MEDINFO 2017: Precision Healthcare through Informatics
A.V. Gundlapalli et al. (Eds.)

doi:10.3233/978-1-61499-830-3-207

PubMedReco: A Real-Time Recommender System for PubMed Citations

Hamman W. Samuel, Osmar R. Zaïane

Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada

Abstract

We present a recommender system, PubMedReco, for real-time suggestions of medical articles from PubMed, a database of over 23 million medical citations. PubMedReco can recommend medical article citations while users are conversing in a synchronous communication environment such as a chat room. Normally, users would have to leave their chat interface to open a new web browser window, and formulate an appropriate search query to retrieve relevant results. PubMedReco automatically generates the search query and shows relevant citations within the same integrated user interface. PubMedReco analyzes relevant keywords associated with the conversation and uses them to search for relevant citations using the PubMed E-utilities programming interface. Our contributions include improvements to the user experience for searching PubMed from within health forums and chat rooms, and a machine learning model for identifying relevant keywords. We demonstrate the feasibility of PubMedReco using BMJ's Doc2Doc forum discussions.

Keywords:

PubMed; Medical Informatics Applications; Information Storage and Retrieval

Introduction

PubMed is the de facto tool for searching biomedical and life sciences literature. PubMed is comprised of the MEDLINE (Medical Literature Analysis and Retrieval System Online) bibliographic database, which covers academic journals on medicine, pharmacy, nursing, dentistry, and medical care, among others, with over 23 million trustworthy and peer-reviewed articles[1]. Essentially, PubMed is the web interface to the MEDLINE database. PubMed is typically used from its home page of the National Center for Biotechnology Information (NCBI) website. Based on the keywords entered in the search engine, matching citations are returned, which can be further explored by clicking on each citation to view abstracts or full-text.

The MEDLINE database is indexed using the Medical Subject Headings (MeSH), an indexing medical vocabulary[2]. To facilitate natural language usage in the PubMed search engine, PubMed uses the process of Automatic Term Mapping, which matches non-MeSH keywords to the MeSH index. A MeSH translation table is used for each query issued by the user to map the natural language keywords to equivalent MeSH keywords. In order to broaden search results, PubMed also leverages mappings of the search query keywords derived from the Unifed Medical Language System (UMLS).

UMLS is a collection of biomedical vocabularies and standards, and PubMed can leverage these vocabularies to expand search queries using hypernymns, hyponyms, synonyms, and other semantic relationships[3]. The interactions between PubMed, MeSH, UMLS, and MEDLINE are depicted in Figure 1.

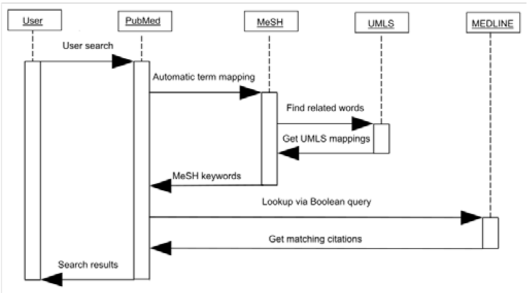


Figure 1 – Sequence diagram depicting interactions between PubMed, MeSH, UMLS and MEDLINE during search

However, using the PubMed web interface for searching is not ideal when users are in a contained environment such as a chat room or a forum discussion thread. Users would have to leave their current web page in order to go to the PubMed website and retrieve the information they need to look up. Furthermore, users need to have a sense of the context of the overall conversation and the key topics being discussed.

Our proposed recommender system integrates PubMed citations into a unified user experience so that medical citations relevant to the on-going conversation are conveniently accessible for the users. From our survey of existing literature, our proposed system is the first of its kind in the medical domain. It should be noted that access to reading an article depends on the individual user's subscription to PubMed; our system displays only the citation, and a clickable hyperlink to the article conveniently within a chat environment interface.

A recommender system is a set of software tools and algorithms that can give useful suggestions to users[4]. The suggestions are given within the context of the user's domain of interest such as what items to buy or shop for, which new people to connect with, or which new movies to watch. Recommender systems are useful from both the perspectives of the content producers (such as sellers, blog writers, videographers, movie makers, and so on) and the content consumers (such as buyers, readers, fans, etc.). Recommendations enable sifting through large amounts of information previously too massive or complicated to practically navigate. Recommendations also enable users to focus on things that interest them personally, thereby increasing the findability of information.

There are various methods for generating recommendations: content-based, collaborative, community-based, demographic, knowledge-based[4]. Content-based recommendations use keywords to suggest new items that are historically similar to previous items that a user may have liked or bought in the past. In collaborative recommender systems, suggestions for one user are based on what other users with similar profiles have liked. Community-based recommendations leverage the preferences of users' friends, a popular notion in Social Networking Sites (SNS). The demographic approach uses age, gender, ethnicity, and other demographic information about a user, as well as the items they like or buy. These properties are matched to demographic stereotypes such as English-speaking customers being directed to US-based websites. The knowledge-based approach is to recommend items using specific domain knowledge about how certain item features meet user needs and preferences.

Recommendation systems in chat rooms have been researched in other domains. PALTask is a personalized automated context-aware web resources listing tool that suggests resources of common interests to users in a chat room[5]. These resources include videos, web articles, and social media links relevant to the conversation. PALTask performs contextual analysis, and determines the context of the conversation based on each conversing user's profile, called Personal Contextual Sphere (PCS), and also from the chat conversation contents. The PCS is determined when the user mentions specific keywords related to their personal interests. As an example, context analysis can determine that a user mentioned a music video in the chat. PALTask can then display links to that music video within the chat window interface.

Another real-time chat recommender system has been proposed for e-commerce websites[6]. The recommender system is based on profiling users while they chat. A user may be interacting with another user to chat about buying things, or may be chatting with a seller. The profile contains five parameters: “unusual", “cute", “cool", “simple", and “luxurious", with each parameter having a value 1-5, with 5 being the strongest value. The recommender updates this profile by analyzing chat conversations based on inferring positive and negative connotations of words in the chat with the parameters, and using words with associated positive feedback to query the product catalog and recommend items.

Another research work of interest looks at the role of temporal dynamics in product recommendations[7]. The research focuses on evolving user preferences over time, arguing that traditional methods do not take into account the temporal changes of user preferences. For example, if a user bought a particular baby food from an e-commerce website, it is very likely that they would not be interested in seeing recommendations related to their purchase after a while because the dietary needs of a child would be changing as the child grows. To recommend new items, collaborative filtering-style similarity metrics are used based on transactional history. By using these metrics in cluster-based and graph-based modeling, the system retrieves similar items to recommend.

Stream and event processing systems have also been explored regarding real-time recommendations[8,9]. Streams are sequences of events, and stream systems require high throughput processing, making them ideal for the real-time recommendation approach.

Social news is an example of stream systems, where new stories are continuously being added, and user ratings can instantly affect recommendations for related interesting news. The recommendation engine, StreamRec, uses collaborative filtering and works in two phases: model building and recommendation generation. In the model building phase, a similarity score is computed for user-rating pairs. In the recommendation generation phase, items are given a predicted rating based on the similarity scores, and items with a high predicted rating are suggested.

In both PALTask and the e-commerce recommender, incorporation of temporal context has not been considered. For instance, if keywords are extracted from the entire conversation, some keywords may be outdated because they were discussed hours ago. Moreover, even if the latest chat conversation is being used, it might not give the best recommendations. The latest message might diverge significantly from the overall conversation, and could be an outlier. For the StreamRec engine and the products recommender, reliance on a history of ratings could potentially result in the cold start problem, whereby the initial set of ratings have to be determined arbitrarily in the absence of a history of ratings[10]. Another issue with using historical ratings for determining recommendations is sparsity of data, whereby only a few users might choose to give feedback or ratings.

It should also be noted that PubMedReco is a content-based recommender system for temporal conversations, and not a Time-Aware Recommender System (TARS)[11]. The focus of TARS is on temporal relationships between recommendations, while PubMedReco looks at temporal and contextual links between the keywords needed to provide recommendations.

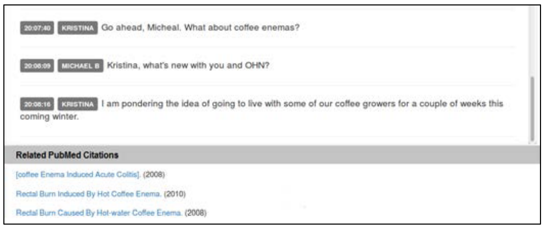
Methods

To demonstrate the feasibility of PubMedReco, we develop a prototype that is populated with conversations taken from a health forum, thereby simulating a chat-like environment where new messages are being added over time. For evaluation, we look at three aspects of accuracy: selection of relevant keywords, agreement about what constitutes keyword relevance, and correlation between recommended citations and selected keywords.

An overview of the prototype is shown in Figure 2, where users are chatting on the topic of coffee enema. For each new message, medical keywords are extracted. A subset of all the keywords extracted are then used to query PubMed and get related citations. In this way, users can view citations related to the overall conversation without needing to go to the PubMed website, or having to determine which specific keywords to use for querying PubMed.

System Design

The PubMedReco system is outlined in Figure 3. As a new message arrives, it is tokenized via regular expressions into individual keywords. Next, common English stopwords are discarded, and the remaining keywords are retained along with relevant information such as the timestamp of the message containing the keyword. In the next step, the incoming keywords from the new message are looked up in an index of all retrieved keywords, which is initially empty. If the incoming keyword exists, only its related information is updated, including the latest message number containing the keyword. Otherwise, the new keyword is added to the index.



20:07:40 KRISTINA Go ahead, Micheal. What about coffee enemas?

20:08:09 MICHAEL B Kristina, what's new with you and OHN?

20:08:16 KRISTINA I am pondering the idea of going to live with some of our coffee growers for a couple of weeks this coming winter.

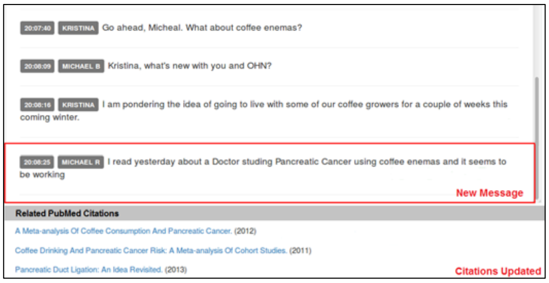
Related PubMed Citations

[coffee Enema Induced Acute Colitis]. (2008)

Rectal Burn Induced By Hot Coffee Enema. (2010)

Rectal Burn Caused By Hot-water Coffee Enema. (2008)

(a) Chat interface at time t_i



20:07:40 KRISTINA Go ahead, Micheal. What about coffee enemas?

20:08:09 MICHAEL B Kristina, what's new with you and OHN?

20:08:16 KRISTINA I am pondering the idea of going to live with some of our coffee growers for a couple of weeks this coming winter.

20:08:25 MICHAEL R I read yesterday about a Doctor studing Pancreatic Cancer using coffee enemas and it seems to be working

Related PubMed Citations

A Meta-analysis Of Coffee Consumption And Pancreatic Cancer. (2012)

Coffee Drinking And Pancreatic Cancer Risk: A Meta-analysis Of Cohort Studies. (2011)

Pancreatic Duct Ligation: An Idea Revisited. (2013)

(b) Chat interface at time t_{i+1}

Figure 2 – PubMedReco Prototype

```
keywordsIndex = NULL
FOREACH newMessage IN allMessages
      keywords = RegExTokenizer(newMessage)
      keywords = StopWordsFilter(keywords)
      FOREACH keyword IN keywords
            IF keyword IN keywordsIndex
                  keywordsIndex.update(keyword)
            ELSE
                  keywordsIndex.add(keyword)
      relevantKeywords = NeuralNet(keywordsIndex)
      citations = EUtilities(relevantKeywords)
      RETURN citations
```

Figure 3 – PubMedReco System Design

The keywords index is then fed to an artificial neural network that selects keywords that are both temporally and contextually relevant. Artificial neural networks or neural nets, are computational models that are based on the structure of the biological brain, where a large collection of neurons and axons can enforce or inhibit signals, and in turn activate different states[12]. Neural nets have been used for modeling binary classification problems, where elements of a given set need to be categorized into two groups based on specific rules. We model the task of determining relevant keywords by inputting various features for each keyword and then outputting 0 if the keyword is irrelevant, or 1 if it is relevant. The keyword features are listed in Table 1.

Neural Net Design

The keyword features are converted to binary form for input into the artificial neural net's neurons, as neuron states are more readily represented as 0 or 1.

This conversion includes the keywords themselves, which are converted to word embeddings using the Word2Vec deep neural network model trained with skip-grams on the entire Doc2Doc dataset[16]. The word embeddings enable each unique keyword to be assigned a corresponding binary vector in the space. Also, keywords with common contexts such as synonyms are positioned close each other in the vector space, and have similar binary vector values.

Table 1 – Neural Net Keyword Features

Property	Description
keyword	Word embeddings representation of keyword
isMed1	Is keyword in the Merriam-Webster's Medical Dictionary[1]
isMed2	Is keyword in SNOMED database[2]
firstPos	Message number/reference where keyword first appears
lastPos	Message number/reference where keyword last appeared
firstTime	Seconds passed since epoch till first occurrence of keyword
lastTime	Seconds passed since epoch till last occurrence of keyword
frequency	Number of times the keyword has appeared in the chat
numMsgs	Number of messages in the entire conversation

Training Dataset

Initially, the neural net needs to learn how to associate these features to the groupings by using a training dataset with keywords already classified as relevant or irrelevant. The neural net model is trained by manual annotation of a subset of BMJ's Doc2Doc forum discussions dataset[3]. The annotation process involves manually inspecting the keywords for each new incoming message, and marking their relevancy based on the current context. The Doc2Doc forums allow doctors to have online discussions with other doctors on various health-related topics. The temporal nature of forum converations is ideal for testing PubMedReco, as forum discussions progress over time like online chats. Also, the technical nature of doctors' conversations makes the dataset suitable for querying PubMed. It should be noted that a forum discussion or chat containing n messages yields n training sets because annotations are made for each new incoming message. The trained model can then be used to predict the relevance of new forum discussions or chats that do not have any manual annotations.

Citations Retrieval

Once the relevant keywords are selected, they are then used to query the PubMed database programmatically using Entrez Programming Utilities (E-utilities), a RESTful programming interface[3]. E-utilities accept natural language queries and converts them into Boolean queries by inserting Boolean operators and using the words as operands. Stemming and lemmatization are also performed on keywords by the E-utilities API which can infer synonyms and other relationships to the query words via UMLS.

E-utilities has options for specifying what citation fields to search within, such as title, abstract, full text (where available), author and others. Our proposed system restricts search to the citation title because our recommendations are ultimately presented as full citation titles. Consequently, the user would decide initial interest or disinterest in the recommendation based on the displayed title. The citations returned can also be sorted using various options available in E-utilities, and we sort by relevancy, which takes into account the frequency of matched keywords within the title. Hence, citations containing more of the search keywords would be ranked higher.

Evaluation Criteria

As mentioned before, three aspects of PubMedReco need to be evaluated for accuracy: neural net, annotations, and recommended citations.

The neural net needs to be appraised for accuracy, in order to determine how it would perform when given datasets that have no annotations. The evaluation is done via the standard precision metric. A sampling out of the total number of training sets generated is selected, and the precision measured. This process is iteratively done in order to see which random sampling provides the best precision value.

The annotations are related to the performance of the neural net. For evaluating the quality of annotations about the relevance of keywords, we use the Kappa score to compare multiple

[1] Merriam-Webster's Medical Dictionary with Audio API available at http://www.dictionaryapi.com

[2] U.S. National Library of Medicine (NLM)'s Systematized Nomenclature of Medicine (SNOMED), International edition

[3] This dataset is no longer available on the BMJ website, but is still accessible via the Internet Archive's Wayback Machine project at http://web.archive.org/web/20160615110024/doc2doc.bmj.com

annotations made on the same dataset. As baseline, we use arbitrary annotations so that there is no planned correlation between a keyword and its relevance.

Finally, for evaluating the quality of the recommended citations, we use Normalized Discounted Cumulative Gain (NDCG)[13] to quantify whether the recommended citations indeed contain the keywords that were used to query PubMed. NDCG measures both the number of matching keywords in a search hit, as well as the usefulness of the hit based on its position in the results. As baseline, we use the "Title" sorting option within E-utilities, so that the top-*n* citations are sorted alphabetically. This sorting option will not rank citations based on number of keyword hits, but rather based on the title's alphabetic ordering.

Results

Firstly, the results of evaluating the neural net are presented. The Doc2Doc dataset contained a total of 1,400 discussions. Out of these discussions, 10 were used as the training dataset with varying numbers of message threads, averaging 9.80 threads per discussion. The total number of training datasets generated from the 10 discussions was 98. The neural net was trained by iteratively and randomly selecting 50 samples out of the 98 training datasets and choosing the model with the highest accuracy. Figure 4 shows the precision for 20 iterations of the sampling. The average precision was 55.87%, while the highest precision was 60.80%.

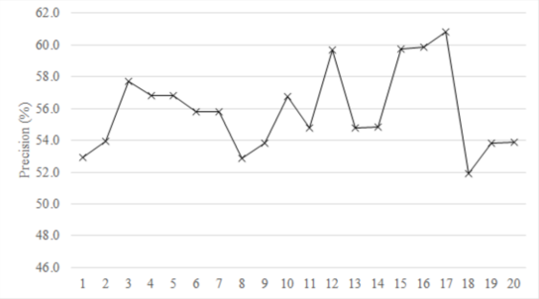


Figure 4 – Precision of Neural Net

Secondly, we present the evaluation of the manual annotations. We selected 10 discussions with annotations, and re-annotated them without cross-referencing the previous annotations. An average Kappa score of 51.87% was achieved, showing borderline acceptable agreement with the annotation method.

Figure 5 shows the Kappa scores for the 10 annotated discussions (M), their counter-part annotations (N), and baseline arbitrary annotations (Base).

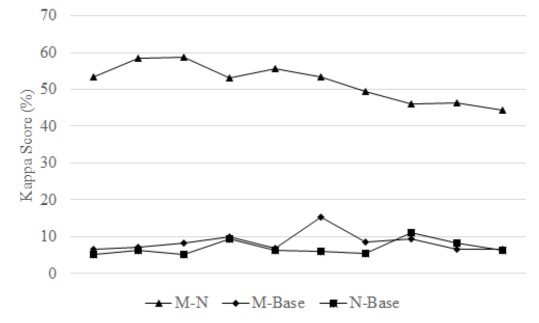


Figure 5 – Kappa Score for Annotations

Thirdly, we present statistics on the quality of the citations using NDCG. We arbitrarily selected 5 discussions and computed NDCG for each query to E-utilities, resulting in 32 data points, and an average number of 6.4 messages in the selected discussions. Figure 6 shows the NDCG values with E-utilities optimal sorting (Relevancy), which averaged 0.798, and also the baseline using E-utilities alphabetic ordering of the citation (Title). Similar results were also achieved for other iterations of arbitrarily selecting 5 discussions.

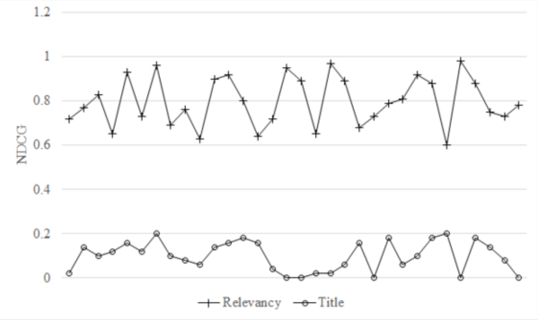


Figure 6 – NDCG for Recommended Citations

Discussion

The results show that our approach was able to retrieve citations based on forum discussion, while taking into account the relevance of keywords. Other methods for retrieving keywords in chat rooms rely on static heuristics such as a fixed time window[14]. As an example, only the last 5 messages could be used to determine the keywords for retrieving recommendations. Our method moves away from static heuristics and applies machine learning for dynamic retrieval of relevant keywords. Figure 7 shows how our neural net's keyword selection relates to a dynamic window metric for three randomly selected Doc2Doc forum discussions over 10 incoming messages. The window is set to the number of messages from the lastest to the one containing the oldest keyword selected by the neural net. Moreover, text summarization methods such as TextRank[15], and ensemble keyword extraction systems such as AlchemyAPI[4] are not adequate for extracting keywords to summarize a forum or chat room discussion because they do not take into consideration the decay in relevance of the keywords over time.

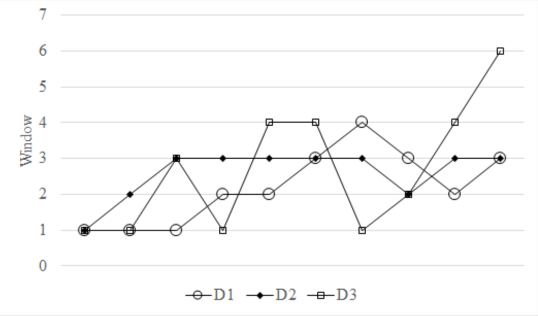


Figure 7 – Dynamic Window for Discussions D1, D2, D3

A limitation of our study is reliance on forum discussions to simulate chats, instead of actual real-time conversations. This

[4] AlchemyAPI available at http://www.alchemyapi.com

impacts our trained neural net because feature properties such as *firstTime* and *lastTime* will have relatively much smaller values for real-time discussions. Another area of improvement is increasing the training dataset size, which could improve precision. Furthermore, having multiple annotations would strengthen the quality of the neural net, and also provide insights into whether dynamic time windows are based on per-user preferences.

For future work, we aim to gather data from actual real-time chat logs for medical professionals from our health portal Cardea (currently under development), where patients and medics can share experiences, ask questions, and chat in real-time within specialized areas for patient-patient, patient-medic, and medic-medic discussions. Within Cardea, the recommender system can also be used to suggest related content and citations in the discussion forums. Moreover, we intend to add feedback mechanisms to PubMedReco so users can positively or negatively vote on the recommended citations.

Conclusion

This research presented the PubMedReco recommender system which can analyze a forum or chat discussion to extract the relevant medical terms, and then query PubMed to suggest citations that are related to the ongoing discussion. PubMedReco overcomes the limitation imposed on users of online discussion environments whereby users would have to leave their chat interface to search for medical articles in a new web browser window, and formulate an appropriate search query to retrieve relevant results. We demonstrated the feasibility of PubMedReco using BMJ's Doc2Doc forum datasets. We also evaluated our system to determine the quality of its neural net's relevance predictions, the training annotations used, and the recommended citations. To the best of our knowledge, the proposed system is the first of its kind in the medical domain. Unlike other real-time chat recommender systems surveyed that use static time window heuristics, the proposed system presents a novel and dynamic machine learning approach for determining keyword relevance within health forums and chat rooms.

Acknowledgements

The authors wish to acknowledge and thank the Alberta Machine Intelligence Institute (Amii), Edmonton, Alberta, Canada, for funding and supporting this research project.

References

[1] Z. Lu. PubMed and Beyond: A Survey of Web Tools for Searching Biomedical Literature. *J Database* (2011).

[2] S. J. Nelson. Medical Terminologies That Work: The Example of MeSH. *IEEE International Symposium on Pervasive Systems, Algorithms, and Networks* (2009), 380-384.

[3] Bethesda (MD): National Center for Biotechnology Information (US). Entrez Programming Utilities Help (2010). http://www.ncbi.nlm.nih.gov/books/NBK25501

[4] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, editors. *Recommender Systems Handbook*. Springer (2011).

[5] P. Jain, A. Bergen, L. Castaneda, and H. A. Muller. PALTask Chat: A Personalized Automated Context Aware Web Resources Listing Tool. *IEEE World Congress on Services* (2013), 154-157.

[6] F. Minami, M. Kobayashi, and T. Ito. A Proposal on Recommender System Based on Observing Web-Chatting. *New Challenges in Applied Intelligence Technologies*, Springer (2008), 87-96.

[7] W. Hong, L. Li, and T. Li. Product Recommendation with Temporal Dynamics. *Expert Systems with Applications* (2012), 12398-12406.

[8] B. Chandramouli, J. J. Levandoski, A. Eldawy, and M. F. Mokbel. StreamRec: A Real-Time Recommender System. *ACM SIGMOD Conference* (2011), 6-8.

[9] D. Kang, D. Han, N. Park, S. Kim, U. Kang, and S. Lee. Eventera: Real-time Event Recommendation System from Massive Heterogeneous Online Media. *IEEE International Conference on Data Mining Workshop* (2014), 1211-1214.

[10] G. Shani and A. Gunawardana. Evaluating Recommendation Systems. *Recommender Systems Handbook* (2011), 257-298.

[11] P. G. Campos, F. Díez, and I. Cantador. Time-Aware Recommender Systems: A Comprehensive Survey and Analysis of Existing Evaluation Protocols. *ACM J User Modeling and User-Adapted Interaction* (2014), 67-119.

[12] J. J. Hopfield. Artificial Neural Networks. *IEEE Circuits and Devices Magazine* (1988), 3-10.

[13] C. D. Manning, P. Raghavan, H. Schütze. Introduction to Information Retrieval, *Cambridge University Press* (2008)

[14] M. Montaner, B. Lopez, and J. L. De La Rosa. A Taxonomy of Recommender Agents on the Internet. *Artificial Intelligence Review* (2003), 285-330.

[15] R. Mihalcea, P. Tarau. TextRank: Bringing Order into Texts. *Association for Computational Linguistics* (2004).

[16] T. Mikolov, K. Chen, G. Corrado, J. Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv* (2013).

Address for correspondence

Hamman W. Samuel, Email: hwsamuel@ualberta.ca