 3
Average human scores for P-sentence extraction.

	Grammaticality	Information coverage
Judge 1	2.83	2.77
Judge 2	2.76	2.65

giving a 3-point score. Similarly, 3 means covering all the important information, 1 means missing some important information, and 2 is the borderline case. Table 3 lists the average scores for both judges.

The inter-judge agreement is measure by Cohen's Kappa, which is 0.67 for grammaticality and 0.59 for information coverage, indicating good agreement. The scores in Table 3 suggest that the extracted p-sentences are mostly grammatical, and most of a sentence's information is covered by the p-sentences it is split into.

4.2. Proposition-level extractive summarization

In this section, we will flesh out the details of the summarization module, i.e., Fig. 3. Proposition is pivotal in that it links the cognitive model of text comprehension and the summarization module.

4.2.1. Proposition ranking

Ranking is not unfamiliar to many traditional extractive summarization approaches, which is often motivated by including the most important information in the summary. But for our task, ranking is motivated by finding the **cognitively salient and coherent** information. Owing to the cognitive model of text comprehension/coherence, the final-state episodic memory contains all text words with their activation scores. The higher the score, the more salient the word in a cognitive sense (i.e., the easier the word is remembered). More importantly, the highest ranking words or word groups (propositions) must be well connected to each other because of the spreading activation mechanism of the cognitive model. This is how we assimilate coherence into the information selection stage of summarization. In comparison, a coherence account is unavailable for most other ranking-based summarization schemes.

With scored words in the episodic memory, we consider a proposition *Prop* made up of a predicate P and m arguments: $Prop = P\ (N_1, \ldots N_m)$, with all proposition elements having an activation score $AS(P), AS(N_1), \ldots AS(N_m)$. According to the propositional structure, $N_1, \ldots N_m$ are parallel to each other and P is associated with them all. So we define the ranking score (*RS*) of *Prop* as:

$$RS(P(N_1 \ldots N_m)) = AS(P)\sum\nolimits_{i=1}^{m} AS(N_i)$$

Proposition ranking is then based on the ranking scores of all propositions.

4.2.2. Sentence realization

Based on p-sentence extraction, realizing propositions as sentences (in fact, p-sentences) is straightforward. To the extracted p-sentences we apply simple modifications to make them real sentences, such as sentence-initial capitalization and sentence-ending punctuation.

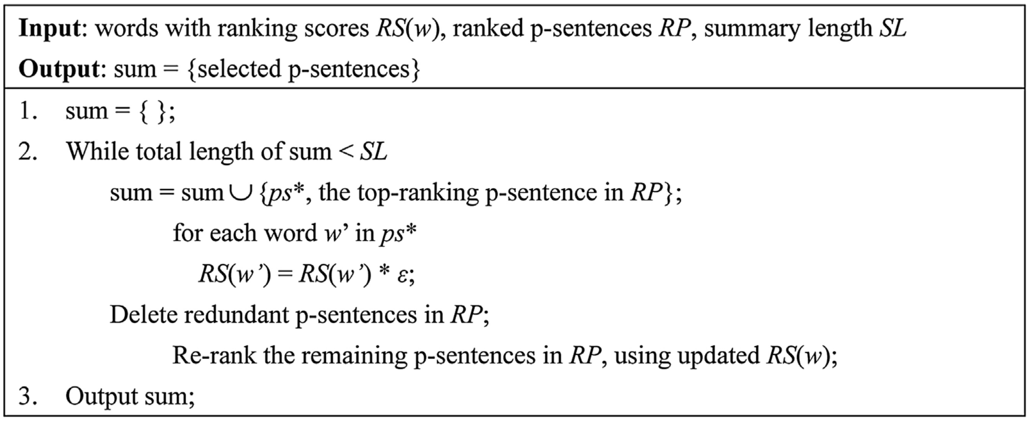
We manually checked all the 289 p-sentences of a text ("Bearskin") from the experimental dataset. It turns out that most of them (282) are grammatical. The ungrammatical cases are all due to parsing errors ("*Thee a coat and a cloak.*") and incomplete propositional structures ("*Rained bullets*.").

After sentence realization, we come up with ranked p-sentences that can be used for summarization by directly using the ranking score of the corresponding propositions. In other words, for a p-sentence PS_i and its corresponding proposition $Prop_i$,

$$RS(PS_i) = RS(Prop_i)$$

Alternatively, we can also discount long p-sentences by sentence length normalization. Suppose $Words(PS_i)$ denotes all the words in PS_i, then

$$RS(PS_i) = RS(Prop_i)/|Words(PS_i)|$$



```
Input: words with ranking scores $RS(w)$, ranked p-sentences $RP$, summary length $SL$
Output: sum = {selected p-sentences}
1.  sum = { };
2.  While total length of sum < $SL$
        sum = sum ∪ {$ps$*, the top-ranking p-sentence in $RP$};
            for each word $w'$ in $ps$*
                $RS(w') = RS(w') * \varepsilon$;
        Delete redundant p-sentences in $RP$;
            Re-rank the remaining p-sentences in $RP$, using updated $RS(w)$;
3.  Output sum;
```

Fig. 7. Algorithm of P-sentence selection.

4.2.3. P-sentence selection

The selection of ranked p-sentences should follow two principles. First, the selected p-sentences rank as high as possible, so that they are not only cognitively salient by themselves, but also well connected to each other. Second, the selected p-sentences overlap as little as possible.

In summarization, sentence overlap or redundancy is generally avoided. For our proposition-based scheme, this problem is exacerbated by p-sentence extraction. Since proposition elements span different sections of the parse tree, the p-sentences of an original sentence may be nearly identical or subsume each other.

The p-sentence selection algorithm is presented in Fig. 7

Summary-worthy p-sentences are selected iteratively until the summary length is reached. In each iteration, we select the top ranking p-sentence ps* and then discount the ranking score of all the words in ps* by multiplying ε (=0.9 in our experiments). Redundant sentences are determined by both string comparison and cosine similarity (=0.75 in our experiments). The remaining sentences are re-ranked using the updated word scores to further avoid redundancy.

4.2.4. P-sentence ordering

Since the cognitive model works only for a **single** narrative text, the summaries to be produced are single-document summaries by nature. To output the final summary, the selected p-sentences are textually ordered, i.e., according to the position of their subsuming original sentences in the original text. P-sentences belonging to the same original sentence are ordered according to their string positions in the original sentence.

5. Experiments with event-centric news and fairy tales

In order to evaluate the effectiveness of the cognitive model of coherence for summarization, we experimented on two kinds of dataset: event-centric news and fairy tales. Essentially, the datasets are narrative, which fit the proposition-based mechanism of the cognitive model.

We select event-centric news and fairy tales for experimentation mainly because the data are freely available and copyright-free. The news articles are also different from the fairy tales in content and style, which provides an opportunity to compare the model's effectiveness on two different kinds of narrative text. Note that presently our model deals only with single-document summarization.

5.1. Event-centric news

In this section, we report the experimental results on the selected DUC 01 and 02 datasets.

5.1.1. Data preparation

The DUC/TAC summarization track provides an abundance of newswire documents, together with human summaries that can be used for evaluation purposes. Among them, only DUC 01, 02, 03, and 04 have single-document

Table 4
Composition of the event-centric news datasets.

	# All news articles	# Event-centric news articles
DUC 01	600	249
DUC 02	567	388
Total	1167	637

summarization tasks.[4] But for DUC 03 and 04, the single-document summaries are very short – 10 words or 75 bytes – for which our approach can hardly show its advantage. In comparison, DUC 01 and 02 ask for 100-word single-document summaries, from which we selected event-centric news articles.

The news articles of DUC 01 and 02 are of two types: event-centric and entity-centric. The former focuses on a central event, such as a terrorist attack or an earthquake; the latter centers on a central person, thing, or other entities, such as a celebrity or a socio-cultural phenomenon. We manually selected the event-centric news articles from the DUC 01 and 02 datasets, totaling 637 documents. Table 4 lists the details.

The DUC annotators provided two human summaries for each news articles, which can be used as reference summaries in automatic evaluation described in the following.

5.1.2. Experimental design

The DUC summarization tasks require single-document summaries of a fixed length: 100 words. We match this length by generating 100-word summaries based on the cognitive model. The evaluation metric is the widely accepted ROUGE (Lin, 2004). Admittedly, ROUGE is a good measure of a summary's information coverage, not its coherence. However, we regard it as an indirect measure of coherence. On the one hand, ROUGE measures how similar an automatic summary is to the human-written reference summary, which is reasonably coherent. On the other hand, coherence underlies information selection according to our cognitive model. The selected information is simultaneously (cognitively) important and coherent, so a high score on information coverage should indicate good coherence. ROUGE is also an expedient choice as manually evaluating thousands of summaries is currently unaffordable.

As the success of the cognitive model depends considerably on its knowledge base – the semantic network, we will first evaluate the different ways of its construction: using different corpora (Wiki, Reuters) and different semantic models (LSA, LDA). Note that the use of updatable LSA/LDA enables us to combine Wiki and Reuters in an incremental way (Wiki&Reuters) and observe the effect. On the Wiki/Wiki&Reuters corpus, the LSA reduced dimensionality and the LDA number of topics are both set to be 400; on the Reuters corpus, both are 100.

With the best cognitive model, we compare two different ways of using the model output (in the episodic memory) to generate summaries: proposition-based summarization and sentence-based summarization. Proposition-based summarization is the approach described in Section 4.2, using p-sentences to compose summaries. By contrast, sentence-based summarization uses the original sentences selected by ranking them with scores calculated as the sum of the word activation scores. This is a straightforward application of the cognitive model to extractive summarization and an implementation of Lemaire et al. (2005), which shows that selecting sentences based on values calculated from the CI model highly correlates with the human selection of sentences to make up a narrative summary. What we are interested in is whether summarizing on the proposition level can improve on summarizing on the sentence level.

Next, the summaries generated from the best model and best summarization scheme are compared with baseline summaries and peer summaries that participated in DUC. The baseline summaries are the "Lead" summaries composed of the first 100 words – a strong baseline for news summarization (Brandow et al., 1995). The two sentence scoring schemes – normalized or un-normalized – are also evaluated.

5.1.3. Evaluation results

We first present the ROUGE scores, including ROUGE-1 (unigram overlap), ROUGE-2 (bigram overlap), ROUGE-SU4 (skip bigrams, up to the distance of four), of using different semantic networks as the cognitive basis. The other summarization parameters are held to be the same: all the summaries are proposition-based using un-normalized

[4] http://www-nlpir.nist.gov/projects/duc/pubs.html.

Table 5
Comparison of semantic network constructions.

	ROUGE-1	ROUGE-2	ROUGE-SU4
LSA + Reuters	0.412*	0.124*	0.185
LSA + Wiki	0.393*	0.118*	0.169*
LSA + Wiki&Reuters	**0.423**	**0.137**	**0.191**
LDA + Reuters	0.401*	0.120*	0.179*
LDA + Wiki	0.386*	0.115*	0.164*
LDA + Wiki&Reuters	0.417	0.129*	0.186

Table 6
Comparison of summarization schemes.

	ROUGE-1	ROUGE-2	ROUGE-SU4
Proposition + Un-normalized	0.423*	0.137	0.191
Proposition + Normalized	**0.434**	**0.141**	**0.196**
Sentence + Un-normalized	0.411*	0.128*	0.185*
Sentence + Normalized	0.417*	0.133*	0.190

Table 7
Comparison of summaries for DUC 01/02 event-centric articles.

	ROUGE-1	ROUGE-2	ROUGE-SU4
DUC 01			
Lead (Baseline)	0.429*	0.140*	0.192
Best DUC peer	0.433	0.142	0.193
Our method	**0.437**	**0.148**	**0.199**
DUC 02			
Lead (Baseline)	0.425	0.128*	0.187
Best DUC peer	0.427	0.133	0.191
Our method	**0.432**	**0.137**	**0.194**

sentence scoring. In the following tables, statistical significance is measured by a paired two-tailed *t*-test between the best score and all the others in the same category. The mark * indicates the observed statistical significance ($p < 0.05$).

According to the results, the LSA-based versions consistently outperform their LDA-based counterparts, which lends credence to the wide use of LSA as a cognitive modeling tool in many domains. The specialized Reuters corpus works better than the generalized Wiki corpus, showing that the cognitive model works better on documents similar to the training corpus. Not surprisingly, enlarging the size of the corpus boosts performance further.

Using the best cognitive basis (LSA + Wiki&Reuters), we compare four summary variants: proposition-based/sentence-based summarization + normalized/un-normalized sentence scoring. Table 6 shows the results.

Proposition-level extraction obviously outperforms sentence-level extraction, which confirms our hypothesis about the significance of proposition in both cognitive modeling and summarization. Normalizing p-sentences also works, which suggests that as length increases, news sentences are likely to include non-essential information.

Based on those results, we proceed to compare the best summaries produced by our system (LSA + Wiki&Reuters, proposition-based, normalized sentence scoring) with DUC peer summaries. There are 11 peer summaries for each DUC 01 source document and 13 peer summaries for each DUC 02 source document, most of which are produced by different systems. Therefore, the summaries for DUC 01 and DUC 02 are evaluated separately. Note that for the best DUC peer, we only select the summaries for the event-centric articles.

The results in Table 7 are hard evidence that our method outperforms the best known systems, although the superiority is not very obvious. In fact, single-document news summarization has been long given little credit to its research value. Part of the reason is the simplicity and robustness of the Lead baseline, as is shown by the little gap (and with no statistical significance, as we observe) between the Lead and the best DUC peer system. The difference between the

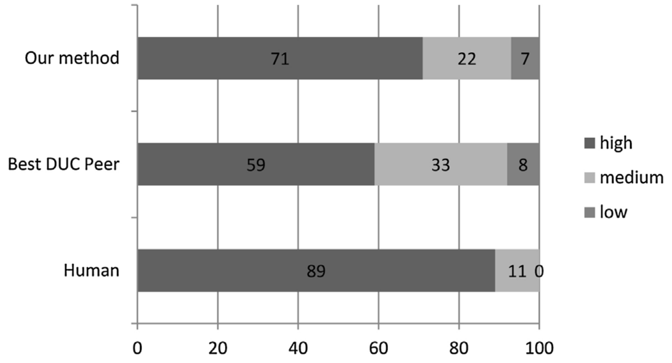


Fig. 8. Human assessment of summary coherence.

Lead and our method, however, is more noticeable. In Table 7, the difference between our method and the Lead is sometimes statistically significant and our method is the better in absolute numbers.

Please note that the DUC 02 results for all systems are consistently poorer than the DUC 01 results. A major reason is that the DUC 01 reference summaries are extracts, i.e., original sentences from the source, and the DUC 02 reference summaries are abstracts, i.e., human-written sentences not necessarily found in the source. The system summaries are mostly extractive, including ours (proposition-level extracts), and there naturally exists a larger gap between extractive system summaries and abstractive reference summaries. Nonetheless, the success of our method in the face of both extractive and abstractive reference summaries testifies the effectiveness and robustness of proposition-level extraction (i.e. p-sentence extraction).

To verify that our method indeed produces more coherent summaries, we adopt human assessment. Since human rating is labor-intensive, we controlled the size of test sets by randomly selecting 100 source documents from the DUC 01 + DUC 02 pool and using 3 summaries for each of them: the human summary, the best DUC peer summary, and our summary. One human judge, a native speaker of English, was employed to rate all the summaries according to their degree of coherence. She was directed to rate each summary as having “high”, “medium”, or “low” coherence based on how smooth (coherent) she thought the passage was. She was given all the 3 summaries of the same source at once and was told that they were all about the same event, but the order of the summaries was randomized so that no pattern could be discerned. This design enables the human rater to compare summaries of the same document and give more consideration to the ratings. We also made it clear that high, medium, and low do not have to be given only once for the 3 summaries. Two or three summaries can be tied with “high” or “medium”, for example.

The result of the human assessment is shown in Fig. 8.

Obviously, the human summaries are considerably more coherent than automatic summaries. But it is encouraging to find out that over 70% of the summaries produced by our cognitive model are highly coherent, 12 percent higher than the best DUC peer summaries.

Now it is interesting to ask whether the cognitive model of narrative text comprehension and coherence also works for non-narrative news text (i.e., entity-centric text). Theoretically, non-narrative news text lacks “plot development” that can be well captured by the cyclic reading process, so the model should not work well. In order to test this hypothesis, we also experimented on all the entity-centric news articles from the DUC 01 and 02 datasets. According to Table 4, there are a total of 530 such articles. Table 8 shows the result, using the same model and summarization scheme. Note that for the best DUC peer, we only select the summaries for the entity-centric articles.

This time, the summaries produced by our method perform poorly, defeated even by the Lead baseline. But none of the differences is statistically significant. We conjecture that no further human verification is needed in this case. Since the different results from Tables 7 and 8 can only derive from the different natures of the text, we conclude that the cognitive model and proposition-based approach works best with narrative text.

5.1.4. Comparison with Fang and Teufel (2014)

A similar work by Fang and Teufel (2014) models the cognitive mechanism and memory retention in summarization based on an early work (Kintsch and van Dijk, 1978). Their implementation also includes dependency parsing-based proposition extraction. So it is interesting to compare their result on the DUC 02 dataset with ours. Nevertheless, they

Table 8
Comparison of summaries for DUC 01/02 entity-centric articles.

	ROUGE-1	ROUGE-2	ROUGE-SU4
DUC 01			
Lead (Baseline)	0.427	0.139	0.189
Best DUC peer	0.430	0.141	0.190
Our method	0.428	0.137	0.185
DUC 02			
Lead (Baseline)	0.429	0.138	0.192
Best DUC peer	0.432	0.143	0.192
Our method	0.427	0.136	0.188

Table 9
Comparison of summaries for ALL the DUC 02 articles.

	ROUGE-1	ROUGE-2	ROUGE-SU4
Lead (Baseline)	0.427*	0.132	0.189
Best DUC peer	0.430	0.136	0.191
Our method	**0.432**	**0.137**	**0.192**

Table 10
Fairy tale dataset length statistics.

	Max	Min	Average
Original text (# words)	48,190	461	4025.6
Summary (# words)	1594	74	396.3
Summary/Original Ratio	0.52	0.01	0.16

do not make a distinction between event-centric and entity-centric articles. So in an additional experiment, we apply our best system on the entire DUC 02 dataset. The result is shown in Table 9.

Note that it is only for comparison with Fang and Teufel (2014) that we evaluate on the DUC 02 dataset as a whole. Obviously our ROUGE scores are higher than the Lead baseline, and since Fang and Teufel's (2014) ROUGE result is inferior to the Lead, we conclude that our method outperforms theirs.[5]

5.2. *Fairy tales*

A more typical genre of narrative text is story such as fairy tales. In this set of experiments, we use fairy tales as they have clear plots and narrative structures, which is ideal for the cognitive model.

5.2.1. *Data preparation*

The fairy tales used as experimental data are mostly by Brothers Grimm and Hans C. Anderson because those classic works are copyright-free and quality human summaries can be found on dedicated websites[6] or Wikipedia. Using free online resources,[7] we built a dataset of 50 fairy tales, each accompanied with a human summary. All the human summaries are manually checked to ensure that they are truly descriptive, not evaluative, summaries (Ceylan and Mihalcea, 2009). Table 10 lists the length statistics.

[5] It is difficult to directly compare their result with ours because they use a different ROUGE score, ROUGE-L and the ROUGE scores may vary with different system configurations. Using the Lead baseline (they called it "First n words") enables us to make a comparison.

[6] http://www.comedyimprov.com/music/schmoll/tales.html.

[7] http://www.surlalunefairytales.com/.

Table 11
Comparison of semantic network constructions.

	ROUGE-1	ROUGE-2	ROUGE-SU4
LSA + FT	0.440*	0.097	0.170
LSA + Wiki	0.447	0.101	0.176
LSA + Wiki&FT	**0.452**	**0.102**	**0.179**
LDA + FT	0.449	0.097	0.174
LDA + Wiki	0.444*	0.100	0.174
LDA + Wiki&FT	0.444*	0.098	0.173

Unlike the news articles used in the first set of experiments, both the fairy tale text lengths and compression (summary/original) ratios vary a lot. So for an automatic summary, we match its length to the human summary length instead of taking a fixed length or ratio, such as the 100 words for news articles.

5.2.2. *Experimental design*

The evaluation objects are similar to those for the event-centric news. First, we compare the different ways of constructing the semantic network to feed the cognitive model: using LSA/LDA and 3 different corpora: Wiki, FT, Wiki&FT. On the Wiki/Wiki&FT corpus, the LSA reduced dimensionality and the LDA number of topics are both set to be 400; on the FT corpus, both are 100. Next, we test the efficacy of proposition-based summarization scheme and sentence normalization.

For summary comparison, no peer summaries are available. So we will compare our summaries with those produced with 3 well-known and popular methods: Luhn's (1958) algorithm, MEAD (Radev et al., 2004) as implemented in Mihalcea and Ceylan (2007), and TextRank (Mihalcea and Tarau, 2004). Luhn's classic algorithm is one of the best known for single-document summarization. MEAD and TextRank are popular summarization methods that have been applied to story summarization (Mihalcea and Ceylan, 2007). All of them produce extractive summaries based on sentence scoring by using word frequency, position information, sentence relation, etc. As in the previous set of experiments, we produce "Lead" summaries for comparison.

Both automatic evaluation and human evaluation will be done for this set of experiments. For the automatic evaluation, we still use the ROUGE measures for reasons explained in Section 5.1.2. But this smaller dataset also makes it possible to do human evaluation so that coherence can be directly evaluated. Using the best summaries from previous results, we ask 2 human judges (both native speakers of English) to score 4 different summaries for each of the 50 fairy tales, on a scale of 5 points, in response to the following statements.

S1: This summary gives me enough information to understand what the story is about.

S2: The sentences in the summary of the story are coherent and well connected to each other.

S3: Except for the last sentence, the sentences in the summary are grammatical and complete.

The human judges were asked to indicate to what degree they agree with the statements. Complete agreement with a statement leads to a score of 5 and complete disagreement leads to a score of 1. The three statements are aimed to evaluate *informativeness*, *coherence*, and *grammaticality* respectively. Note that because of the truncation to meet the word limit, the last sentence of an automatic summary is probably incomplete. This factor should be excluded in grammaticality evaluation.

5.2.3. *Evaluation results*

Using different semantic network constructions to build the cognitive model, we report the ROUGE scores in Table 11. As in the first set of experiments, the other summarization parameters all take default settings, i.e., proposition-based summarization and un-normalized sentence scoring. For all the ROUGE results (Tables 11–13), the mark * indicates statistical significance ($p < 0.05$) on a paired two-tailed t-test between the best score and all the others in the same category.

Compared with Table 5, the results are less consistent. Using the specialized FT corpus, the LSA-based model underperforms the LDA-based model. But using the larger Wiki and Wiki&FT corpora, the LSA-based model performs better. Interestingly, if LDA is used, a larger corpus does not necessarily help fairy tales whereas it does help news (Table 5). Since LDA works with topic modeling, a plausible explanation is that the topics of fairy tales, which include

Table 12
Comparison of summarization schemes.

	ROUGE-1	ROUGE-2	ROUGE-SU4
Proposition + Un-normalized	**0.452**	**0.102**	**0.179**
Proposition + Normalized	0.446	0.098	0.172
Sentence + Un-normalized	0.430*	0.097	0.170
Sentence + Normalized	0.412*	0.093*	0.162*

Table 13
Comparison of summaries for fairy tales.

	ROUGE-1	ROUGE-2	ROUGE-SU4
Our method	**0.452**	**0.102**	**0.179**
Lead	0.395*	0.080*	0.147*
Luhn (1958)	0.410*	0.088*	0.157*
MEAD	0.419*	0.091	0.160*
TextRank	0.421*	0.092	0.163*

particular characters and settings, are more specific than those of news and the mostly non-fairy tale text in Wiki cannot help in finding such topics to build the semantic network. Combining Wiki and FT introduces a lot of noise to fairy tale topics and is thus counterproductive. The LSA-based models, on the other hand, are more robust and consistently benefit from larger training corpora.

We observe that when the training corpus is small and specialized, LDA is more effective than LSA. But when the training corpus is large and generic, LSA shows pronounced advantage. Similar to the results on event-centric news, LSA + all available training data (Wiki&FT) gives the best performance. Based on this construction of the semantic network, we compare summaries produced from the different combinations of proposition-based/sentence-based summarization and normalized/un-normalized sentence scoring.

Using the cognitive model output, proposition-level extraction proves more effective than sentence-level extraction for fairy tales as well as event-centric news. But unlike the summarization of event-centric news, sentence normalization is counterproductive for fairy tale summarization. This shows a textual difference between news and fairy tales. In terms of narrative content (proposition elements), longer sentences in news contain more noise (non-narrative content). During p-sentence extraction, such noise is usually indispensable for syntactic completeness. By contrast, sentences in fairy tales contain mostly narrative content and during p-sentence extraction, long sentences can often be decomposed into shorter p-sentences. An illustrative case is shown in examples (7) and (8).

(7)

(original sentence) *SQUADS of workers fanned out across storm-battered Louisiana yesterday to begin a massive rebuilding effort after Hurricane Andrew had flattened whole districts, killing two people and injuring dozens more, agencies report from Florida and New Orleans.*

(p-sentences)

SQUADS of workers fanned out across storm-battered Louisiana yesterday to begin a massive rebuilding effort.
Hurricane Andrew had flattened whole districts, killing two people and injuring dozens more.
Agencies report from Florida and New Orleans.

(8)

(original sentence) *So long as the war lasted, all went well, but when peace was made, he received his dismissal, and the captain said he might go where he liked.*

(p-sentences)

The war lasted.
Peace was made.
He received his dismissal.
The captain said.

Table 14
Average human scores for the fairy tale summaries.

	Informativeness	Coherence	Grammaticality
Human	4.32*	4.63*	4.88*
Our method – proposition-level	**3.27**	**3.39**	**3.87**
Our method – sentence-level	3.10	2.95*	3.95
TextRank	2.98*	2.87*	3.84

He might go.
He liked.

(7) is selected from the news dataset and (8) from the fairy tales dataset. Obviously, the p-sentences of (8) are more compact than those of (7) in terms of narrative content. Consequently, sentence normalization for fairy tales is not helpful.

Next, we compare the best summaries produced by our system (LSA + Wiki&FT, proposition-based, un-normalized sentence scoring) with 4 peer summaries introduced in Section 5.2.2: Lead, Luhn (1958), MEAD, and TextRank. For fairness, except for Lead, the sentence scoring for the peer summaries are un-normalized. The result is shown in Table 13.

It seems that the superiority of our method over the peer systems is obvious on fairy tales. This result, joined with the result on event-centric news (Table 7), testifies the efficacy and robustness of the cognitive model and proposition-based approach to narrative summarization. Interestingly, the Lead summaries of fairy tales perform the worst, showing that a commonly held strong baseline for single-document summarization does not work well in a typical narrative domain. Therefore, developing new and powerful summarization techniques for narrative text is a very meaningful endeavor.

ROUGE scores can indirectly measure the coherence of the output summaries. But the human evaluation of coherence provides a more direct yardstick. Moreover, since coherence is ultimately measured in human terms, it makes good sense to validate the end product with human criteria.

For each of the 50 fairy tales, we provide two human judges with 4 summaries: one human summary, one best peer summary (TextRank, according to Table 13), and two summaries produced by our method which differ only in the level of sentence extraction – one uses proposition-level extraction and the other sentence-level extraction. Similar to the assessment for event-centric news summaries, the judges were given all the 4 summaries of the same source at once and were told that they were summaries of the same story. But the judges had no access to the original stories and could find no pattern because the 4 summaries were randomized. As described in Section 5.2.2, the judges were directed to score the summaries as they responded to the statements.

As is introduced in Section 5.2.2, we asked two human judges to score summaries for coherence as well as informativeness and grammaticality. The human assessment of informativeness will lend further credence to the ROUGE metric. Grammaticality is also evaluated because it is important to find out even though proposition-level extractive summarization renders more informative/coherent summaries than sentence-level extractive summarization, whether it is done at the cost of grammaticality.

For each scoring category, inter-judge agreement is measured by Cohen's Kappa, which ranges between 0.48 and 0.63, indicating good agreement. Then we take the average of the two human scores over the 50 fairy tales on each category and report the result in Table 14. In this table, statistical significance of the proposition-level extractive summaries ("Our method – proposition-level") against all the other summaries is indicated by * ($p < 0.01$) on a paired two-tailed t-test.

The "proposition-level" version represents the best output of our method. Informatively, it is superior to the "sentence-level" version and TextRank summaries, which is consistent with the ROUGE results. In terms of coherence, the proposition-level version outperforms the sentence-level version and TextRank significantly, proving the validity of the cognitive model-driven coherence when effectively integrated into summarization. This is also hard evidence that the proposition-level extractive summarization outperforms sentence-level extractive summarization not only in essential information coverage, but also (and more importantly) in coherence.

Are the gains in informativeness and coherence achieved at the cost of grammaticality? This concern is relieved by the small gap between the proposition-level version and the sentence-level version, the former being slightly better than TextRank. Such differences, however, are statistically insignificant.

A huge gap does exist between the human summaries and all the automatic summaries in all aspects, a cold fact showing that fairy tale summarization is indeed a challenge. The cognitive model and the summarization scheme pioneered by our work, however, make a good attempt to take the challenge.

6. Conclusion and future work

In this paper, we have broken new ground in text summarization by presenting a novel approach to coherent narrative summarization with a cognitive model. From a cognitive perspective, coherence is interpreted as a built-in mechanism in text comprehension. Modeling such coherence is technically equivalent to modeling text comprehension.

The computational model of text comprehension and coherence is based on theoretical models from psychology and cognitive science. A semantic network is computed from a corpus to simulate knowledge stored in the long-term memory, and a proposition-based cyclic comprehension algorithm is proposed to model the human reading process and the interactions between different parts of the human memory. Upon completion of all the reading cycles, the episodic memory contains all proposition elements with their activation scores.

The scored proposition elements are used to select cognitively salient and coherent information for summarization. Different from most other extractive summarization approaches, we summarize on the proposition level. Propositions are first ranked according to the predicate-argument structure and the word activation scores in the episodic memory. Then they are realized as grammatical sentences, or p-sentences, that are the proper constituents of a summary. The highest ranking and non-redundant p-sentences are then selected for the summary.

The cognitive model-driven coherence works best on narrative text. Therefore, we experimented with two datasets of narrative text: event-centric news and fairy tales. On both datasets, our method outstrips peer systems, proving that for single-document narrative summarization, cognitive model-driven coherence can benefit both informativeness and coherence in the output summaries.

In future work, we will explore computerizing cognitive models other than Kintsch (1998) and compare their effects. Many model parameters, now heuristically set, can be learned from annotated data or stochastic modeling. The proposition processing is a promising direction for finer-level extractive summarization, but better tree-adjustment algorithms as well as a good integration of proposition ranking with the cognitive model set future agendas for this line of research.

Acknowledgments

We are very grateful to Oscar Lai and Yaoyun Zhang for their help with the evaluation. The work described in this paper was supported by a Shanghai Natural Science Fund (15ZR1443800), a Shanghai Social Sciences Fund (2013BYY003), the grant GRF PolyU 152094/14E, and NSFC 61272291.

References

Alonso i Alemany, L., Fuentes, F.M., 2003. Integrating cohesion and coherence for automatic summarization. In: Proceedings of EACL2003, Budapest, Hungary, pp. 1–8.

Anderson, J.R., 1976. Language, Memory and Thought. Erlbaum, Mahwah, NJ.

Barzilay, R., Elhadad, M., 1997. Using lexical chains for text summarization. In: Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization, pp. 10–17.

Barzilay, R., Lapata, M., 2005. Modeling local coherence: an entity-based approach. In: Proceedings of the 43rd Annual Meeting of the ACL, Ann Arbor, pp. 141–148.

Barzilay, R., Lapata, M., 2008. Modeling local coherence: an entity-based approach. Comput. Linguist. 34, 1–34.

Barzilay, R., Lee, L., 2004. Catching the drift: probabilistic content models, with applications to generation and summarization. In: HLT-NAACL Proceedings of the Main Conference, pp. 113–120.

Barzilay, R., McKeown, K., 2005. Sentence fusion for multidocument news summarization. Comput. Linguist. 31 (3), 297–328.

Berman, M.G., 2009. In search of decay in verbal short term memory. J. Exp. Psychol.: Learn. Mem. Cognit. 35 (2), 317–333.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent Dirichlet allocation. J. Mach. Learn. Res. 3 (4–5), 993–1022.

Brandow, R., Mitze, K., Rau, L.F., 1995. Automatic condensation of electronic publications by sentence selection? Inf. Process. Manag. 31 (5), 675–685.

Brennan, S.E., Marilyn, A.F., Carl, J.P., 1987. A centering approach to pronouns. In: Proceedings of ACL 1987, Stanford, CA, pp. 155–162.

Ceylan, H., Mihalcea, R., 2009. The decomposition of human-written book summaries. In: Proceedings of the 10th International Conference on Computational Linguistics and Intelligent Text Processing, pp. 582–593.

Christensen, J., et al., 2013. Towards coherent multi-document summarization. In: Proceedings of NAACL-HLT, Atlanta, GA, pp. 1163–1173.

Cohn, T., Lapata, M., 2008. Sentence compression beyond word deletion. In: Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), Manchester, pp. 137–144.

Conroy, J.M., Schlesinger, J.D., Goldstein, J., 2006. CLASSY tasked based summarization: back to basics. In: Proceedings of the Document Understanding Conference (DUC-06).

Cristea, D., Iftene, A., 2010. Discourse coherence – a built-in cognitive mechanism? In: Tufis, D., Forascu, C. (Eds.), Multilinguality and Interoperability in Language Processing with Emphasis on Romanian. Editura Academiei Române, Bucuresti, pp. 362–379.

Cristea, D., Ide, N., Romary, L., 1998. Veins theory: a model of global discourse cohesion and coherence. In: Proceedings of COLING/ACL'98, Montreal, pp. 281–285.

Elsner, M., Austerweil, J., Charniak, E., 2007. A unified local and global model for discourse coherence. In: Proceedings of NAACL HLT, Rochester, NY, pp. 436–443.

Fang, Y., Teufel, S., 2014. A summariser based on human memory limitations and lexical competition. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pp. 732–741.

Gernsbacher, M.A., 1990. Language Comprehension as Structure Building. Lawrence Erlbaum Associates, Hillsdale, NJ.

Gernsbacher, M.A., 1996. Coherence cues mapping during comprehension. In: Costermans, J., Fayol, M. (Eds.), Processing Interclausal Relationships in the Production and Comprehension of Text. Lawrence Erlbaum Associates, Mahwah, NJ, pp. 3–21.

Goyal, A., Riloff, E., Daume III, H., 2010. Automatically producing plot unit representations for narrative text. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp. 77–86.

Grosz, B.J., Aravind, K.J., Scott, W., 1995. Centering: a framework for modeling the local coherence of discourse. Comput. Linguist. 21 (2), 203–225.

Halliday, M.A.K., Hasan, R., 1976. Cohesion in English. Longman, London.

Hasler, L., 2004. An investigation into the use of centering transitions for summarization. In: Proceedings of CLUK'04, Birmingham, UK, pp. 100–107.

Hatzivassilogiou, V., Klavans, J.L., Holcombe, M.L., Barzilay, R., Kan, M.-Y., McKeown, K., 2001. SimFinder: a flexible clustering tool for summarization. In: Proceedings of the Workshop on Automatic Summarization, pp. 41–49.

Hoffman, M.D., Blei, D.M., Bach, F., 2010. Online learning for latent Dirichlet allocation. In: Lafferty, J., Williams, C.K.I., Shawe-Taylor, J., Zemel, R., Culotta, A. (Eds.), Advances in Neural Information Processing Systems 23. , p. 856.

Jing, H., 2000. Sentence reduction for automatic text summarization. In: Proceedings of the 6th Applied Natural Language Processing Conference, pp. 310–315.

Jing, H., McKeown, K., 2000. Cut and paste based text summarization. In: Proceedings of the Sixth Applied Natural Language Conference (ANLP-00) and the First Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-00), pp. 178–185.

Just, M.A., Carpenter, P.A., 1992. A capacity theory of comprehension: individual differences in working memory? Psychol. Rev. 99 (1), 122–149.

Kazantseva, A., 2006. An approach to summarizing short stories. In: Proceedings of the Student Research Workshop at the 11th Conference of the European Chapter of the Association for Computational Linguistics, pp. 55–62.

Kazantseva, A., Szpakowicz, S., 2010. Summarizing short stories. Comput. Linguist. 36, 71–109.

Kibble, R., Power, R., 2004. Optimizing referential coherence in text generation. Comput. Linguist. 30, 401–416.

Kintsch, W., 1988. The role of knowledge in discourse comprehension: a construction-integration model. Psychol. Rev. 95, 163–182.

Kintsch, W., 1998. Comprehension: A Paradigm for Cognition. Cambridge University Press, New York.

Kintsch, W., 2001. Predication. Cogn. Sci. 25, 173–202.

Kintsch, W., Mangalath, P., 2011. The construction of meaning. Top. Cogn. Sci. 3, 346–370.

Kintsch, W., van Dijk, T., 1978. Toward a model of text comprehension and production. Psychol. Rev. 85 (5), 363.

Klein, D., Manning, C.D., 2003. Accurate unlexicalized parsing. In: Proceedings of the 41st Meeting of the Association for Computational Linguistics, pp. 423–430.

Knott, A., Oberlander, J., O'Donnell, M., Mellish, C., 2001. Beyond elaboration: the interaction of relations and focus in coherent text. In: Sanders, T., Schilperoord, J., Spooren, W. (Eds.), Text Representation: Linguistic and Psycholinguistic Aspects. Benjamins, pp. 181–196.

Landauer, T.K., Foltz, P.W., Laham, D., 1998. Introduction to latent semantic analysis. Discourse Process. 25, 259–284.

Lapata, M., 2003. Probabilistic text structuring: experiments with sentence ordering. In: Proceedings of the Annual Meeting of ACL, Sapporo, Japan, pp. 545–552.

Lee, H., Peirsman, Y., Chang, A., Chambers, N., Surdeanu, M., Jurafsky, D., 2011. Stanford's multi-pass sieve coreference resolution system at the CoNLL-2011 shared task. In: Proceedings of the CoNLL-2011 Shared Task, pp. 28–34.

Lehnert, W.G., 1981. Plots units and narrative summarization. Cogn. Sci. 5 (4), 293–331.

Lehnert, W.G., 1999. Plot units: a narrative summarization strategy. In: Mani, I., Maybury, M.T. (Eds.), Advances in Automatic Text Summarization. MIT Press, Cambridge, MA, pp. 177–214.

Lemaire, B., Denhiere, G., Bellissens, C., Jhean-Larose, S., 2006. A computational model for simulating text comprehension. Behav. Res. Methods 38 (4), 628–637.

Lemaire, B., Mandin, S., Dessus, P., Denhière, G., 2005. Computational cognitive models of summarization assessment skills. In: Proceedings of the 27th Annual Conference of the Cognitive Science Society (CogSci'2005), pp. 1266–1271.

Lin, C.-Y., 2004. ROUGE: a package for automatic evaluation of summaries. In: ACL 2004 Workshop on Text Summarization Branches Out, Post-conference Workshop of ACL, pp. 74–81.

Lobo, P.V., de Matos, D.M., 2010. Fairy tale corpus organization using latent semantic mapping and an item-to-item top-n recommendation algorithm. In: Language Resources and Evaluation Conference – LREC 2010, European Language Resources Association (ELRA), Malta, pp. 1472–1475.

Luhn, H.P., 1958. The automatic creation of literature abstract. IBM J. Res. Dev. 2 (2), 159–165.

Mani, I., Gates, B., Bloedorn, E., 1999. Improving summaries by revising them. In: Proceedings of ACL99, College Park, MD, pp. 558–565.

Marcu, D., 1997. The rhetorical parsing of natural language texts. In: Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL-97), pp. 96–103.

Marcu, D., 1999. Discourse trees are good indicators of importance in text. In: Mani, I., Maybury, M.T. (Eds.), Advances in Automatic Text Summarization. MIT Press, Cambridge, MA, pp. 123–136.

Marcu, D., 2000. The Theory and Practice of Discourse Parsing and Summarization. The MIT Press, Cambridge, MA.

Mihalcea, R., Ceylan, H., 2007. Explorations in automatic book summarization. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 380–389.

Mihalcea, R., Tarau, P., 2004. TextRank: bringing order into texts. In: Proceedings of EMNLP, Barcelona, Spain, pp. 404–411.

Orăsan, C., 2003. An evolutionary approach for improving the quality of automatic summaries. In: Proceedings of the Multilingual Summarization and Question Answering—Machine Learning and Beyond Workshop, Sapporo, Japan, pp. 37–45.

Palmer, M., Fellbaum, C., Cotton, S., Delfs, L., Dang, H.T., 2001. English tasks: all-words and verb lexical sample. In: Proceedings of SENSEVAL-2: Second International Workshop on Evaluating Word Sense Disambiguation Systems, Toulouse, France.

Pastor, E.L., (PhD thesis) 2011. Text Summarization based on Human Language Technologies and its Applications. Universidad de Alicante.

Pastor, E.L., 2012. Text summarisation based on human language technologies and its applications. Proces. Leng. Nat. 48, 119–122.

Quesada, J., 2007. Creating your own LSA spaces. In: Landauer, T.K., McNamara, D.S., Dennis, S., Kintsch, W. (Eds.), Handbook of Latent Semantic Analysis. Erlbaum, Mahwah, NJ, pp. 71–88.

Radev, D., Jing, H., Styś, M., Tam, D., 2004. Centroid-based summarization of multiple documents. Inf. Process. Manag. 40, 919–938.

Řehůřek, R., 2011. Subspace tracking for latent semantic analysis. In: Advances in Information Retrieval, Volume 6611 of Lecture Notes in Computer Science. Springer, pp. 289–300.

Riedl, M.O., Young, R.M., 2010. Narrative planning: balancing plot and character. J. Artif. Intell. Res., 164–167.

Soricut, R., Marcu, D., 2006. Discourse generation using utility-trained coherence models. In: Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pp. 803–810.

Steyvers, M., Griffiths, T.L., 2008. Rational analysis as a link between human memory and information retrieval. In: Chater, N., Oaksford, M. (Eds.), The Probabilistic Mind: Prospects for a Bayesian Cognitive Science. Oxford University Press, Oxford, England, pp. 329–350.

Tapiero, I., (Postdoctoral thesis for the Habilitation á diriger des recherches) 2000. Construire une représentation mentale cohérente: Structures, relations et connaissances (Building a Coherent Mental Representation: Structures, Relations, and Knowledge). University of Lyon 2, Lyon, France.

Tapiero, I., 2007. Situation Models and Levels of Coherence: Towards a Definition of Comprehension. Lawrence Erlbaum Associates, Mahwah, New Jersey.

van den Broek, P., Risden, K., Fletcher, C.R., Thurlow, R., 1996. A ‘landscape’ view of reading: fluctuating patterns of activation and the construction of a stable memory representation. In: Britton, B.K., Graesser, A.C. (Eds.), Models of Understanding Text. Erlbaum, Mahwah, NJ, pp. 165–187.

van Dijk, T.A., Kintsch, W., 1983. Strategies of Discourse Comprehension. Academic Press, New York.

Wolf, F., Gibson, E., 2004. Paragraph-, word-, and coherence-based approaches to sentence ranking: a comparison of algorithm and human performance. In: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain, pp. 383–390.

Wolf, F., Gibson, E., 2006. Coherence in Natural Language. MIT Press, Cambridge, MA.

Yousfi-Monod, M., Prince, V., 2008. Sentence compression as a step in summarization or an alternative path in text shortening. In: COLING, Manchester, UK, pp. 139–142.

Zajic, D.M., Dorr, B.J., Lin, J., 2008. Single-document and multi-document summarization techniques for email threads using sentence compression. Inf. Process. Manag. 44 (4), 1600–1610.

Zhang, R., 2011. Sentence ordering driven by local and global coherence for summary generation. In: ACL 2011, Student Session, pp. 6–11.

Zwaan, R.A., Langston, M.C., Graesser, A.C., 1995. The construction of situation models in narrative comprehension: an event-indexing model? Psychol. Sci. 6 (5), 292–297.

Renxian Zhang is an Associate Professor in the Department of Computing, Tongji University, China. He received his PhD degree from the Department of Computing, the Hong Kong Polytechnic University. His current research interests include natural language processing, text mining, and document summarization.

Wenjie Li is currently an Associate Professor in Department of Computing, the Hong Kong Polytechnic University, Hong Kong. She received her PhD degree from Department of Systems Engineering and Engineering Management in the Chinese University of Hong Kong, Hong Kong, in 1997. Her main research topics include natural language processing, information extraction and document summarization.

Naishi Liu is an Associate Professor in the Department of English, School of Foreign Languages, Shanghai Jiaotong University, China. She received her PhD degree from the Department of English, Fudan University. Her current research interests include cognitive linguistics, computational linguistics, and computational humor.

Dehong Gao currently works at Alibaba.Inc. (Hangzhou, China) as a Senior Engineer. His current research interests include information retrieval and user modeling.