StalemateBreaker: A Proactive Content-Introducing Approach to Automatic Human-Computer Conversation

Xiang Li,[1,*] **Lili Mou,**[1,2] **Rui Yan,**[3] **Ming Zhang**[1]
[1]School of EECS, Peking University, China {lixiang.eecs,mzhang_cs}@pku.edu.cn
[2]Key Laboratory of High Confidence Software Technologies (Peking University),
Ministry of Education, China doublepower.mou@gmail.com
[3]Natural Language Processing Department, Baidu Inc., China yanrui02@baidu.com

Abstract

Existing open-domain human-computer conversation systems are typically *passive*: they either synthesize or retrieve a reply provided with a human-issued utterance. It is generally presumed that humans should take the role to lead the conversation and introduce new content when a stalemate occurs, and that computers only need to "respond." In this paper, we propose STALEMATEBREAKER, a conversation system that can proactively introduce new content when appropriate. We design a pipeline to determine when, what, and how to introduce new content during human-computer conversation. We further propose a novel reranking algorithm Bi-PageRank-HITS to enable rich interaction between conversation context and candidate replies. Experiments show that both the content-introducing approach and the reranking algorithm are effective. Our full STALEMATEBREAKER model outperforms a state-of-the-practice conversation system by $+14.4\%$ p@1 when a stalemate occurs.

1 Introduction

Automatic human-computer conversation is believed to be one of the most challenging problems in artificial intelligence (AI). For decades, researchers have developed various systems based on human-crafted rules [Webb, 2000; Varges *et al.*, 2009], information retrieval methods [Misu and Kawahara, 2007; Yan *et al.*, 2016], or natural language generators like neural networks [Shang *et al.*, 2015]. In these systems, the computer either searches or synthesizes a *reply* given an utterance (called *query*) issued by a user. It is generally presumed that "humans" should play a leading role in human-computer conversation. Hence traditional AI conversation is a passive process: what a computer does is just to "respond."

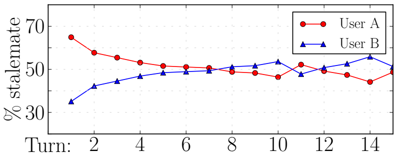


Figure 1: In a multi-turn human-human conversation started by User A, we plot the percentage at which each user should take the role of conversation leading. In other words, we detect how likely a stalemate occurs to a particular user.

In human-human conversation, however, both participants have the duty to play a leading role in a continuous dialogue session. The phenomenon is supported by the statistics of conversation data collected from an online forum[1] (Figure 1). Our observation is that, shortly after the conversation begins, both parts are likely to be the stalemate breaker. If only one side keeps finding something to talk while the other side responds in an unmindful way, the conversation becomes less attractive and is likely to end pretty soon. Therefore, in human-computer conversation, the computer side should also be initiative and introduce new content when necessary.

The problem of content introducing is also raised from industry. Although real-world conversation will end sooner or later, industrial conversation systems shall always try to attract users for commercial purposes (except when users explicitly terminate a session). Thus, stalemate breaking is of particular importance to industrial conversation products.

Existing mixed-initiative dialogue systems are typically designed in vertical domains. For example, Ferguson *et al.* [1996] develop a rule-based system, named TRAINS-95, in the transportation domain; Glas *et al.* [2015] leverage predefined topics in a museum-guiding system. Such design methodology, however, hardly applies to non-task-specific, chat-style dialogues. Since users are free to say anything, it is virtually impossible to specify rules or design templates for open-domain conversations. Moreover, the content to be introduced is nearly certain in those task- or goal-oriented applications [Seon *et al.*, 2014], whereas the nature of open-domain conversations shows that a variety of replies are plausible, but some are more meaningful, and others are not. Consequently, open-domain conversations are different from task-specific dialogues; the same thing holds for content introducing in these two scenarios.

In this paper, we propose STALEMATEBREAKER, a conversation system that can proactively introduce new content during human-computer conversations. We first detect

---

[*] Contribution partially done during internship at Baidu Inc. Corresponding author: Rui Yan (yanrui02@baidu.com).

[1] http://www.douban.com

whether a stalemate occurs by keyword filtering like "…" or "Errr," so that our system knows *when* the stalemate-breaking mechanism should be triggered. To determine *what* to introduce, we backtrack previous utterances (called *context*) in a dialogue session and apply named entity recognition in the context. The detected entities are searched for more related entities in a large knowledge graph. All these entities are used to retrieve candidate replies in a conversation database. We believe named entities highly reflect users' interest and provide informative clues for content introducing. We then propose a Bi-PageRank-HITS algorithm to address *how* to introduce. By matching the relationship between the context and candidate replies (containing the entities) in a reinforced co-ranking manner, we obtain a ranked list, indicating the relevance of each candidate reply. In this way, our system is well aware of *when*, *what*, and *how* to proactively introduce new context in a continuous human-computer conversation.

We build our proactive content-introducing system upon a large conversation database for retrieval (9.8 million candidate query-reply pairs) plus an external knowledge graph (3.7 million tuples). We evaluate our system on conversation logs from real-world users. Our approach outperforms several strong baselines as well as a state-of-the-practice system.

2 Related Work

2.1 Dialogue systems

- **Domain-specific systems.** Automatic human-computer conversation has long attracted attention in several vertical domains [Bernsen *et al.*, 1994; Rickel and Johnson, 2000]. In such task- or goal-oriented applications, researchers have developed mixed-initiative systems to more effectively guide users in conversation. Several examples include TRAINS-95 for route planning [Ferguson *et al.*, 1996], MIMIC for movie show-time information [Chu-Carroll, 2000], and AuToTutor for learner advising [Graesser *et al.*, 2005]. These systems rely heavily on human-designed rules or templates. Other systems may require intensive domain knowledge to be initiative in conversation, e.g., museum guiders [Glas *et al.*, 2015], children companion systems [Adam *et al.*, 2010; Macias-Galindo *et al.*, 2012], etc.

- **Open-domain systems.** Human-engineered rules may also be applied to the open domain as Han *et al.* [2015] do, but their generated sentences are subject to 7 predefined forms and hence are highly restricted; they leverage external knowledge bases to enhance content in the responses. Recently, more and more studies and systems are tackling the real challenges of the open domain: great flexibility and diversity. Retrieval-based methods "query" a user-issued utterance in a large database of existing dialogues, and return appropriate responses [Higashinaka *et al.*, 2014; Ji *et al.*, 2014]. Generative methods—typically using statistical machine translation techinques [Ritter *et al.*, 2011; Sugiyama *et al.*, 2013; Mairesse and Young, 2014] or neural networks [Shang *et al.*, 2015; Sordoni *et al.*, 2015]—can synthesize new replies, although the generated sentences are not guaranteed to be a legitimate natural language text. Industrial products like `Siri` of Apple, `Xiaobing` of Microsoft, and `Xiaodu` of Baidu, are among state-of-the-practice systems; they are increasingly affecting people's everyday life.

To the best of our knowledge, existing open-domain chatbot-like conversation are a *passive* process: the computer only needs to "respond" to human inputs and does not take the role of conversation leading. Instead, we propose a *proactive* system, which can determine when, what, and how to be proactive and to introduce new content into the conversation.

2.2 Random Walk-Based Ranking

Our system follows a retrieval-and-reranking schema to select replies from a candidate pool. In this part, we briefly review (re)ranking algorithms like PageRank and its variants.

In the field of information retrieval, research shows that random walks over hyper-link graphs, i.e., PageRank, can reflect the relationship between different web pages and rank high important ones [Page *et al.*, 1999]. Many studies are devoted to the application and extention of PageRank [Haveliwala, 2002; Jeh and Widom, 2003]. Random walks over bipartite graphs can model two heterogeneous types of items, e.g., the HITS algorithm for queries and documents in a click-though graph [Kleinberg, 1999; Deng *et al.*, 2009]. Its variants have been widely used for ranking tasks in the information retrieval community [Cao *et al.*, 2008; Song *et al.*, 2012]. In our scenario, the matching between user utterances and candidate replies can also be modeled as a bipartite graph; to enhance their interaction, we extend existing models and propose Bi-PageRank-HITS, which is a novel algorithm for reranking.

3 The Proposed Approach

3.1 Architecture

Figure 2 shows the overall architecture of our STALEMATE-BREAKER system; Figure 3 further illustrates the process flow of content introducing. Our system comprises mainly four steps:

Stalemate detection. The system is built upon a conventional retrieval-based conversation system, which is typically passive. The proactive content introducing starts from stalemate detection. We apply keyword matching of meaningless expressions like "…" or "Errr." In total, we have nearly a hundred filters. Although simple, the approach works in a pragmatic way and is not the main focus of this paper. In future work, we would like to apply learning-based sentence modeling (e.g., Mou *et al.* [2015]) for stalemate detection.

Named entity detection. Once the content-introducing mechanism is triggered, our system backtracks previous utterances in the current conversation session, and detects all named entities within a window. Currently, we keep four utterances, i.e., two turns, as context. We believe recently mentioned named entities highly reflect users' interest; hence we search related entities in a knowledge base for introducing new content. Concretely, the knowledge base is composed of tuples like $\langle e_1, e_2, w \rangle$, indicating the entity e_1 is correlated to e_2 with a weight of w. Notice that the relation between e_1 and e_2 in this tuple is directed because weights are asymmetric. For each entity in the current context, we search it in the knowledge base and keep top (highest weighted) five returned entities for further processing.

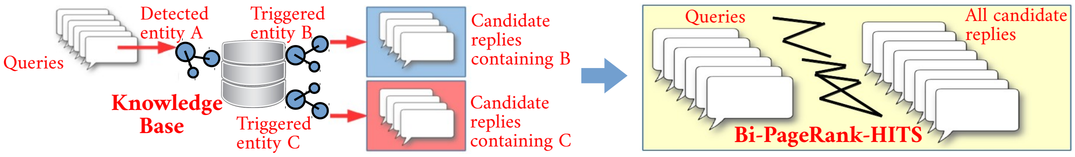


Figure 2: Overview of our STALEMATEBREAKER system.

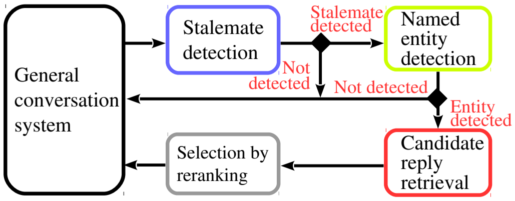


Figure 3: Process flow of triggering content introducing.

In our work, the knowledge base we used was constructed from query logs of an information retrieval system. Without loss of generality, we can also exploit similar resources, e.g., ontologies [Xu *et al.*, 2014] or information networks [Cao *et al.*, 2014]. Leveraging knowledge bases for content introducing, in fact, is also applied in Han *et al.* [2015]. However, they plug related entities to several predefined templates for response generation, whereas we have further developed complicated retrieval-and-reranking approaches.

It should be mentioned that if the conversation is not in a stalemate or we could not recognize any named entity in previous several utterances, our system will return to the general conversation mode. In other words, the proactive content-introducing method can be viewed as an "add-on" mechanism to a mature conversation system.

Candidate reply retrieval. We then use the entities and conversation context to retrieve up to fifty candidate replies from a large pool of collected dialogue data. A candidate reply contains at least one entity. This process is accomplished by standard keyword-based retrieval methods, similar to the `Lucene` system.

Selection by reranking. Finally, the candidate replies are reranked by a random walk-like algorithm. To enhance interaction between conversation utterances and candidate replies, we further propose Bi-PageRank-HITS, a novel algorithm that combines PageRank and HITS into a single framework. (See next subsection.) The highest (re)ranked candidate is selected as the reply.

3.2 Reranking Algorithm

In this part, we describe in detail the proposed Bi-PageRank-HITS algorithm, which is a combination of PageRank [Page *et al.*, 1999] and HITS [Kleinberg, 1999; Deng *et al.*, 2009].[2]

We formulate the utterances in context (called *queries*) and candidate replies as a bipartite graph. Following the notations in HITS, we denote queries as "hubs" and replies as "authorities." Then our random walk-style algorithm alternately ranks either side in the query-candidate graph (PageRank step) and interacts between the two sides (HITS step).

Our intuition is that a high hub score indicates the query is important, providing a clue for content introducing; the interaction between two sides assesses the appropriateness of a reply to all queries (reweighted by query importance). Then another PageRank over authorities suggests high-quality replies; such information is propagated back to queries in a similar way. So on and so forth, the hub and authority scores are iteratively computed in a reinforced fashion. After convergence, we obtain an overall ranking list for candidate replies. Because PageRank applies to both hubs and authorities, the model in this paper extends our previous work [Yan *et al.*, 2012b; 2015], and we name the algorithm Bi-PageRank-HITS. In the rest of this subsection, we present individual PageRank and HITS steps and then describe how they are combined.

PageRank step. For either side (e.g., queries) of the query-reply bipartite graph, we use PageRank [Page *et al.*, 1999] for scoring. We do not consider the other side (e.g., replies) in this step.

Considering the set of all queries (utterances in the conversation context), we define a random walk over an undirected graph $G_q = (V_q, E_q)$, whose nodes are the queries and edges are the relationships between a query-query pair. The weight of an edge $i \to j$ is defined to be the "similarity" between queries i and j, i.e., $M_{q,ij} = \mathrm{sim}(q_i, q_j)$, where we use the cosine measure as similarity based on two queries' *tf·idf* vectors.

To incorporate a prior distribution $\mathbf{x}$ over queries, we follow Yan *et al.* [2012a] and define the PageRank formula as

$$\mathbf{q}^{(i+1)} = (1-\mu)[\mathrm{Diag}(\mathbf{x})\mathrm{M}_q^\top]\mathbf{q}^{(i)} + \mu\mathbf{x} \tag{1}$$

Likewise, for the set of candidate replies, we have

$$\mathbf{r}^{(i+1)} = (1-\mu)[\mathrm{Diag}(\mathbf{y})\mathrm{M}_r^\top]\mathbf{r}^{(i)} + \mu\mathbf{y} \tag{2}$$

where $[\cdot]$ denotes column normalization, that is to say, each column is the transition probability of its corresponding node. Superscripts indicate the number of iterations in a particular PageRank step (called *local iteration*).

PageRank is inspired by the following intuition. A candidate reply is important if it is "voted" by many other candidates. The prior ($\mathbf{x}$ for queries or $\mathbf{y}$ for replies) is initialized as a uniform distribution, but may change to emphasize particular nodes suggested by the HITS step during interaction between queries and replies.

HITS step. After the above step, we obtain PageRank scores for either queries or candidate replies. To propagate such information to the other side in the query-reply bipartite graph, we perform another random walk with links between

---

[2] "HITS" is the acronym of *hyperlink-induced topic search*.

queries and replies representing the structural information of hubs and authorities.

Formally, the bipartite graph $G = (V, E)$ has vertexes $V = \{V_q \cup V_r\}$, where V_q are queries and V_r are replies. We define the weight matrix by a relevance scoring function $\phi(q, r)$ between queries and replies. $\phi(\cdot, \cdot)$ was learned via a *learning-to-rank* model similar to Burges *et al.* [2005] with rich features including textual similarity, translation models, as well as `word2vec` word embeddings [Mikolov *et al.*, 2013]. In other words, $\phi(\cdot, \cdot)$ returns the relatedness between a query and a reply in the range $(0, 1)$.

Moreover, since we would like to make use of PageRank scores for queries or replies, the HITS links are judged by not only the static relevance score $\phi(\cdot, \cdot)$, but also the information given by PageRank. To be concrete, the (unnormalized) weight matrix is given by either

$$\tilde{\mathbf{W}}_{ij} = \phi(q_i, r_j) \cdot q_i \qquad \text{(query}\rightarrow\text{reply)} \qquad (3)$$

$$\text{or} \qquad \tilde{\mathbf{W}}_{ij} = \phi(r_i, q_j) \cdot r_i \qquad \text{(reply}\rightarrow\text{query)} \qquad (4)$$

Here, q_i and r_i are the *i*-th element in the vectors $\mathbf{q}$ and $\mathbf{r}$, which are obtained in the PageRank phase by Equations 1–2. If information is propagated from queries to replies, we use the former equation for weight update, and *vice versa*.

The mutual-reinforcing relationship of hub and authority scores can be expressed in matrix representation as follows.

$$\mathbf{x}^{(i+1)} = \alpha_x \cdot \left[\tilde{\mathbf{W}}\right] \mathbf{y}^{(i)} + (1 - \alpha_x) \cdot \hat{\mathbf{x}} \qquad (5)$$

$$\mathbf{y}^{(i+1)} = \alpha_y \cdot \left[\left[\tilde{\mathbf{W}}\right]^\top\right] \mathbf{x}^{(i)} + (1 - \alpha_y) \cdot \hat{\mathbf{y}} \qquad (6)$$

where $\mathbf{x}$ is query scores, $\mathbf{y}$ reply scores; superscripts denote the number of local iterations in HITS update. Each column of the weights is normalized to be a valid probability. Moreover, to compute the weight for $\mathbf{y}$, the matrix $\tilde{\mathbf{W}}^\top$ should be first row-normalized (given by $[\tilde{\mathbf{W}}]^\top$); otherwise, the effect of the prior q_i in Equation 3 is ruled out by column normalization. Notice that the transition matrix is fixed in one step of HITS, but they may also change as our algorithm proceeds like the PageRank step.

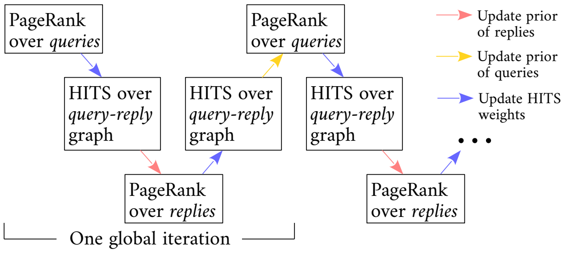
In the above equations, the first term is the standard HITS, which is entirely determined by the linkage structure between hubs and authorities. The second term indicates that, with a certain probability, the hub and authority scores will be influenced by their prior scores $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$. Such idea of combining additional information of hubs and authorities is proposed in Deng *et al.* [2009] and called Co-HITS.[3]

We define the prior score of a query to be proportional to the averaged relevance (textual similarity) score to all replies, i.e., $\hat{x}_i \propto \frac{1}{\#r'} \sum_{r'} \mathrm{sim}(q_i, r')$. Likewise, $\hat{y}_i \propto \frac{1}{\#q'} \sum_{q'} \mathrm{sim}(q', r_i)$. The vectors $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ are also self-normalized so as to be valid probabilities.

Iteration over PageRank and HITS. After performing PageRank over one side (e.g., queries) of the bipartite query-reply graph and propagating the information with HITS, we shall perform another PageRank on the other side (e.g., replies). As mentioned, we use the results $\mathbf{x}$ or $\mathbf{y}$ obtained by HITS as the prior information, and recompute the transition weights in PageRank (Equation 1 or 2). Then, the information is propagated back by HITS, and PageRank is applied again for a better estimation.

Algorithm 1: Bi-PageRank-HITS

```
Input: Queries (utterances) and candidate replies
Output: Ranking list of replies for content introducing
begin
    //Global iteration in Bi-PageRank-HITS
    repeat
        Update query priors
        repeat
            PageRank update over queries by Eqn. 1
        until Local convergent;
        Update HITS weights by Eqn. 3 (query→reply)
        repeat
            HITS update over query-reply bipartite graph
        until Local convergent;
        Update reply priors
        repeat
            PageRank update over replies by Eqn. 2
        until Local convergent;
        Update HITS weights by Eqn. 4 (reply→query)
        repeat
            HITS update over query-reply bipartite graph
        until Local convergent;
    until Global convergent;
```



As depicted in Algorithm 1, our Bi-PageRank-HITS algorithm performs PageRank and HITS steps alternately and computes the importance of a query/reply in a reinforced way. Note that the transition matrix in each step of PageRank or HITS is fixed, but they change dynamically during global iterations. Thus our model is different from the original PageRank or HITS.

For convergence concerns, we have normalized all prior distributions and all columns in transition matrices. Therefore, the convergence of PageRank and HITS is guaranteed. For alternation between the two steps, we shall empirically analyze its convergence in Section 4.4. In practice, we terminate our algorithm when the mean square difference between two successive HITS scores in global iteration is less than a threshold (10^{-6} in our study).

After convergence, we use replies' HITS scores $\mathbf{y}$ for final reply selection. Compared with PageRank scores $\mathbf{r}$, HITS scores convey more structural information between queries and replies. A higher score indicates the reply is more appropriate for content introducing.

---

[3] Precisely, our model uses the Co-HITS variant. But for simplicity, we denote it as HITS, if not confused, for notational purposes.

4 Evaluation

4.1 Datasets and Experimental Setups

Our content-introducing open-domain conversation system is built upon a large database of conversation data for retrieval. We collected massive resources from (Chinese) forums, microblog websites, and community question-answering platforms including Baidu Zhidao, Baidu Tieba, Douban forum, Sina Weibo, etc.[4] In total, we extracted nearly 10 million query-reply pairs. Besides, we leveraged a knowledge graph mined from Baidu search logs.

To evaluate the proposed STALEMATEBREAKER, we resorted to human evaluation, following Ritter *et al.* [2011] and Shang *et al.* [2015]. Objective scores like BLEU and traditional evaluation for dialogue systems (e.g., accuracy of template classification) are less applicable to our scenario, because open-domain conversation is highly diverse—one query can have a lot of suitable replies that appear different to each other. Human evaluation, on the other hand, conforms to the ultimate goal of open-domain conversation systems. In our experiments, we used 180 sessions from real-world user conversation logs. For each entity in the context (4 previous utterances in the session), we sought 5 most related entities in the knowledge graph, and for each related entity, we retrieved top-10 candidate replies (containing the entity).

We hired workers on a Chinese crowdsourcing platform to annotate all retrieved results with **1 Point** (appropriate) or **0 Point** (inappropriate). A candidate reply was annotated by 3 workers in an independent and blind fashion. We regarded the majority voting as the "ground truth" indicating whether the reply is appropriate for content introducing. We also evaluated the kappa score: $\kappa = 0.768$, showing high inner-annotator agreement [Fleiss, 1971].

Our website[5] provides additional data statistics and rating criterion.

4.2 Competing Methods

We compared the proposed proactive content-introducing method with passive conversation systems. As our work is an "add-on" mechanism to a deployed system, we presumed that a stalemate had been detected in our evaluation. In other cases, our proposed method does not corrupt the existing system.

For fairness, our baselines (passive conversation systems) were also aware of context information for candidate reply retrieval, since an utterance like "Errr" itself contains little substance. We also performed data cleaning like Pitler *et al.* [2010] by removing candidates of low linguistic quality such as extremely short ones or meaningless babblings.

Regarding ranking algorithms, we compared the proposed Bi-PageRank-HITS with the following methods:

- **Textual similarity.** This method ranks candidate replies according to textual similarity, predicted by a regression model with human engineered features. It is a state-of-the-practice system[6] which our experimental environment was built upon.

Group	Reranking Method	p@1	MAP	nDCG
No content introducing	Textural similarity†	0.406	0.498	0.648
	HITS	0.467	0.550	0.684
	Reply PageRank	0.428	0.514	0.660
	Co-HITS	0.472	0.552	0.686
	Bi-PageRank-HITS	*0.483*	**0.556**	**0.690**
Entity-based content introducing	Textural similarity	0.511	0.551	0.742
	HITS	0.494	0.542	0.733
	Reply PageRank	0.467	0.436	0.660
	Co-HITS	0.511	0.555	0.743
	Bi-PageRank-HITS‡	**0.550**	**0.562**	**0.750**

Table 1: Performance of our method and baseline systems. †A state-of-the-practice system which our model is built upon. ‡The full STALEMATEBREAKER system. Notice that the MAP and nDCG metrics are not comparable outside a group because the retrieved candidates are different.

- **Reply PageRank.** PageRank is a widely used ranking algorithm [Page *et al.*, 1999], and is, actually, a component of the Bi-PageRank-HITS model.
- **HITS.** HITS is a link analysis algorithm suitable for modeling bipartite graphs like web click-through data [Kleinberg, 1999].
- **Co-HITS.** Co-HITS [Deng *et al.*, 2009], a variant of HITS, is another component of our Bi-PageRank-HITS model. Details are described in Section 3.2.

4.3 Overall Performance

We first evaluated the performance using the p@1 metric, which is believed to be the most direct judgment of conversation systems [Wang *et al.*, 2013; Shang *et al.*, 2015]. It reflects exactly the "accuracy" of the highest-ranked reply. Further, we applied mean average precision (MAP) and normalized discounted cumulative gain (nDCG), as both our system and baselines return a ranking list containing multiple candidates. For details of our metrics, we refer interested readers to Järvelin and Kekäläinen [2002] and Kishida [2005]. Formulas are also listed on our website.

Table 1 shows the performance of our STALEMATEBREAKER system as well as a variety of baselines.

The main result is that proactive entity-based content introducing is generally better (higher p@1) than passive conversation systems regardless of the ranking algorithm. Although passive systems can reply to important previous utterances in the current conversation session to some extent (because the baselines are also context-aware), they are more likely to repeat existing topics and be stuck in the stalemate.

By contrast, our STALEMATEBREAKER proactively seees a knowledge base, retrieves candidate replies that contain related entities, and reranks more appropriate ones. Thus, entity-based content introducing methods yield higher scores in terms of p@1, i.e., accuracy of the top-ranked reply.

Regarding the Bi-PageRank-HITS ranking algorithm, it outperforms its base models, PageRank and HITS/Co-HITS, as well as a feature-rich regression model based on textural similarity. The results are conservative in all metrics (p@1, MAP, and nDCG) and in both introducing and non-introducing groups, showing that our ranking algorithm is potentially applicable to other tasks.

---

[4] http://{zhidao.baidu|tieba.baidu|douban|weibo}.com

[5] https://sites.google.com/site/stalematebreaker/

[6] http://duer.baidu.com

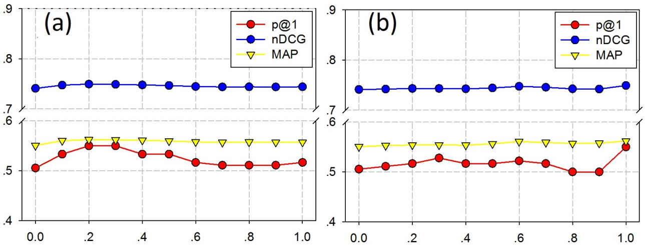


Figure 4: Parameter analysis. (a) α_x and (b) α_y were tuned from 0 to 1 with a granulairty of 0.1 given the other parameter was fixed (at the stationary point of grid search in the parameter space).

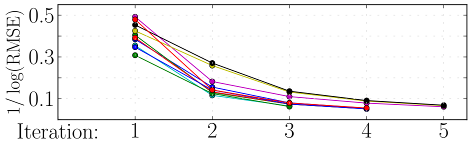


Figure 5: Convergence analysis of global iterations of Bi-PageRank-HITS. 10 randomly chosen samples are plotted.

To sum up, the experimental results show that both our entity-based content introducing and Bi-PageRank-HITS are effective. When a stalemate occurs, the full STALEMATE-BREAKER yields a $+14.4\%$ boost of accuracy (p@1) compared with a state-of-the-practice system which our model is built upon.

4.4 Analysis and Discussion

Parameter Settings. In our Bi-PageRank-HITS model, we have three main parameters, μ in the PageRank phase, and α_x, α_y in the HITS phase. μ was set to 0.15 following Yan *et al.* [2012a] and not tuned in our experiment. For α_x and α_y, we tried different values with a granularity of 0.1. The results are shown in Figure 4. If α is set to 0, the HITS update vanishes (the first term in Equations 5–6), and the system solely depends on "prior" information. The result is worse than using HITS update.

When α increases, we observe an interesting phenomenon: queries and replies respond differently. For queries, i.e., context utterances, the performance peaks when α_x is small (Figure 4a). This suggests that textual information $\hat{\mathbf{x}}$ (based on query-reply similarity) does recommend important queries. On the contrary, textual information $\hat{\mathbf{y}}$ for replies is inimical: the highest performance was achieved when $\alpha_y = 1$, where $\hat{\mathbf{y}}$ was discarded. One plausible explanation is that, based on textual similarity, the algorithm is likely to recommend similar replies to the queries and be stuck in stalemate. In summary, the parameters we used were $\alpha_x = 0.3$ and $\alpha_y = 1$.

Convergence. In our Bi-PageRank-HITS, the convergence of each local step (either PageRank or HITS) is guaranteed because we have properly normalized the transition matrices. We further empirically analyze the global convergence by plotting the root mean square error between two successive HITS scores. Figure 5 shows the curves of 10 randomly chosen samples; the results indicate that Bi-PageRank-HITS typically converges quickly in 3–5 global iterations. It works well in practice.

Human:	以后叫你伊娃。(I will call you Eva.)
Computer:	伊娃不是那个机器人动画片？ (Isn't Eva a robot cartoon?)
Human:	你会聊天吗？ (Can you talk?)
Computer:	不是聊的好好的吗？ (Aren't we talking now?)
Human:	啊. . . (Errr. . .)
Computer:	我不是瓦力，我没有伊娃(I'm not WALL-E and I don't have Eva.)
Remark: WALL-E and Eva are two robots in the movie WALL-E.	
Human:	林依晨(Yichen Lin.)
Computer:	北川景子(Kitagawa Keiko.)
Human:	我看电影去了(I will go to watch TV.)
Computer:	不如看电影(Better go to a movie.)
Human:	啊. . . (Errr. . .)
Computer:	我比较喜欢胡哥和林依晨的天外飞仙 (I like "Marriage of the Fairy Princess" casted by Ge Hu and Yichen Lin)
Remark: "Marriage of the Fairy Princess" is a TV play; Kitagawa Keiko, Yichen Lin, and Ge Hu are three actors/actresses.	

Table 2: Examples obtained by STALEMATEBREAKER.

Case Study. We illustrate two examples in Table 2 obtained by Bi-PageRank-HITS. As seen, the (human) user doen not know waht to say by "Err. . . ," but our system can introduce new content (entities like "WALL-E" and the TW play "Marriage of the Fairy Princess") to attract users.

5 Conclusion

In this paper, we addressed the problem of content introducing for stalemate breaking in open-domain conversation systems. We proposed a pipeline of content introducing based on an external knowledge graph. To enhance the interaction between queries (utterances in the conversation) and candidate replies (retrieved from a massive database), we further proposed the random walk-style Bi-PageRank-HITS reranking approach. Experiments show the effectiveness of both our content-introducing method and the ranking algorithm.

Acknowledgments

This paper is partially supported by the National Natural Science Foundation of China (Grant Nos. 61272343 and 61472006), the Doctoral Program of Higher Education of China (Grant No. 20130001110032), and the National Basic Research Program (973 Program No. 2014CB340405 and No. 2014CB340505).

References

[Adam *et al.*, 2010] C. Adam, L. Cavedon, and L. Padgham. Flexible conversation management in an engaging virtual character. In *Workshop on Interacting with ECAs as Virtual Characters*, 2010.

[Bernsen *et al.*, 1994] N. Bernsen, L. Dybkjær, and H. Dybkjær. A dedicated task-oriented dialogue theory in support of spoken language dialogue systems design. In *ICSLP*, 1994.

[Burges *et al.*, 2005] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *ICML*, pages 89–96, 2005.

[Cao *et al.*, 2008] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li. Context-aware query suggestion by mining click-through and session data. In *SIGKDD*, pages 875–883, 2008.

[Cao *et al.*, 2014] B. Cao, X. Kong, and P. Yu. Collective prediction of multiple types of links in heterogeneous information networks. In *ICDM*, pages 50–59, 2014.

[Chu-Carroll, 2000] J. Chu-Carroll. MIMIC: An adaptive mixed initiative spoken dialogue system for information queries. In *ANLP*, pages 97–104, 2000.

[Deng *et al.*, 2009] H. Deng, M. Lyu, and I. King. A generalized co-hits algorithm and its application to bipartite graphs. In *SIGKDD*, pages 239–248, 2009.

[Ferguson *et al.*, 1996] G. Ferguson, J. Allen, B. Miller, et al. Trains-95: Towards a mixed-initiative planning assistant. In *AIPS*, pages 70–77, 1996.

[Fleiss, 1971] J. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378, 1971.

[Glas *et al.*, 2015] N. Glas, K. Prepin, and C. Pelachaud. Engagement driven topic selection for an information-giving agent. In *Workshop on the Semantics and Pragmatics of Dialogue*, 2015.

[Graesser *et al.*, 2005] A. Graesser, P. Chipman, B. Haynes, and A. Olney. Autotutor: An intelligent tutoring system with mixed-initiative dialogue. *IEEE Trans. Education*, 48(4):612–618, 2005.

[Han *et al.*, 2015] S. Han, J. Bang, S. Ryu, and G. Lee. Exploiting knowledge base to generate responses for natural language dialog listening agents. In *SIGDIAL*, pages 129–133, 2015.

[Haveliwala, 2002] T. Haveliwala. Topic-sensitive PageRank. In *WWW*, pages 517–526, 2002.

[Higashinaka *et al.*, 2014] R. Higashinaka, N. Kobayashi, T. Hirano, C. Miyazaki, T. Meguro, T. Makino, and Y. Matsuo. Syntactic filtering and content-based retrieval of Twitter sentences for the generation of system utterances in dialogue systems. In *IWSDS*, pages 113–123, 2014.

[Järvelin and Kekäläinen, 2002] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *TOIS*, 20(4):422–446, 2002.

[Jeh and Widom, 2003] G. Jeh and J. Widom. Scaling personalized web search. In *WWW*, pages 271–279, 2003.

[Ji *et al.*, 2014] Z. Ji, Z. Lu, and H. Li. An information retrieval approach to short text conversation. *arXiv preprint arXiv:1408.6988*, 2014.

[Kishida, 2005] K. Kishida. *Property of Average Precision and its Generalization: An Examination of Evaluation Indicator for Information Retrieval Experiments*. 2005.

[Kleinberg, 1999] J. Kleinberg. Authoritative sources in a hyperlinked environment. *JACM*, 46(5):604–632, 1999.

[Macias-Galindo *et al.*, 2012] D. Macias-Galindo, W. Wong, J. Thangarajah, and L. Cavedon. Coherent topic transition in a conversational agent. In *INTERSPEECH*, pages 743–746, 2012.

[Mairesse and Young, 2014] F. Mairesse and S. Young. Stochastic language generation in dialogue using factored language models. *Computational Linguistics*, 40(4):763–799, 2014.

[Mikolov *et al.*, 2013] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[Misu and Kawahara, 2007] T. Misu and T. Kawahara. Speech-based interactive information guidance system using question-answering technique. In *ICASSP*, pages I45–148, 2007.

[Mou *et al.*, 2015] L. Mou, H. Peng, G. Li, Y. Xu, L. Zhang, and Z. Jin. Discriminative neural sentence modeling by tree-based convolution. In *EMNLP*, pages 2315–2325, 2015.

[Page *et al.*, 1999] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the Web. http://ilpubs.stanford.edu:8090/422/, 1999.

[Pitler *et al.*, 2010] E. Pitler, A. Louis, and A. Nenkova. Automatic evaluation of linguistic quality in multi-document summarization. In *ACL*, pages 544–554, 2010.

[Rickel and Johnson, 2000] J. Rickel and W. Johnson. Task-oriented collaboration with embodied agents in virtual worlds. *Embodied Conversational Agents*, pages 95–122, 2000.

[Ritter *et al.*, 2011] A. Ritter, C. Cherry, and W. Dolan. Data-driven response generation in social media. In *EMNLP*, 2011.

[Seon *et al.*, 2014] C. Seon, H. Lee, H. Kim, and J. Seo. Improving domain action classification in goal-oriented dialogues using a mutual retraining method. *PRL*, 45:154–160, 2014.

[Shang *et al.*, 2015] L. Shang, Z. Lu, and H. Li. Neural responding machine for short-text conversation. In *ACL*, 2015.

[Song *et al.*, 2012] Y. Song, D. Zhou, and L. He. Query suggestion by constructing term-transition graphs. In *WSDM*, 2012.

[Sordoni *et al.*, 2015] A. Sordoni, M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J. Nie, J. Gao, and B. Dolan. A neural network approach to context-sensitive generation of conversational responses. In *NAACL-HLT*, pages 196–205, 2015.

[Sugiyama *et al.*, 2013] H. Sugiyama, T. Meguro, R. Higashinaka, and Y. Minami. Open-domain utterance generation for conversational dialogue systems using web-scale dependency structures. In *SIGDIAL*, pages 334–338, 2013.

[Varges *et al.*, 2009] S. Varges, S. Quarteroni, G. Riccardi, A. Ivanov, and P. Roberti. Leveraging pomdps trained with user simulations and rule-based dialogue management in a spoken dialogue system. In *SIGDIAL*, pages 156–159, 2009.

[Wang *et al.*, 2013] H. Wang, Z. Lu, H. Li, and E. Chen. A dataset for research on short-text conversations. In *EMNLP*, 2013.

[Webb, 2000] N. Webb. Rule-based dialogue management systems. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.22.2854, 2000.

[Xu *et al.*, 2014] Y. Xu, G. Li, L. Mou, and Y. Lu. Learning non-taxonomic relations on demand for ontology extension. *IJSEKE*, 24(08):1159–1175, 2014.

[Yan *et al.*, 2012a] R. Yan, M. Lapata, and X. Li. Tweet recommendation with graph co-ranking. In *ACL*, pages 516–525, 2012.

[Yan *et al.*, 2012b] R. Yan, X. Wan, M. Lapata, W. Zhao, P. Cheng, and X. Li. Visualizing timelines: Evolutionary summarization via iterative reinforcement between text and image streams. In *CIKM*, pages 275–284, 2012.

[Yan *et al.*, 2015] R. Yan, Y. Song, C. Li, M. Zhang, and X. Hu. Opportunities or risks to reduce labor in crowdsourcing translation? Characterizing cost versus quality via a pagerank-HITS hybrid model. In *IJCAI*, pages 1025–1032, 2015.

[Yan *et al.*, 2016] R. Yan, Y. Song, and H. Wu. Learning to respond with deep neural networks for retrieval based human-computer conversation system. In *SIGIR*, 2016.