Semantic News Recommendation Using WordNet and Bing Similarities

Michel Capelle
michelcapelle@gmail.com

Frederik Hogenboom
fhogenboom@ese.eur.nl

Alexander Hogenboom
hogenboom@ese.eur.nl

Flavius Frasincar
frasincar@ese.eur.nl

Erasmus University Rotterdam
PO Box 1738, NL-3000 DR
Rotterdam, the Netherlands

ABSTRACT

While traditionally content-based news recommendation was performed using the word vector space model, more recent approaches also take into account semantics, often through the use of semantic lexicons. However, named entities are rarely taken into account, as they are often absent in such lexicons. Nevertheless, they can play a crucial role in determining user interest for specific news articles. Therefore, in this work, we extend the state-of-the-art semantic lexicon-driven Semantic Similarity (SS) recommendation method by additionally considering named entities. First, as in SS, we calculate similarities between WordNet synonym sets in unread news items and synonym sets in read news items (stored in user profiles). Then, we use the page counts of named entities that are retrieved from the Bing Web search engine to compute named entity similarities between unread and read news items. Results show that our recommendation method, BingSS, outperforms SS in terms of F_1, precision, accuracy, and specificity.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information filtering, Relevance feedback*; I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods—*Representation Languages*

General Terms

Algorithms, Design, Performance

Keywords

Semantics-based recommender, Semantic Similarity, Bing Similarity, News

SAC'13 March 18-22, 2012, Coimbra, Portugal.

1. INTRODUCTION

The current Web provides us with an enormous and ever growing amount of valuable information on almost any subject of interest. As a consequence, arbitrary Web users must deal with an increasing load of data to process in order to obtain the items to their liking [26]. Although there is a history in the field of information technology to solve this user problem, Web information overload developments pose specific challenges that need to be faced. Digesting natural language Web sources has proven to be a non-trivial task due to difficulties related to comprehending the information content of unstructured data.

In the mid-1990s, the first recommender systems emerged, aiming to personalize and filter a flow of information to the interests or preferences of their users [1]. For this, similarities are measured between the content of unseen items (products, movies, news messages, etc.) and the contents of previously viewed items, commonly stored in user profiles acquired through monitoring browsing behaviour or through preference elicitation interfaces. While early recommender systems merely focused on content-based, lexical comparisons through the usage of vector space models and cosine similarities [25], more recent systems additionally take into account (domain) semantics by considering the meaning (i.e., sense) of words in an attempt to tackle machine interpretation problems of human-generated natural language texts [7, 12, 30]. A popular application of recommender systems is news recommendation and personalization, which is hence also the focus of our current endeavours.

In previous work [7], we investigated a semantics-based approach to news recommendation by making use of synonym sets (synsets) from WordNet [11], a large English lexical database. WordNet contains approximately 117,000 synsets, which are interlinked through semantic relations, such as synonymy, hyponymy, merynomy, troponymy, antonymy, and entailment. We exploited WordNet synsets in order to compute similarities between unread news articles and articles stored in user profiles in the Semantic Similarity (SS) method. An important drawback is the lack of support for named entities (e.g., 'Microsoft', 'Steve Ballmer', etc.) in WordNet. Nevertheless, a vast amount of named entities appear in news articles, and hence they could provide crucial information when constructing user profiles.

Therefore, in this paper, we propose the BingSS recommendation method, which extends our previously introduced

SS method by combining semantic information from WordNet with similarity based on page counts of named entities stemming from a Web search engine. Page counts are defined as the number of Web sites that contain one or more specific named entities. The more a pair of named entities co-occur on Web sites, the more likely it is that there is a similarity between both entities [5]. In our efforts, we make use of the Bing Web search engine[1], since Bing was the only large Web search engine that provided free access to search results and page counts at the time we performed this research. In order to evaluate its performance against the original SS method, BingSS is implemented in Ceryx [7], an extension to the Athena news recommendation component within the Hermes news personalization framework [12]. In this paper, we focus on improving our original SS method, and omit a comparison with other methods such as TF-IDF [25], SF-IDF [7], and CF-IDF [13], as these have been extensively compared in our previous work [7, 13].

This paper is organized as follows. First, Section 2 discusses related work. Then, we present our framework and its implementation in Sections 3 and 4. Next, we evaluate the results of our recommender in Section 5. Last, we conclude our paper and propose future work in Section 6.

2. RELATED WORK

Initial approaches to recommendation can be characterized as lexical approaches, which often borrow techniques from related fields like information retrieval and text mining. A widely used and acknowledged method is the term frequency – inverse document frequency (TF-IDF) [25]. The term weight in the vector space model is calculated by multiplying term frequencies with inverse document frequencies. The term frequency represents the frequency of term t in document d, and hence the more a term appears in a document, the more likely it is that the term is relevant to the topic of the document. The inverse document frequency is defined as the inverse of the frequency of term t throughout all documents d in set D, and hence the more a term appears in more documents, the less relevant it is to the topic of a single document.

However, as semantics are not taken into account, such lexical approaches have shown their limitations with respect to comprehending the meaning conveyed by specific words, which is crucial for recommender systems [12]. This drove the development of more interesting techniques that exploit semantics. These techniques make use of domain ontologies (i.e., ontologies that are specific for a certain domain) or lexical ontologies (i.e., semantic lexicons as WordNet [11]), containing synsets associated with corresponding lexical representations. In previous research, we extended the TF-IDF measure by using ontological concepts in CF-IDF [13] and semantic lexicon synsets in SF-IDF [7], and results have shown that our developed similarity-based SS method [7] performs equally good or better than the other methods.

The SS method for news recommendation compares WordNet synsets found in unread news items with WordNet synsets originating from all news items stored in a user profile by pairing the elements of the two sets with a common part-of-speech. In order to measure the similarity, a vector in the n-dimensional space is created containing all possible combinations of WordNet synsets from an unread news item on the one hand, and the WordNet synsets from a user profile on the other hand. Subsequently, a subset is extracted that contains all the combinations which have a common part-of-speech. Then, for every combination in the subset, a specific semantic similarity measure is used. The final similarity rank of an unread news item is defined as the sum of all the combinations' similarities divided by the total number of combinations.

We distinguish between five semantic similarity measures, i.e., Jiang & Conrath [19], Leacock & Chodorow [20], Lin [22], Resnik [24], and Wu & Palmer [29]. Each measure evaluates the semantic distance between two synsets (represented as nodes in a taxonomy, i.e., a hierarchy of 'is-a' relationships between nodes), where for instance 'turkey' should be closer to 'animal' than to 'boat'. The measures of Jiang & Conrath, Resnik, and Lin are based on the information content of the nodes, while Leacock & Chodorow and Wu & Palmer make use of the path length between the nodes. In earlier work [7], we identified the Wu & Palmer method as the best performing similarity measure for SS.

An alternative to measuring content-based similarities, is using similarities that are based on page counts that are gathered by Web search engines like Google[2] or Bing. Page counts are defined as the number of Web sites that contain specific entities. The more a pair of entities co-occur on Web sites, the more likely it is that there is a similarity between both entities [5]. A frequently studied similarity measure based on page counts is the Normalized Google Distance (NGD) [8, 9, 28], which is a normalized semantic distance between 0 and 1 that is calculated using probabilities related to the number of hits associated with the two separate entities, the number of hits associated with the two entities appearing together, and the number of indexed Web pages. The NGD is based on Kolmogorov complexity, normalized information distance, and normalized compression distance. Unfortunately, Google's API was not available any more as a free service, so we have used Bing, which still offered an API for its search service for free at the time we carried out this research.

3. FRAMEWORK

The SS and BingSS recommendation methods are implemented in the Ceryx framework, which is an extension to the Athena framework [16], allowing for semantics-based recommendation within the Hermes news personalization framework [12]. Our semantics-based methods make use of a user profile, which is defined as a set of read news items. Based on the assumption that users only read articles of interest, the user profile is considered to be representative for the user preferences. Hence, upon reading a previously unseen news item, a user profile can be constructed or updated by adding the item it.

3.1 SS Recommendation

When applying SS recommendation, semantic similarity is measured between a set U of k WordNet synsets $u_1, \ldots, u_k$ that are derived from an unread news item, and a set R of l WordNet synsets $r_1, \ldots, r_l$ derived from a user profile. Subsequently, we create a vector V containing all possible pairs between the synsets found in news item U and user profile R, i.e.,

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

[2] http://www.google.com

$$V = (\langle u_1, r_1\rangle, \cdots, \langle u_k, r_l\rangle) \; \forall \; u \in U, \; r \in R \; , \qquad (1)$$

where u_i represents a WordNet synset in the unread news item, r_j denotes a WordNet synset in the user profile, and k and l are the number of WordNet synsets in the unread news item and user profile, respectively.

Next, we create a subset W where all pairs have a common part-of-speech, which is defined as

$$W \subseteq V \; \forall \; (u, r) \in W : POS(u) = POS(r) \; , \qquad (2)$$

where $POS(u)$ represents the part-of-speech of synset u in the unread news item, and $POS(r)$ is the part-of-speech of synset r in the user profile.

Last, for every pair, a similarity score is computed by means of a semantic similarity measure. In our current work, we make use of the Wu & Palmer similarity measure [29], as this proved to be the best performing one in our previous work [7]. The Wu & Palmer similarity measure makes use of the distance between two synsets in a semantic graph, and is defined as

$$sim_{WP}(u, r) = \frac{2 \times depth(LCS(u, r))}{length(u, r) + 2 \times depth(LCS(u, r))} \; , \qquad (3)$$

where $depth(LCS(u, r))$ denotes the depth of the lowest common subsumer of both synsets in the WordNet graph, and $length(u, r)$ represents the path length between the two synsets in the graph.

The similarity score for an unread news item is defined as the sum of similarity scores for all pairs, divided by the number of pairs, i.e.,

$$sim_{SS} = \frac{\sum\limits_{(u,r)\in W} sim_{WP}(u, r)}{|W|} \; , \qquad (4)$$

where $|W|$ is the number of WordNet synset pairs within the unread news item and the user profile. The unread news items with similarity scores exceeding a predefined cut-off value are recommended to the user.

3.2 BingSS Recommendation

When extending the SS recommendation method to BingSS, we do not only consider words stemming from a semantic lexicon (i.e., WordNet), but we additionally take into account named entities which are usually not found in semantic lexicons. The BingSS approach is similar to the SS approach in that it also computes the semantic similarity between synsets from a semantic lexicon, yet the difference is that BingSS only takes into account the synset pairs with the highest similarity scores, and hence we redefine (4) in order to reflect these changes. Now, semantic similarity sim_{SS} is defined as

$$sim_{SS} = \frac{\sum\limits_{(u,r)\in W} sim_{WP}(u, r) \in TOP_W^{\beta_{SS}}}{|TOP_W^{\beta_{SS}}|} \; , \qquad (5)$$

where $TOP_W^{\beta_{SS}}$ is the set of synset pairs with the highest similarity in W, and β_{SS} is a predefined positive integer (optimized during testing), representing the top-β_{SS} similarities from pairs in W. We assume here that not all named entities appearing in a news item are relevant for defining the user interests. For example, for EU news on the Greek debt, the EU named entity is not relevant when the user's interest is on the Greek debt.

Additionally, BingSS takes care of handling named entities in news items. Named entities are often not included in WordNet, and are hence derived from news items by means of a named entity recognizer. They are captured in a set U, containing k named entities $u_1, \ldots, u_k$ for a certain news item. Named entities are also retrieved from a user profile, and are stored in set R, containing l named entities $r_1, \ldots, r_l$. Subsequently, sets V – containing all possible pairs between the synsets found in news item U and user profile R – and W – containing all pairs that have a common part-of-speech – can be constructed as specified in (1) and (2).

For every pair (u, r) in V we compute a similarity score using the Point-Wise Mutual Information (PMI) co-occurrence similarity measure [6] based on the page counts retrieved from the Bing Web search engine. The page count is the number of Web sites that contain the named entities u or r or the pair of named entities (u, r), found by the Bing Web search engine. PMI is a measure of association between two probabilities. It measures the difference between the actual and expected joint probability of the occurrence of two named entities in a query on a Web search engine, based on the two single probabilities of the two named entities while assuming independence. In our case, we define the PMI-based similarity score as

$$sim_{PMI}(u, r) = \log \frac{\frac{c(u,r)}{N}}{\frac{c(u)}{N} \times \frac{c(r)}{N}} \; , \qquad (6)$$

where $c(u, r)$ is the page count for the named entity pair (u, r), and $c(u)$ and $c(r)$ are the page counts for the named entities u and r. N is the total number of indexed Web pages by Bing (approximately 15 billion [14]).

The final similarity score of an unread news item is calculated as

$$sim_{Bing} = \frac{\sum\limits_{(u,r)\in V} sim_{PMI}(u, r) \in TOP_V^{\beta_{Bing}}}{|TOP_V^{\beta_{Bing}}|} \; , \qquad (7)$$

where $TOP_V^{\beta_{Bing}}$ is the set of top-β_{Bing} pairs with the highest similarity in V, and β_{Bing} is a predefined positive integer representing the top-β_{Bing} similarities from pairs in set V.

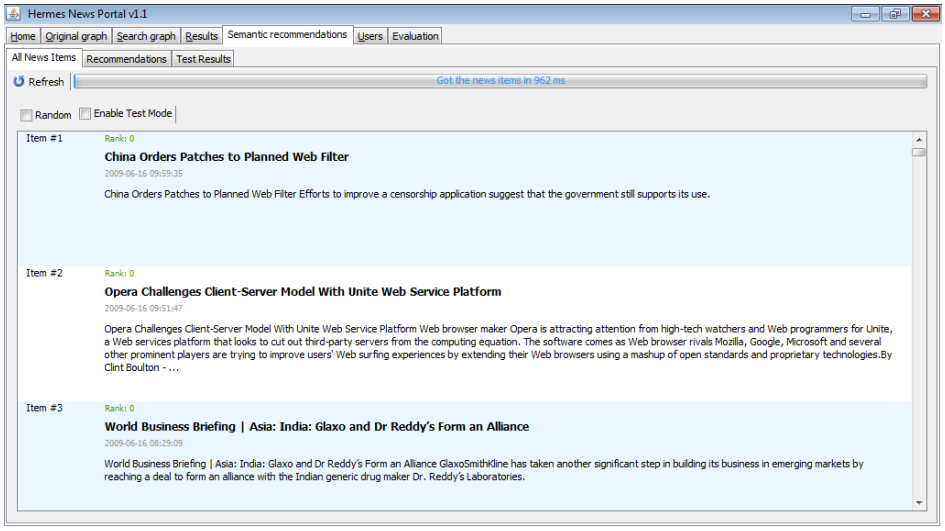
Last, we calculate the BingSS similarity score sim_{BingSS} by combining the SS and Bing similarity scores by means of a weighted average, i.e.,

$$sim_{BingSS} = \alpha \times sim_{Bing} + (1 - \alpha) \times sim_{SS} \; , \qquad (8)$$

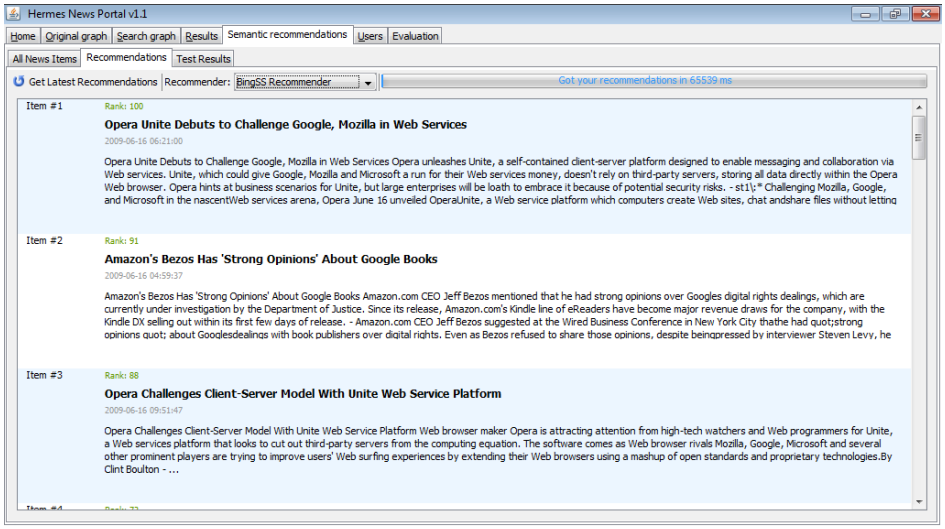
where α is a predetermined weight. When employing the BingSS similarity score, news items with scores exceeding a specific, predefined cut-off value are recommended.

4. IMPLEMENTATION

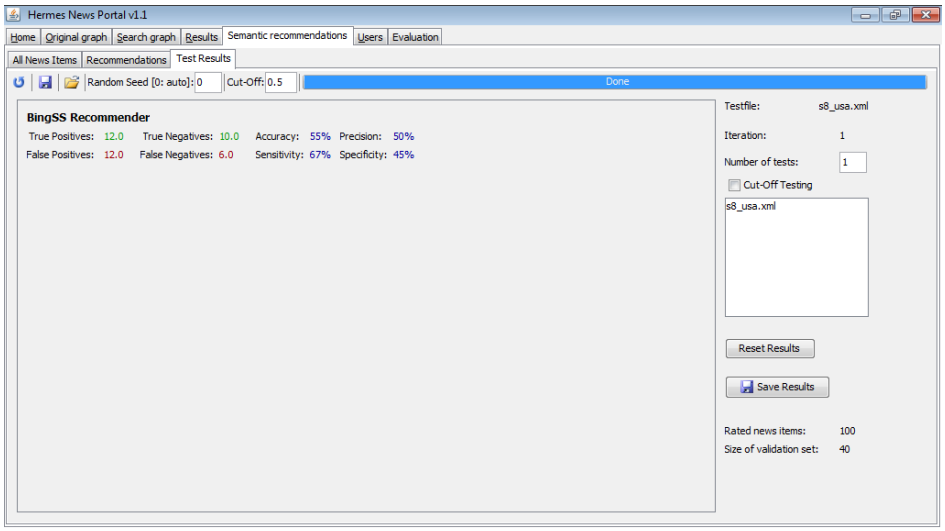
Our framework is implemented as an extension to the Ceryx [7] plugin created for Athena, a news recommendation component employed in the implementation of the Hermes framework, i.e., the Hermes News Portal (HNP) [12]. The HNP application allows users to formulate and execute queries using a domain ontology in order to retrieve relevant news items, and is a stand-alone, Java-based tool which makes use of various Semantic Web technologies. Its internal knowledge base is an OWL domain ontology constructed by domain experts, and is queried using extended



(a) Overview of all news items.



(b) Resulting recommended news items.



(c) Recommender evaluation results.

Figure 1: Ceryx user interface.

SPARQL queries. In HNP, the classification of the news articles is done using the GATE natural language processing software [10] and the WordNet [11] semantic lexicon.

4.1 Ceryx User Interface

The Ceryx plugin provides a tabbed user interface for semantic news recommendations, and presents the user an overview of all available news items, recommended news items, and evaluation results. In the first tab, depicted in Fig. 1(a), the user is able to browse through all the news items that are found by the system in specified RSS feeds. Each item is displayed with a title, date, and abstract. Whenever an item is selected, the Web page containing the news item is opened in the user's default Web browser. The news item is then added to the user profile as interesting item.

In the recommendations tab (Fig. 1(b)), the user can choose from several recommendation methods which generate different lists of recommended news items. The Ceryx plugin provides support for many recommenders, e.g., TF-IDF, CF-IDF, and SS, and we have extended the plugin to also include BingSS. The resulting list of items looks similar to the list shown in the first tab, yet also displays similarity scores ranging from 0 to 100, with 0 being the associated with the lowest ranking news items, and 100 with the highest ranking news items (best reflecting the user's interests). Additionally, the list is sorted on similarity scores.

The last tab displays test results, and enables the user to test the recommendation methods on different performance measures. The user can load a test file, which contains a human judgment on every news item whether it should be marked for recommendation or not based on a pre-specified user profile. The recommendation methods are evaluated on accuracy, precision, sensitivity, and specificity. In the depicted screenshot in Fig. 1(c), Ceryx displays an evaluation of the BingSS recommender.

4.2 SS Recommendation

The implementation of the SS recommender requires the news extracted synsets. These are obtained through part-of-speech tagging, stop word removal, and word sense disambiguation. Part-of-speech tagging is performed by means of the Stanford Log-Linear Part-of-Speech Tagger [27], which has a 97.24% accuracy on Penn Treebank WSJ data. Stop words (i.e., non-meaningful words) are subsequently removed from the news item using a list of stop words that can be found in the Onix Text Retrieval Toolkit API reference documents [21]. Word sense disambiguation is performed using an implementation [18] of the Lesk algorithm [3]. After these initial processing steps, our implementation computes pairwise similarities with the Wu & Palmer similarity measure [29], using a Java implementation by Hope [15], which is a conversion from a Perl implementation [23]. Final similarity scores are calculated by averaging all pairwise similarities.

4.3 BingSS Recommendation

In contrast to the implementation of the SS recommender, the BingSS recommendation method does not only require synsets (obtained in the same way as for the SS recommender's implementation), but also named entities. These are extracted using the named entity recognizer from Aliasi's LingPipe 4.1.0 [2]. Synset similarities are subsequently calculated in a similar way as is the case for our SS implementation, although final similarity scores are computed by taking the average of the top-β_{SS} similarity scores. For named entities, we compute the PMI similarities by using page counts retrieved from queries on named entities that are performed on the Bing API 2.0 [4]. Final entity similarity scores are calculated by taking the average of the top-β_{Bing} scores. Last, scores are weighted using an optimized α, maximizing F_1-scores.

5. EVALUATION

We now continue with an evaluation of the performance of our BingSS approach when compared to the SS news recommender. Our experiments are based on 100 news articles that are collected from a Reuters news feed on technology companies. Three users (Economics & Informatics students from the Erasmus University of Rotterdam) were presented with the articles, and had to indicate whether a news article is related to one of the given topics. We distinguish between eight topics, i.e., 'Asia or its countries', 'financial markets', 'Google or competitors', 'Internet or Web services', 'Microsoft or competitors', 'national economies', 'technology', and 'United States of America'. Out of these user ratings, a user profile was constructed for every topic using a minimum inter-annotator agreement (IAA) of 66% (i.e., two out of three users stated that a news item is relevant for a topic). Table 1 displays the resulting number of non-interesting and interesting news items per topic (*Items–* and *Items+*, respectively), as well as their associated agreements (i.e., *IAA–* and *IAA+*, respectively).

In order to evaluate the BingSS recommendation method, we compare its performance to the performance of SS recommendation in terms of accuracy, precision, recall, specificity,

Table 1: The number of non-interesting news items (*Items–*), the number of interesting news items (*Items+*), and their associated inter-annotator agreements (*IAA–* and *IAA+*, respectively) for each topic.

Topic	Items–	Items+	IAA–	IAA+
Asia or its countries	79	21	0.97	1.00
Financial markets	76	24	0.68	0.75
Google or competitors	74	26	0.95	1.00
Internet or Web services	74	26	0.92	0.96
Microsoft or competitors	71	29	0.96	1.00
National economies	67	33	0.85	0.94
Technology	71	29	0.87	0.86
United States of America	55	45	0.84	0.87
Average	70.9	29.1	0.88	0.92

and F_1. Performances are evaluated for the individual topics using a range of cut-off values (i.e., items with similarity scores above a specific value are recommended). Additionally, we optimize the cut-off values for both recommenders as well as parameters for BingSS per topic by employing a supervised learning approach based on the maximization of the resulting F_1-scores, and hence our data set is randomly but proportionally divided into a test set (40%) and training set (60%).

In order to optimize the parameters for BingSS using our training set, we evaluate β_{SS} and β_{Bing} values ranging from 1 to the maximum number of compared pairs with a step size of 1, and the α parameter is evaluated within the range of 0 to 1, with a step size of 0.01. The optimized β_{SS} value is equal to the total number of evaluated synset pairs, while the optimized β_{Bing} value is optimized to 9. Last, scores are weighted using an optimized α of 0.72, giving a substantial weight to Bing similarities.

When plotting the global F_1 measure against the cut-off value, we obtain the graph that is depicted in Fig. 2. Especially for high cut-off values, BingSS outperforms SS, whereas for lower cut-off values, differences between both recommenders are fairly small. This phenomenon could be explained by the fact that for low cut-off values, the emphasis is on high recall and not so much on high precision. High recall values are obtained for both recommendation methods, because more recommendations are made using this setting, hence increasing the probability that the recommended articles are in fact correct. This phenomenon diminishes the influence of the proposed additions to SS in our BingSS recommendation method. On the other hand, when using high cut-off values, the emphasis is on high precision, hereby favoring BingSS, as it also considers named entities, contributing to a higher precision.

Table 2 supports these observations, as using the optimal cut-off value leads to a configuration of BingSS that has a considerably higher precision than the best performing SS recommendation method configuration. Optimizing the cut-off value for SS yields a value of 0.49, with an associated F_1-score of 54.3%, while the optimized cut-off value of BingSS is 0.63, which gives a higher F_1-score of 58.1%. Not only does BingSS outperform SS in terms of F_1, but also in terms of precision, accuracy, and specificity.

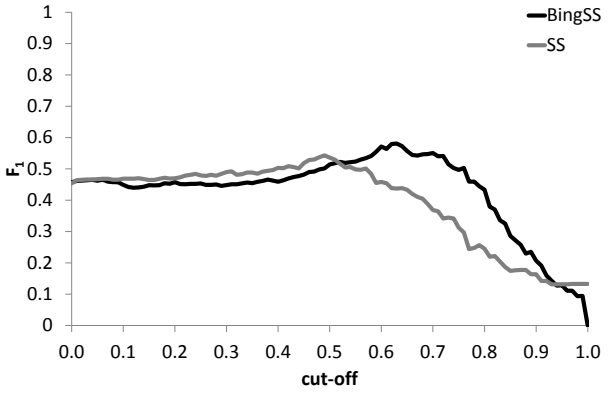


Figure 2: Averaged F_1-measure performance for various cut-off values of the SS and BingSS recommendation methods.

Table 2: Averaged test results for SS and BingSS recommendation using optimized cut-off values.

Measure	SS @ 0.49	BingSS @ 0.63
Accuracy	64.2%	73.1%
Precision	44.0%	54.0%
Recall	73.1%	62.9%
Specificity	60.2%	77.4%
F_1-measure	54.3%	58.1%

6. CONCLUSIONS

In this paper, we explored the possibilities of extending the state-of-the-art semantic similarities-based approach to news recommendation, SS, which makes use of similarities between semantic lexicon synsets found in news items and user profiles. As named entities usually do not appear in semantic lexicons, in our proposed approach, we additionally take into account named entities, by using a similarity based on page counts retrieved from the Bing Web search engine. Results show that our recommendation method, BingSS, outperforms SS in terms of F_1, precision, accuracy, and specificity. Additionally, our experiments show that for BingSS, named entity similarities are more important than synset similarities, as the former have a weight of 0.72, compared to 0.28 for the latter.

For future work, we would like to investigate the possibility to measure page counts not only for named entities, but also for semantic lexicon synsets, and compare the accuracy of the SS approach with this new approach. Here, we basically compare the Wu & Palmer similarity with the Bing similarity in an extrinsic way. Additionally, it would be worthwhile to perform additional analysis on similar extensions to other recommendation methods, such as TF-IDF, CF-IDF, and SF-IDF. Last, one could also consider using other co-occurrence measures for named entities besides the PMI measure, e.g., Jaccard similarity [17].

7. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.

[2] Alias-i. LingPipe 4.1.0. `http://alias-i.com/lingpipe`, 2008.

[3] S. Banerjee and T. Pedersen. An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet. In A. F. Gelbukh, editor, *4th International Conference on Computational Linguistics and Intelligent Text Processing (CICLING 2002)*, pages 136–145. Springer-Verlag, 2002.

[4] Bing. Bing API 2.0. `http://www.bing.com/developers/s/APIBasics.html`, 2012.

[5] D. Bollegala, Y. Matsuo, and M. Ishizuka. Measuring Semantic Similarity between Words Using Web Search Engines. In C. L. Williamson, M. E. Zurko, P. F. Patel-Schneider, and P. J. Shenoy, editors, *16th International Conference on World Wide Web (WWW 2007)*, pages 757–766. ACM, 2007.

[6] G. Bouma. Normalized (Pointwise) Mutual Information in Collocation Extraction. In C. Chiarcos, R. E. de Castilho, and M. Stede, editors, *Biennial*

GSCL Conference 2009 (GSCL 2009), pages 31–40. Gunter Narr Verlag Tübingen, 2009.

[7] M. Capelle, M. Moerland, F. Frasincar, and F. Hogenboom. Semantics-Based News Recommendation. In R. Akerkar, C. Bădică, and D. Dan Burdescu, editors, *2nd International Conference on Web Intelligence, Mining and Semantics (WIMS 2012)*. ACM, 2012.

[8] R. Cilibrasi and P. M. B. Vitányi. Similarity of Objects and the Meaning of Words. In J. yi Cai, S. B. Cooper, and A. Li, editors, *3rd International Conference on Theory and Applications of Models of Computation (TAMC 2006)*, volume 3959 of *Lecture Notes in Computer Science*, pages 21–45, 2006.

[9] R. Cilibrasi and P. M. B. Vitányi. The Google Similarity Distance. *IEEE Transactions on Knowledge and Data Engineering*, 19(3):370–383, 2007.

[10] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *40th Anniversary Meeting of the Association for Computational Linguistics (ACL 2002)*, pages 168–175. Association for Computational Linguistics, 2002.

[11] C. Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[12] F. Frasincar, J. Borsje, and L. Levering. A Semantic Web-Based Approach for Building Personalized News Services. *International Journal of E-Business Research*, 5(3):35–53, 2009.

[13] F. Goossen, W. IJntema, F. Frasincar, F. Hogenboom, and U. Kaymak. News Personalization using the CF-IDF Semantic Recommender. In R. Akerkar, editor, *International Conference on Web Intelligence, Mining and Semantics (WIMS 2011)*. ACM, 2011.

[14] I. R. Group. WordWideWebSize.com. `http://www.worldwidewebsize.com`, 2012.

[15] D. Hope. Sussex University: NLP Lab, Homepage. `http://www.cogs.susx.ac.uk/users/drh21/`, 2012.

[16] W. IJntema, F. Goossen, F. Frasincar, and F. Hogenboom. Ontology-Based News Recommendation. In F. Daniel, L. M. L. Delcambre, F. Fotouhi, I. Garrigós, G. Guerrini, J.-N. Mazón, M. Mesiti, S. Müller-Feuerstein, J. Trujillo, T. M. Truta, B. Volz, E. Waller, L. Xiong, and E. Zimányi, editors, *International Workshop on Business intelligencE and the WEB (BEWEB 2010) at Thirteenth International Conference on Extending Database Technology and Thirteenth International Conference on Database Theory (EDBT/ICDT 2010)*. ACM, 2010.

[17] P. Jaccard. Étude Comparative de la Distribution Florale dans une Portion des Alpes et des Jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 37:547–579, 1901.

[18] A. S. Jensen and N. S. Boss. Textual Similarity: Comparing Texts in Order to Discover How Closely They Discuss the Same Topics. Bachelor's Thesis, Technical University of Denmark, 2008.

[19] J. J. Jiang and D. W. Conrath. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In *10th International Conference on Research in Computational Linguistics (ROCLING 1997)*, pages 19–33, 1997.

[20] C. Leacock and M. Chodorow. *WordNet: An Electronic Lexical Database*, chapter Combining Local Context and WordNet Similarity for Word Sense Identification, pages 265–283. MIT Press, 1998.

[21] Lextek. Onix Text Retrieval Toolkit – API Reference. `http://www.lextek.com/manuals/onix/stopwords1.html`, 2012.

[22] D. Lin. An Information-Theoretic Definition of Similarity. In J. W. Shavlik, editor, *15th International Conference on Machine Learning (ICML 1998)*, pages 296–304. Morgan Kaufmann, 1998.

[23] T. Pedersen, S. Patwardhan, and J. Michelizzi. WordNet::Similarity – Measuring the Relatedness of Concepts. In D. L. McGuinness and G. Ferguson, editors, *19th National Conference on Artificial Intelligence (AAAI 2004)*, pages 1024–1025. AAAI Press / MIT Press, 2004.

[24] P. Resnik. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *14th International Joint Conference on Artificial Intelligence (IJCAI 1995)*, pages 448–453. Morgan Kaufmann, 1995.

[25] G. Salton and C. Buckley. Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[26] J. Schafer, J. Konstan, and J. Riedi. Recommender Systems in E-Commerce. In *1st ACM Conference on Electronic Commerce (ACM-EC 1999)*, pages 158–166, 1999.

[27] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLTNAACL 2003)*, pages 252–259, 2003.

[28] P. M. B. Vitányi. Universal Similarity. In *2005 IEEE Information Theory Workshop (ITW 2005)*, pages 6–10, 2005.

[29] Z. Wu and M. S. Palmer. Verb Semantics and Lexical Selection. In *32nd Annual Meeting of the Association for Computational Linguistics (ACL 1994)*, pages 133–138. Association for Computational Linguistics, 1994.

[30] C.-N. Ziegler. Semantic Web Recommender Systems. In W. Lindner, M. Mesiti, C. Türker, Y. Tzitzikas, and A. Vakali, editors, *EDBT 2004 Workshops*, volume 3268 of *Lecture Notes in Computer Science*, pages 78–79. Springer, 2004.